EMILIE LEBARBIER

Quelques approches pour la détection de ruptures à horizon fini

Thèses d'Orsay, 2002

<http://www.numdam.org/item?id=BJHTUP11_2002__0624__P0_0>

ORSAY
N° d'ordre : 6950

UNIVERSITÉ DE PARIS-SUD
U.F.R SCIENTIFIQUE D'ORSAY

THÈSE

présentée
pour obtenir
Le GRADE de DOCTEUR EN SCIENCES
DE l'UNIVERSITÉ PARIS XI ORSAY
Spécialité : Mathématiques

par

Emilie Lebarbier

Sujet : **QUELQUES APPROCHES POUR LA DÉTECTION DE RUPTURES À HORIZON FINI**

Soutenue le **04 Juillet 2002** devant la Commission d'examen composée de :

M. Grégoire Gérard	Rapporteur
Mme. Huet Sylvie	Rapporteur
M. Lavielle Marc	Directeur
M. Massart Pascal	Président
M. Poggi Jean-Michel	Examinateur
M. Prum Bernard	Examinateur

à ma famille,

Remerciements

Je remercie Marc Lavielle pour la confiance qu'il m'a accordée et pour m'avoir fait découvrir le monde des ruptures. J'ai pu apprécier durant ces 4 années sa disponibilité et son intérêt pour les mathématiques appliquées.

Je tiens à remercier Pascal Massart de m'avoir fait l'honneur de présider ce jury, et pour m'avoir fait découvrir les secrets de la sélection de modèle. Merci pour les discussions enrichissantes que nous avons eues, pour sa confiance et son enthousiasme.

Je tiens à remercier Gérard Grégoire et Sylvie Huet d'avoir accepté de rapporter ma thèse. Je leur suis extrêmement reconnaissante de l'intérêt qu'ils ont porté à mon travail. Je remercie aussi Bernard Prum et Jean-Michel Poggi d'avoir accepté de faire partie de mon jury de thèse.

Je remercie les universités d'Evry et d'Orsay de m'avoir accueillie en tant que vacataire et ATER.

Je tiens à exprimer mon amitié à toutes les personnes du laboratoire de Mathématiques d'Orsay que j'ai rencontrées. En particulier à

À Elodie et Servane pour leur collaboration si enrichissante, répondant avec patience à tous mes nombreux "Et pourquoi?", pour avoir supporté mes doutes réguliers, pour leurs soutiens, et surtout pour leur amitié.

À Catherine et Isabelle pour tous les services qu'elle m'ont rendus et pour les discussions.

À Béatrice, Cécile et Jean Coursol pour leur aide concernant mes enseignements, ainsi qu'à Sophie pour toutes ces heures passées à discuter du prochain "TP/TD" et pour m'avoir fait partager sa passion de l'enseignement.

À Yves Misiti pour m'avoir sauvée continuellement de mes problèmes informatiques.

Aux co-occupants des bureaux 112 et 114. Particulièrement à Karelle pour son écoute et son soutien, à Violaine et Marc pour avoir partagé tous ces bons moments dans "notre salle informatique", et aussi pour le "J'te ramène un dessert?". Merci aussi aux membres des autres bureaux pour tous les bavardages et tous leurs "éclairages" mathématiques.

Enfin un grand merci,

À Marie-Laure pour ses innombrables relectures, ses conseils, sa patience et enfin surtout pour son amitié.

À tous mes Z'amis pour leur amitié ô combien essentielle ... pour leurs pensées positives ... les comment ça va? du matin ... pour leurs écoutes "no-limit" ... pour m'avoir maintes fois rassurée!

À toute ma "grande" famille. Merci pour les p'tit mots et les encouragements. Particulièrement, à Lélène et Fred pour les clins d"'yeux" toujours aux bons moments, à Dona et Clémence pour les folles parties NBA, à Popo et Vincent pour tous les SMS, enfin à mes parents, pour m'avoir toujours soutenue et épaulée, pour notre complicité. Merci.

Abstract:
This thesis is devoted to the detection of multiple change-points. The first part considers the case of change-points in the mean for Gaussien signal. In Chapter 1, we adopt a bayesien approach : the posterior distribution of the change-points sequence is estimated by Monte Carlo by Markov chain methods. A stochastic version of the EM algorithm is used for estimating the hyper-parameters of the model. In Chapter 2, the change-points and the means are simultaneously estimated by recovering the underlying piecewice constant function denoted by s which is a penalized least-square estimator. We give the penalty form and a non-asymptotic risk bound for the corresponding penalized estimator. The penalty depends on two constants and the noise level which are unknown. In Chapter 3, given known variance we determine the optimal values for the two constants for any function s and size of sample by simulation study. In Chapter 4, rather than estimating the noise level, a heuristic method is used to estimate the penalty itself using the data. We calibrate it and test it on various simulated data sets. In Chapter 5, we propose an hybrid algorithm combining the CART algorithm and a partial exhaustive search for the application for large samples.

The second part considers the case of change-points in the distribution of a sequence of independant random variables. In Chapter 6, we associate the distribution to a function s which we estimate by maximizing the penalized likelihood and we give a risk bound for the obtained estimator. In Chapter 7, we adapt the hybrid algorithm proposed in Chapter 5 to detect homogeneous regions in DNA sequences of two bacteria.

Keywords : Change-points detection - Penalized contrasts - Model selection - MCMC algorithm - SAEM algorithm - CART algorithm - DNA sequences segmentation.

AMS Classification: 62G08, 62-07, 62J02, 62L20, 62P10, 62M05, 68Q25

Table des matières

Introduction

Le problème de détection de rupture est très important en statistique et fait l'objet d'un grand nombre d'études dans différents contextes depuis plus de quarante ans (voyez, par exemple, [4, 14, 21, 42] pour un état de l'art). Cet intérêt est largement motivé par les nombreuses applications pratiques dans des domaines aussi variés que le domaine médical [4, 10, 39, 29, 16, 60, 53], la géophysique [63], l'industrie [67], la biologie [26, 12, 3, 11], la finance [5, 44], etc....En effet, de nombreux phénomènes sont soumis à des changements brusques. Il s'agit alors de détecter et localiser ces changements ou ruptures, mais aussi d'identifier leur nature afin d'analyser le phénomène ou d'en comprendre sa structure sous-jacente.

Nous nous intéressons dans cette thèse à deux modèles de ruptures : dans les deux première parties, nous considérons le cas particulier de ruptures dans la moyenne d'un signal gaussien et dans la troisième celui de ruptures dans la distribution marginale d'une suite d'observations prenant un nombre fini de valeurs.

Le problème de détection de ruptures peut être formulé "en ligne" où la détection se traite au fur et à mesure que les observations apparaissent, ou "hors ligne" où elle se traite sur les observations complètes simultanément. Dans cette thèse, nous nous plaçons uniquement dans la dernière situation. Historiquement, les auteurs se sont d'abord penchés sur le cas d'une rupture unique, puis sur celui d'un nombre connu de ruptures. Quand le nombre de ruptures est inconnu, la situation devient plus complexe. Plusieurs auteurs se sont consacrés à ce problème et ont proposé différentes méthodes. Ils obtiennent généralement des estimateurs ayant de bonnes propriétés quand la taille de l'échantillon est grande. Bien sûr, en pratique, elle n'est pas toujours assez grande pour pouvoir considérer ces estimateurs.

Dans une optique d'application, il s'agit de proposer des méthodes d'estimation des instants de ruptures en se plaçant dans un cadre non-asymptotique, c'est-à-dire pour une taille d'échantillon fini. Nous considérons deux approches différentes : une approche bayésienne, qui fait l'objet de la partie 1 et une approche de sélection de modèle par pénalisation, qui fait l'objet de la partie 2. Nous reprenons cette dernière approche dans la partie 3.

Détection de ruptures dans la moyenne d'un signal gaussien

Nous considérons le modèle suivant :

$$y_t = s(t) + \epsilon_t \quad t = 1, \ldots, n$$

où $\epsilon = (\epsilon_t)$ est une suite de variables aléatoires indépendantes et identiquement distribuées de loi gaussienne centrée de variance σ^2. La fonction s est supposée constante par morceaux. Ainsi, il existe des instants $t_0 = 0 < t_1 < \ldots < t_{K_r} = n$ et une suite finie $(s_1, \ldots, s_{K_r})$ tels que

$$s = \sum_{k=1}^{K_r} s_k \mathbb{1}_{I_k} \text{ avec } I_k =]t_{k-1}, t_k]$$

L'objectif est l'estimation de la fonction inconnue s. Notons que l'estimation des instants de ruptures et des moyennes recouvrent l'estimation de s.

Estimation par méthodes de Monte Carlo par chaînes de Markov

Dans le chapitre 1, nous adoptons une approche bayésienne. Nous introduisons une suite $r = (r_t)$ prenant la valeur 1 aux instants de ruptures et 0 entre deux sauts (paramétrisation proposée par Lavielle [39] ou Tourneret *et al.* [62, 63]). Les suites inconnues des configurations des instants de ruptures r et des moyennes $s = (s_k)$ sont considérées comme des variables aléatoires et sont munies d'une distribution a priori dont l'ensemble des paramètres est noté θ. L'objectif est l'estimation de la distribution a posteriori de la suite des instants de ruptures conditionnellement au signal observé $p(r|y;\theta)$ pour un θ donné. Ceci permettra d'extraire des caractéristiques intéressantes telles que la probabilité d'avoir une rupture en un instant donné t, ou la distribution a posteriori du nombre de ruptures. La distribution a posteriori $p(r|y;\theta)$ devient très vite incalculable dès lors que la taille du signal est grande et nous l'estimons par un algorithme de Monte Carlo par chaîne de Markov (MCMC) de type Hasting-Métropolis. Cet algorithme consiste simplement à échantillonner les suites de 0 et de 1 de longueur fixe $n - 1$. De plus, l'application de cet algorithme rapide à basse "température" permet d'estimer les configurations des instants de ruptures de plus grande probabilité.

Nous nous intéressons ensuite à la distribution jointe a posteriori $p(r, s|y;\theta)$. Nous proposons un algorithme hybride simple combinant l'algorithme d'Hasting-Métropolis avec l'échantillonnage de Gibbs [6, 27] prenant en compte la structure hierarchique naturelle :

$$p(s, r|y;\theta) = p(r|y;\theta)p(s|r, y;\theta).$$

Une étude de simulation montre que cet algorithme converge plus rapidement que le "reversible jump MCMC" introduit par Green [31], qui lui ne prend pas en compte cette

structure.
Néanmoins, cela mène à une version lissée du signal s alors que nous souhaitions obtenir une fonction constante par morceaux. Une autre approche consiste à estimer la moyenne conditionnellement à la configuration des ruptures de plus grande probabilité. Cette approche a plus de sens dans ce contexte et fournit une bonne estimation de la suite des moyennes.

Ces méthodes supposent la connaissance des paramètres des distributions a priori θ. Plutôt que de les choisir arbitrairement, nous proposons une procédure couplant une version stochastic de l'algorithme de EM, proposé par Delyon, Lavielle et Moulines [22] avec l'algorithme d'Hasting-Métropolis. Elle consiste à chaque itération à adapter les paramètres θ selon une configuration des instants de ruptures simulée par un certain nombre d'itérations du MCMC. Cet algorithme, permettant d'approcher l'estimateur du maximum de vraisemblance des données observées, fournit une "bonne" estimation de θ.

Estimation par méthode de sélection de modèle

L'objectif de la seconde partie est l'estimation non-paramétrique de la fonction s par une méthode de sélection de modèle par pénalisation proposée par Birgé et Massart [7] dans le cadre des modèles gaussiens. Le principe est de construire un critère uniquement à partir des données qui permette d'estimer s avec le plus petit risque quadratique possible. C'est une situation dans laquelle il peut être préférable d'ignorer certaines ruptures correspondant à des sauts de moyennes très faibles.

Dans le chapitre 2, nous décrivons les trois étapes de la méthode. Tout d'abord, nous considérons la collection de toutes les partitions de la grille $\{1, \ldots, n\}$ notée $\mathcal{M}_n$. Nous associons à chaque partition m de dimension notée D_m de $\mathcal{M}_n$, un modèle $\mathcal{S}_m$ qui est le sous-espace linéaire des fonctions constantes par morceaux construites sur cette partition. Nous estimons ensuite la fonction s sur chacun des modèles en minimisant le critère empirique des moindres carrés noté γ_n défini pour u par :

$$\gamma_n(u) = \frac{1}{n}\sum_{t=1}^{n}(y_t - u(t))^2.$$

Nous obtenons ainsi une collection d'estimateurs $\{\hat{s}_m, m \in \mathcal{M}_n\}$. Finalement le meilleur estimateur est $\tilde{s} = \hat{s}_{\hat{m}}$ où $\hat{m}$ est la meilleure partition obtenue par minimisation d'un critère des moindres carrés pénalisé.

Yao [66], Miao *et. al* [51], Lavielle et Moulines [42] proposent différentes pénalités et obtiennent des résultats de consistance de l'estimateur pénalisé correspondant. Les propriétés sont asymptotiques, c'est-à-dire pour une taille d´échantillon qui tend vers l'infini. Mais face à des données réelles, le choix d'une bonne pénalité nécessite des résultats non-asymptotiques, c'est-à-dire pour une taille d'échantillon finie.

À l'aide d'un résultat établi par Birgé et Massart [7] sur les processus gaussiens, nous obtenons que pour une partition m de $\mathcal{M}_n$, une pénalité de la forme

$$pen_n(m) = \frac{\sigma^2}{n} D_m \left(c_1 \log\left(\frac{n}{D_m}\right) + c_2 \right),$$

fournit une borne non-asymptotique du risque quadratique pour l'estimateur $\mathbb{E}_s[\|\tilde{s}-s\|_n^2] = \mathbb{E}_s[1/n \sum_{t=1}^n (s(t) - \tilde{s}(t))^2]$:

$$\mathbb{E}_s\left[\|\tilde{s} - s\|_n^2\right] \leqslant C(c_1, c_2) \inf_{m \in \mathcal{M}_n} \left[\|s - s_m\|_n^2 + pen_n(m)\right] + C'(c_1, c_2)\frac{\sigma^2}{n}.$$

Nous en déduisons que

$$\mathbb{E}_s\left[\|\tilde{s} - s\|_n^2\right] \leqslant C \log(n) \inf_{m \in \mathcal{M}_n} \mathbb{E}_s\left[\|\hat{s}_m - s\|_n^2\right].$$

Donc l'estimateur pénalisé $\tilde{s}$ fait aussi bien à un $\log n$ près que le meilleur des estimateurs de la collections $\{\hat{s}_m, m \in \mathcal{M}_n\}$ en terme de risque, comme si on connaissait s.

Le terme en $\log(n/D_m)$ dans la pénalité (et donc dans cette majoration), inhabituel dans les critères classiques tels que C_p de Mallows [47], provient de la richesse de la collection $\mathcal{M}_n$. La pénalité ne dépend de la partition m que par sa dimension : le problème de sélection de partitions peut se voir comme un problème de sélection de dimensions et $\tilde{s} = \hat{s}_{\hat{m}(\hat{D})}$.

Les valeurs optimales des constantes c_1 et c_2 de la pénalité ne sont pas accessibles théoriquement. De plus, la variance est inconnue en pratique. Nous procédons en deux étapes pour obtenir la fonction de pénalité pen_n.

Dans le chapitre 3, nous supposons que la variance du bruit est connue et plutôt que d'avoir à choisir les constantes c_1 et c_2, nous proposons de déterminer leurs valeurs optimales : nous choisissons les valeurs c_1 et c_2 qui minimisent, uniformément en s et n, le rapport du risque de l'estimateur associé $\tilde{s}(c_1, c_2)$ à l'infimum des risque des meilleurs estimateurs de dimensions D, $\{\hat{s}_D = \hat{s}_{\hat{m}(D)}, D = 1, \ldots, n\}$, sur toutes les dimensions D :

$$\frac{\mathbb{E}_s[\|\tilde{s}(c_1, c_2) - s\|_n^2]}{\inf_{D=1,\ldots,n} \mathbb{E}_s[\|\hat{s}_D - s\|_n^2]}$$

Nous estimons ce rapport par Monte-Carlo pour une collection de fonctions s et une collection de taille d'échantillon n. Pour une dimension D fixée, rechercher la meilleure partition $\hat{m}(D)$ de dimension D parmi la collection $\mathcal{M}_n$, prend environ $\mathcal{O}(n^D)$ opérations. Pour réduire le temps de programmation, nous employons un algorithme dynamique, de complexité $\mathcal{O}(n^2)$.

Dans le chapitre 4, nous supposons que la variance est inconnue. Plutôt que d'utiliser un estimateur de la variance, nous considérons la variance comme une constante et nous

utilisons une méthode heuristique proposée par Birgé et Massart [8] qui consiste à trouver la pénalité à partir des données. La pénalité a maintenant la forme générale suivante :

$$pen(D) = \beta f_n(D) \quad \text{pour tout } D \geq 1,$$

où f_n est une fonction bien définie. Pour estimer β, nous mettons en oeuvre la méthode heuristique dont le principe est le suivant : le critère des moindres carrés $\gamma_n(\hat{s}_D)$ est une fonction affine en $f_n(D)$ et la pente estimée est l'opposé de la moitié de la constante β de la pénalité. Une étude par simulations montre que la qualité de la pénalité estimée peut dépendre du choix des dimensions sur lesquelles estimer la pente. Nous utilisons alors une extension de cette méthode proposée par Birgé et Massart [8] qui mène à définir des dimensions "particulières" sur lesquelles estimer cette pente. Nous avons calibré cette méthode et l'avons rendu complètement automatique. Nous avons réalisé plusieurs études de simulations pour étudier sa performance en terme de risque. Tout d'abord, nous avons comparé les résultats avec ceux obtenus quand la procédure est appliquée à variance connue et à variance inconnue (la variance est substituée dans la pénalité par l'estimateur proposé par Hall *et. al* [32]). Les résultats sont satisfaisants en terme de risque et encourage son utilisation sur d'autres problèmes. Des simulations sont ensuite proposées pour montrer le comportement de la méthode face à des cas qui ne vérifient pas les hypothèses : la fonction s n'est pas constante par morceaux et le bruit n'est pas gaussien. Nous discutons enfin des cas où la méthode calibrée ne permet pas de sélectionner la "bonne" partition et nous proposons une méthode non-calibrée qui permet à l'utilisateur le choix éventuel de plusieurs partitions. Nous avons appliqué la méthode non-calibrée pour identifier des changements dans le comportement des français face au virus HIV : nous segmentons la période entre le mois de Février 1987 et le mois d'Octobre 1991 selon le nombre de tests HIV effectués chaque mois en France.

La méthode d'estimation ne peut être appliquée que pour des signaux de taille modérée ($n \leqslant 5000$). Dans le chapitre 5 qui présente un travail en collabaration avec Servane Gey[1], nous considérons le cas où le signal est de très grande taille. Nous proposons un algorithme hybride combinant l'algorithme CART introduit par Breiman *et al.* [13] et un algorithme de recherche exhaustive qui consiste à considérer toutes les partitions possibles d'une grille donnée. L'algorithme CART, de complexité $\mathcal{O}(n \log n)$, est utilisé ici comme un préselectionneur de partitions "pertinentes". Nous avons réalisé une étude de simulation pour comparer les performances des différents algorithmes en terme de risque et en temps de calculs. Cette étude montre que l'algorithme hybride permet d'obtenir un estimateur proche de l'optimal en terme de risque en un temps relativement court.

1. Université Paris Sud, France

Détection de ruptures dans la distribution marginale d'une suite de variables aléatoires discrètes

Ce travail a été réalisé en collaboration avec Élodie Nédélec[2].

Nous considérons une suite $(Y_t)_{1\leqslant t\leqslant n}$ de n variables aléatoires indépendantes discrètes où Y_t prend un nombre fini r de valeurs avec $r \geq 2$. Des changements brusques affectent la distribution de la suite : les variables aléatoires ont la même distribution sur chaque segment d'une partition notée m_0 de $\{1, \ldots, n\}$ et ont des distributions différentes d'un segment à l'autre. Nous relions la distribution des variables à une fonction notée s définie par :

$$s = \sum_{I \in m_0} \sum_{i=1}^{r} s_I(i) \; \mathbb{1}_{I \times i},$$

où

$$P(Y_t = i) = s(t,i) = s_I(i) \qquad \text{pour tout } t \in I \text{ et } i = 1, \ldots, r.$$

Ce modèle signifie que pour chaque segment $I \in m_0$, $Y_I = (Y_t)_{t \in I}$ est une suite de variables aléatoires indépendantes et identiquement distribuées de loi commune s_I. L'objectif est l'estimation de la fonction s par la procédure de sélection de modèle présentée dans la partie précédente. Nous considérons une collection $\mathcal{F}_n$ suffisamment riche de partitions de la grille $\{1, \ldots, n\}$. L'estimateur de s est obtenu par minimisation de l'opposé de la log-vraisemblance pénalisée définie pour une partition $m \in \mathcal{F}_n$ par :

$$-\sum_{J \in m} \sum_{i=1}^{r} N_J(i) \log\left(\hat{s}_J(i)\right) + pen(m),$$

où

$$N_J(i) = \sum_{t \in J} \mathbb{1}_{\{Y_t = i\}},$$

est le nombre de i sur le segment J.

Nous établissons que si la pénalité vérifie pour une partition m de dimension D_m de $\mathcal{F}_n$

$$pen(m) = rD_m \left(K_1 \log\left(\frac{n}{D_m}\right) + K_2 \right)$$

alors nous disposons d'une majoration du risque de l'estimateur pénalisé correspondant :

$$E_s[\, l(s, \tilde{s})] \leqslant \inf_{m \in \mathcal{F}_n} \left\{ C \inf_{u \in \mathcal{S}_m} l(s,u) + (C+1) pen(m) \right\} + C'(r, \rho, C),$$

2. Université Paris Sud, France

étant donnée $C > 1$ et où,

$$l(s,u) = \sum_{I \in m_0} \sum_{J \in m} |I \cap J| \sum_{i=1}^{r} s_I(i) \log\left(\frac{s_I(i)}{u_J(i)}\right).$$

Pour obtenir ce résultat, il est nécessaire de supposer que les distributions s_I sur chaque segment I de la partition m_0 sont strictement positives et que la taille du plus petit segment de toutes les partitions de $\mathcal{F}_n$ n'est pas trop petite. Nous imposons les hypothèse suivantes :

- Nous supposons qu'il existe une constante strictement positive ρ telle que pour tout $I \in m_0$ et tout $i \in \{1, \ldots, r\}$

$$s_I(i) \geq \rho.$$

- Nous prenons $\mathcal{F}_n$ comme étant l'ensemble des partitions de $\{1, .., n\}$ construites sur une partition "dite la plus fine" et notée m_f telle que la taille Γ_{m_f} du plus petit segment de m_f vérifie :

$$\Gamma_{m_f} \geq \Gamma \log^2(n)$$

pour Γ une constante strictement positive.

La restriction sur la fonction s permet d'obtenir un contrôle du risque de l'estimateur pénalisé mais a pour conséquence que les constantes K_1 et K_2 de la pénalité dépendent de cette restriction ρ.

La preuve du résultat suit la même démarche que celle suivie par Birgé et Massart [48] dans le cadre de fonctions de contraste bornés et s'appuie sur une inégalité de concentration de type Talagrand dûe à Massart [48] qui permet de contrôler des termes intervenant dans la fonction de risque.

Nous proposons ensuite de nous écarter de l'hypothèse d'indépendance des Y_t en considérant que les Y_t suivent la loi d'une chaîne de Markov d'ordre 1 sur chaque segment de la partition m_0 : nous cherchons à détecter des ruptures dans les probabilités de transition. Nous obtenons un résultat similaire.

Application pour la détection de régions homogènes dans une séquence d'ADN

Une séquence d'ADN est constituée par un enchaînement de bases codées par un alphabet à quatre lettres $\mathcal{Y} = \{A, C, G, T\}$. Elle est formée de régions aux fonctions biologiquement déterminées, comme par exemple un gène, une zone codante, etc... À l'intérieur de ces régions dites homogènes, il existe une stabilité en fréquence des différentes bases. D'un

point de vue statistique, il s'agit de détecter des changements dans la distribution des différentes bases. La séquence d'ADN de longueur n est représentée par une suite $Y_1, \ldots, Y_n$ de n variables aléatoires à valeurs dans l'ensemble fini d'entiers $\{1, \ldots, 4\}$ représentant l'alphabet fini $\{A, C, G, T\}$. La méthode présentée dans le chapitre 6 permet d'estimer la loi de $Y = (Y_1, \ldots, Y_n)$, donc d'obtenir une segmentation de la séquence et les régions ainsi délimitées sont caractérisées d'un point de vue statitique par l'estimation des distributions sur chacunes des régions. En pratique, la collection de partitions $\mathcal{F}_n$ est l'ensemble de toutes les partitions possibles de la grille $\{1, \ldots, n\}$. D'un point de vue algorithmique, il s'agit de prendre en compte la longueur des séquences qui peut atteindre plusieurs millions de bases. Nous adaptons l'algorithme hybride proposé dans le chapitre 5. Pour la seconde étape de cet algorithme, par analogie à l'estimation de densité par histogrammes, la pénalité est choisie égale à

$$pen(m) = \alpha D_m \left(\log\left(n/D_m\right) + 2.5\right).$$

Nous avons appliqué la procédure sur deux séquence d'ADN de la bactérie B.subtilis et du bactériophage Lambda en considérant les deux modèles : le modèle markovien et le modèle indépendant. Les résultat obtenus sont similaires et montrent que sur ces deux exemples, le modèle markovien permet de détecter des ruptures dans la loi des $(Y_t)_{t=1,\ldots,n}$, et qu'il n'existe pas de ruptures dans les probabilités de transition. L'étude de la bactérie B.subtilis met en évidence des régions composés de gènes particuliers qui sont des gènes codants pour l'ARN ribosomiques.

Part I

An application of MCMC methods for the multiple change-points problem

Chapitre 1

An application of MCMC methods for the multiple change-points problem

M. Lavielle [1] and E. Lebarbier

Cet article est paru dans le journal Signal processing en 2000.

Abstract

We present in this paper a multiple change-point analysis for which a MCMC sampler plays a fundamental role. It is used for estimating the posterior distribution of the unknown sequence of change-points instants, and also for estimating the hyperparameters of the model. Furthermore, a slight modification of the algorithm allows one to compute the change-points sequences of highest probabilities. The so-called reversible jump algorithm is not necessary in this framework, and a very much simpler and faster procedure of simulation is proposed. We show that different interesting statistics can be derived from the posterior distribution. Indeed, MCMC is powerful for simulating joint distributions, and its use should not be restricted to the estimation of marginal posterior distributions, or posterior means.

Keyword : CHANGE-POINT DETECTION – GIBBS SAMPLER – HASTINGS-METROPOLIS

[1] Université Paris-V and Université Paris Sud, France

ALGORITHM – REVERSIBLE JUMP – SAEM ALGORITHM

Résumé

Nous nous interessons dans ce papier à un problème de détection de ruptures multiples, pour lequel un algorithme MCMC joue un rôle fondamental. Il est utilisé pour estimer la distribution a posteriori de la suite inconnue des instants de rupture, et également pour estimer les hyper-paramètres du modèle. De plus, une très légère modification de cet algorithme permet de déterminer les configurations de ruptures de plus haute probabilité. Le reversible jump n'est pas utile dans ce contexte, tandis qu'un algorithme beaucoup plus simple et plus rapide est proposé. Différentes statistiques interessantes peuvent être tirées de la distribution a posteriori. En effet, MCMC est une méthode puissante pour simuler des distributions jointes, et son utilisation ne devrait pas être réduite à l'estimation des marginales ou de moyennes a posteirori.

Mots clés: DÉTECTION DE RUPTURES – GIBBS SAMPLER – ALGORITHME D'HASTINGS-METROPOLIS – REVERSIBLE JUMP – ALGORITHME SAEM

1.1 Introduction

The subject of change-points analysis has been important in statistics for many years. This significant activity is largely motivated by the big amount of applications in signal processing (EEG, EMG and ECG analysis, geophysics, etc.) [4, 10, 39, 63], and many theoretical results have been obtained in various contexts, see for example [4, 14, 21, 42].

Among the different approaches, we can mention the on-line (or sequential) detection of change-points. In an off-line context, the unknown sequence of change-points instants can be estimated by minimizing a well-suitable contrast function, see [39]. We shall adopt here a Bayesian approach. Then, the change-point problem consists mainly in estimating the *posterior* distribution of the change-points sequence. That allows, for example, to estimate the probability that a change has occured at a given instant t. The posterior distribution of the number of changes can also be derived. The MAP (Maximum a Posteriori) estimator is obtained by maximizing this posterior distribution. When the changes affect the mean of the signal, we show that the MAP estimator is a penalized least-squares estimator, that possesses good statistical properties [42].

An MCMC method is really suitable for estimating the posterior distribution of the change-points sequence. The Reversible Jump algorithm proposed by Green [31], is based

on the fact that the dimension of the model can change, according to the number of segments. Unfortunately, this algorithm converges slowly, and many iterations are needed for estimating correctly the posterior distribution.

Another parametrization is shown to be more appropriate than the sequence (τ_k) of change points. In consists in introducing a sequence (r_t) that takes the value 1 at the change-points instants, and 0 between two jumps. The advantage of this parametrization is that the dimension of the sequence r is fixed. When the length of the observed signal is n, the Hastings-Metropolis method proposed in this paper simply consists in sampling sequences of 0 and 1, of fixed length $n-1$. Furthermore, running this algorithm at a *low temperature* allows to estimate the most likely configurations of changes. An hybrid MCMC algorithm, combining the basic Hastings-Metropolis algorithm with the Gibbs sampler [6, 27], can be used for estimating also the distribution of the mean sequence. Nevertheless, we show that the posterior expectation of the mean is not appropriate in this context, since it yields a smooth version of the signal, instead of a step function. The distribution of the mean sequence, conditionally to the most likely configuration of change-points, has more sense, and provides a good estimation. At the end, we show that another slight modification of our MCMC algorithm allows to estimate the hyper-parameters of the model. The Stochastic Approximation of Expectation Maximization (SAEM) procedure proposed by Delyon *et al.* [22] merely consists in updating the set of hyperparameters at each iteration of MCMC. This algorithm converges to a maxima of the likelihood, and provides automatically a "good" *prior* distribution for the unknown sequences.

The paper is organized as follows : Section 2 describes the model of change-points in the mean and details the prior modelling. The Hastings-Metropolis samplers used for estimating the posterior distribution of the change-points instants are described in Section 3. Section 4 addresses the problem of recovering also the sequence of means, and the Reversible Jump algorithm is presented. Section 5 is dedicated to the SAEM algorithm, for the estimation of the hyper-parameters of the model.

1.2 Model and notations

Let $y = (y_t,\ t \geq 1)$, be a real process such that, for any $t \geq 1$,

$$y_t = s(t) + \varepsilon_t \tag{1.2.1}$$

where $(\varepsilon_t,\ t \geq 1)$ is a sequence of zero-mean random variables. Here, the function s to recover is assumed to be piecewise constant. Thus, there exists some instants $(\tau_k,\ k \geq 0)$, such that the function s is constant between two successive change-points instants. In other words, there exists a sequence $(m_k,\ k \geq 1)$ such that, for any $k \geq 1$,

$$s(t) = m_k \ \text{ for all } \tau_{k-1} + 1 \leqslant t \leqslant \tau_k \tag{1.2.2}$$

with the convention $\tau_0 = 0$.

As already suggested by Lavielle [39] or Tourneret *et al.* [62, 63], it is convenient to introduce a change-points process $(r_t,\ t \geq 1)$ that takes the value 1 at the change instants and is zero between two changes:

$$r_t = \begin{cases} 1 & \text{if there exists } k \text{ such that } t = \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{1.2.3}$$

The estimation of the change-points instants reduces to the estimation of the sequence (r_t). Then, the unknown function s will be recovered by estimating the sequences (r_t) and (m_k).

To solve this inverse problem, we shall adopt a Bayesian approach. That means that we have to define the distribution of the non observed sequences, conditionally to the set of observations. This distribution is usually called the *posterior* distribution and requires to define first the *prior* distribution of (r_t) and (m_k).

Assume that the observed sequence (y_t) is available between instants $t = 1$ and $t = n$. First, we consider that (r_t) is a sequence of independent and identically distributed (*i.i.d.*) Bernoulli random variables with parameter λ. Then, for any $r = (r_t,\ 1 \leqslant t \leqslant n-1)$ in $\Omega = \{0,1\}^{n-1}$,

$$\pi(r;\lambda) = \lambda^{\sum_{t=1}^{n-1} r_t}\ (1-\lambda)^{n-1-\sum_{t=1}^{n-1} r_t}. \tag{1.2.4}$$

On the other hand, $(s(t),\ 1 \leqslant t \leqslant n)$ is modeled as a sequence of *i.i.d.* Gaussian random variables with mean μ and variance V. Then,

$$\pi(s(1),\ldots,s(n);\mu,V) = \prod_{t=1}^{n} (2\pi V)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2V}(s(t)-\mu)^2\right\}. \tag{1.2.5}$$

For a given configuration of change-points r, $\sum_{t=1}^{n-1} r_t$ is the number of change-points. Then, let $K_r = \sum_{t=1}^{n-1} r_t + 1$ be the number of segments, $n_k = \tau_k - \tau_{k-1}$ be the length of segment k and $m = (m_k,\ 1 \leqslant k \leqslant K_r)$ be the vector of means. Then,

$$\begin{aligned} \pi(m|r;\mu,V) &= \pi(m_1,\ldots,m_{K_r}|r;\mu,V) \\ &= \pi(s(1),\ldots,s(n)\ ,\ s(t) = m_k,\ \tau_{k-1}+1 \leqslant t \leqslant \tau_k,\ 1 \leqslant k \leqslant K_r\ ;\mu,V) \\ &= \prod_{k=1}^{K_r} \left(\frac{2\pi V}{n_k}\right)^{-\frac{1}{2}} \exp\left\{-\frac{n_k}{2V}(m_k-\mu)^2\right\}. \end{aligned} \tag{1.2.6}$$

Thus, the m_k's are independent, and m_k is Gaussian with mean μ and variance V/n_k.

On the other hand, $(\varepsilon_t,\ t \geq 1)$ is assumed to be a sequence of independent Gaussian random variables with mean 0 and variance σ^2. Thus, the conditional distribution of the observations is defined by :

$$h(y|r,m;\sigma^2) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{k=1}^{K_r}\sum_{t=\tau_{k-1}+1}^{\tau_k}(y_t-m_k)^2\right\}, \tag{1.2.7}$$

Let $\theta = (\mu, \lambda, V, \sigma^2)$ be the set of hyper-parameters of the model. Then, the prior distribution of (r, m) is given by

$$\pi(r, m; \theta) = \pi(m|r; \mu, V)\pi(r; \lambda), \tag{1.2.8}$$

the complete likelihood of (y, r, m) is

$$f(y, r, m; \theta) = h(y|r, m; \sigma^2)\pi(m|r; \mu, V)\pi(r; \lambda), \tag{1.2.9}$$

and the posterior distribution of (r, m) can be decomposed as

$$p(r, m|y; \theta) = p(r|y; \theta)p(m|y, r; \theta). \tag{1.2.10}$$

For a given value of r, the conditional distribution of m is easy to compute. Indeed, let $\bar{y}_k = n_k^{-1} \sum_{t=\tau_{k-1}+1}^{\tau_k} y_t$ be the empirical mean of y in segment k. Then, equations (1.2.6) and (1.2.7) yield

$$p(m|y, r; \theta) = \prod_{k=1}^{K_r} (2\pi V_k)^{-\frac{1}{2}} \exp\{-\frac{1}{2V_k}(m_k - \mu_k)^2\} \tag{1.2.11}$$

where

$$V_k = \frac{V\sigma^2}{n_k(V + \sigma^2)} \tag{1.2.12}$$

and

$$\mu_k = \frac{V\sigma^2}{V + \sigma^2}\left(\frac{\bar{y}_k}{\sigma^2} + \frac{\mu}{V}\right). \tag{1.2.13}$$

Thus, conditionally to the observations, the m_k's remain independent and Gaussian (a short demonstration of these formulae is given in the Appendix A).

The following Lemma gives the posterior distribution of r:

Lemma 1.2.1. *For any configuration of change-points r, let K_r be the number of segments and let $S_r = \sum_{k=1}^{K_r} \sum_{t=\tau_{k-1}+1}^{\tau_k} (y_t - \bar{y}_k)^2$. Then, the posterior distribution of r is defined by :*

$$p(r|y; \theta) = C(y, \theta) \exp\{-\phi S_r - \gamma K_r\} \tag{1.2.14}$$

where

$$\phi = \frac{V}{2\sigma^2(\sigma^2 + V)} \quad , \quad \gamma = \frac{1}{2}\log\left(\frac{\sigma^2 + V}{\sigma^2}\right) + \log\left(\frac{1 - \lambda}{\lambda}\right).$$

and where $C(y, \theta)$ is a normalizing constant.

(The proof of the Lemma is in the Appendix A)

Remarks:

1. It is important to insist on the fact that the posterior distribution $p(r|y;\theta)$ is the joint distribution of a vector of size $n-1$. Thus, it cannot be used as it stands and should be summarized to some characteristics. Between many others, we can consider the following characteristics:

   - For any $1 \leqslant t \leqslant n-1$, the marginal posterior distribution $p(r_t|y;\theta)$ gives the probability to have a change-point at instant t, conditionally to the observations.
   - The MAP (Maximum a Posteriori) estimator is the particular value of r that maximizes $p(r|y;\theta)$. In other words, it is the most likely configuration of change-points, according to the prior and to the observations.
   - For any instants τ_a and τ_b, $\mathbb{P}\left(\sum_{t=\tau_a}^{\tau_b} r_t = k|y;\theta\right)$ is the probability to have exactly k change-points between these two instants. In particular, when $\tau_a = 1$ and $\tau_b = n-1$, we consider the posterior distribution of the total number of change-points.

2. The posterior distribution of r can be written

   $$p(r|y;\theta) = C(y,\theta)\ \exp\{-U_\theta(y,r)\} \tag{1.2.15}$$

   where $U_\theta(y,r) = \phi S_r + \gamma K_r$ is a penalized contrast, usually called energy function, and which is the sum of two terms : the first term S_r, measures the fidelity to the observations y while the second term K_r corresponds to a penalization term, related to the number of change-points. The coefficients ϕ and γ indicate the relative weights given to these two criteria. A small value of γ in front of ϕ favor configurations with a big number of change-points, while a big value of γ penalizes such configurations. The MAP estimator of r minimizes the energy function $U_\theta(y,r)$. In this particular example, the MAP estimator reduces to a penalized least-squares estimate. We can mention that theoretical results concerning this estimator have been obtained by Lavielle and Moulines [42] under very general conditions.

3. Unfortunately, the normalizing constant $C(y,\theta)$ in (1.2.14) and (1.2.15) cannot be computed, since it is the sum over all the possible configurations r of $\exp\{-U_\theta(y,r)\}$, that is, a sum of 2^{n-1} terms. In other words, the posterior distribution of r is known up to this constant and a Monte-Carlo Markov Chain method should be used to sample it and estimate some of its characteristics.

1.3 The Hastings-Metropolis algorithm

1.3.1 The basic algorithm

The main idea of this algorithm is to generate an ergodic Markov chain $(r^{(i)},\ i \geq 0)$ so that $p(\cdot\,|y;\theta)$ is its stationary distribution. Then, the ergodic Theorem implies that, for any measurable function f,

$$\bar{f}_N = \frac{1}{N}\sum_{i=1}^{N} f(r^{(i)}) \tag{1.3.16}$$

is a strongly consistent estimator of $\mathbb{E}(f(r)|y;\theta)$, *i.e.* $\bar{f}_N$ converges almost surely to $\mathbb{E}(f(r)|y;\theta)$ when $N \to \infty$, see [50] for example.

An interesting application of this result is the estimation of probabilities of specified events, when f is an indicator function. For example, the marginal posterior distributions of the r_t's and the a posterior distribution of the number of segment K_r, can easily be estimated. Indeed, for any $1 \leqslant t \leqslant n-1$ and any $k \geq 0$,

$$\frac{1}{N}\sum_{i=1}^{N} r_t^{(i)} \;\to\; \mathbb{P}(r_t = 1|y;\theta) \quad \text{a.s.} \tag{1.3.17}$$

$$\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{(K_r^{(i)}=k)} \;\to\; \mathbb{P}(K_r = k|y;\theta) \quad \text{a.s.} \tag{1.3.18}$$

where $K_r^{(i)} = \sum_{t=1}^{n-1} r_t^{(i)} + 1$ is the number of segments in the configuration $r^{(i)}$.

The Hastings-Metropolis algorithm is an iterative procedure. At iteration i, we carry out the following two steps:

- an admissible new value $\tilde{r}$ is drawn from a *proposal kernel* $q(r^{(i)}, \tilde{r})$
- $\tilde{r}$ is accepted as the new state, *i.e.* $r^{(i+1)} = \tilde{r}$, with the following probability :

$$\alpha(r^{(i)}, \tilde{r}) = \min\left\{1, \frac{p(\tilde{r}|y;\theta)}{p(r^{(i)}|y;\theta)}\frac{q(\tilde{r}, r^{(i)})}{q(r^{(i)}, \tilde{r})}\right\} \tag{1.3.19}$$

Remarks:

1. If the kernel q is irreducible, then the Markov Chain $(r^{(i)})$ is irreducible. Furthermore, the aperiodicity of the chain is ensured if there exists two configurations (r, r') such that $\alpha(r, r') < 1$. Under these conditions, the chain $(r^{(i)})$ is uniformly ergodic, since it takes its values in a finite space.

2. An initial *burn-in* period is introduced before collecting samples, so that the estimation weakly depends on the initial guess (see [58]). If N_b is the length of this burn-in period, then, the estimator of $\mathbb{E}(f(r)|r;\theta)$ proposed in (1.3.16) is replaced by

$$\bar{f}_N = \frac{1}{N}\sum_{i=N_b+1}^{N_b+N} f(r^{(i)}) \tag{1.3.20}$$

3. For any (r, r'), let $\Delta U(r, r') = U_\theta(y, r') - U_\theta(y, r)$. Then, (1.2.15) yields

$$\frac{p(r'|y;\theta)}{p(r|y;\theta)} = e^{-\Delta U(r,r')}. \tag{1.3.21}$$

Since the energy $U_\theta(y, r)$ is a sum of local potentials, a local perturbation of the current state $r^{(i)}$ will affect few terms of this sum and the probability of acceptance $\alpha(r^{(i)}, \tilde{r})$ will be easy to compute.

1.3.2 The proposal kernels

As it is mentioned just above, any irreducible proposal kernel q can be used. From a practical point of view, it is important to allow more communications between the states of high probability in order to increase the convergence speed. In our example of application, that can be done by using successively the following three kernels at each iteration:

1. q_1 is such that the candidate $\tilde{r}$ is drawn independently of the current state r: $q_1(r, \tilde{r}) = \pi(\tilde{r};\theta)$. Let $\beta = 1/2\log((\sigma^2 + V)/\sigma^2)$. Then, we obtain

$$\alpha(r, \tilde{r}) = \min\{1, \exp\{-\phi(S_{\tilde{r}} - S_r) - \beta(K_{\tilde{r}} - K_r)\}\}. \tag{1.3.22}$$

2. q_2 is such that a new change-point is created or an existing change-point is removed. An instant s is chosen randomly in $\{1, , \ldots, n-1\}$ and we set $\tilde{r}_t = r_t$ for all $t \neq s$ while $\tilde{r}_s = 1 - r_s$. The acceptance probability turns out to be

$$\alpha(r, \tilde{r}) = \min\{1, \exp\{-\phi[S_{\tilde{r}} - S_r] \pm \gamma\}\}. \tag{1.3.23}$$

3. With the third considered kernel q_3, an existing change-point instant is moved. Two instants (s, s') are randomly chosen such that $r_s = 1$ and $r_{s'} = 0$. Then, $\tilde{r}_t = r_t$ for all $t \neq s, s'$ while $\tilde{r}_s = 0$ and $\tilde{r}_{s'} = 1$. In this case, the acceptance probability is

$$\alpha(r, \tilde{r}) = \min\{1, \exp\{-\phi(S_{\tilde{r}} - S_r)\}\}. \tag{1.3.24}$$

We propose an example to illustrate this algorithm. We simulate a sequence $y = (y_1, \ldots, y_n)$ with $n = 500$. There are four change-points at $\tau_1 = 75$, $\tau_2 = 150$, $\tau_3 = 250$, and $\tau_4 = 400$. The vector of mean is $m = (0.125, 0.5, 0.4, 0.5, 0.125)$. The variance of the

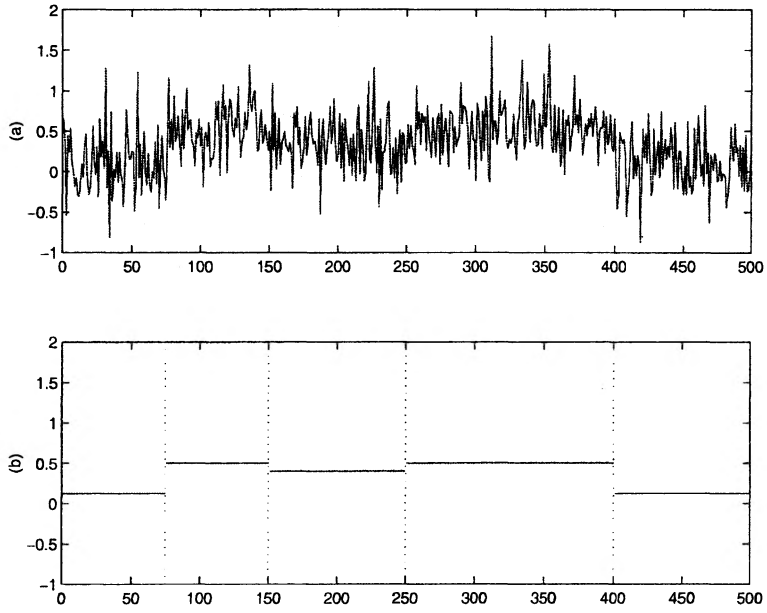


Figure 1.1: (a) The observed signal y, (b) the mean of y and the change-point instants.

additive noise is $\sigma^2 = 0.1$. The observed series and the mean are plotted in Figure 1-a and Figure 1-b.

First of all, because the set of hyper-parameters θ is unknown, it is estimated by using the SAEM procedure described in Section 1.5. Then, the estimated value $\widehat{\theta} = (\widehat{\lambda}, \widehat{\mu}, \widehat{V}, \widehat{\sigma^2}) = (0.012,\ 0,346,\ 2.688,\ 0.106)$ is used in the MCMC algorithm. We run the MCMC algorithm with 5 000 burn-in iterations. The estimations of the marginal posterior probabilities $\{\mathbb{P}(r_t = 1|y;\theta)\}$ obtained after 15 000 and 150 000 iterations are plotted in Figure 2. The posterior distribution of the number of segments K_r is displayed Figure 2-c (estimated after 150 000 iterations).

First of all, we can remark that the estimations obtained after 15 000 iterations are closed to those obtained after 150 000 iterations. That means that this algorithm converges quite quickly, and only "few iterations" are enough to detect very well the four change-points. Theoretical aspects concerning the convergence control of MCMC methods can be found in [58].

These diagrams can be seen as histograms around each change-points. For example, we obtain a very accurate estimation of the position of the first change-point (at 75) since the estimated posterior distribution of r is very spiky around this instant: the estimates of $\mathbb{P}(r_{75} = 1|y;\theta)$ and $\mathbb{P}(r_{76} = 1|y;\theta)$, obtained with 150 000 iterations, are respectively 0.42 and 0.49. On the other hand, the jumps of the mean are smaller at 150 and 250, and the detection of these two change-points is not so accurate: the estimated marginal probabilities are very small around 150 and 250 (around 0.1). Nevertheless, the probability

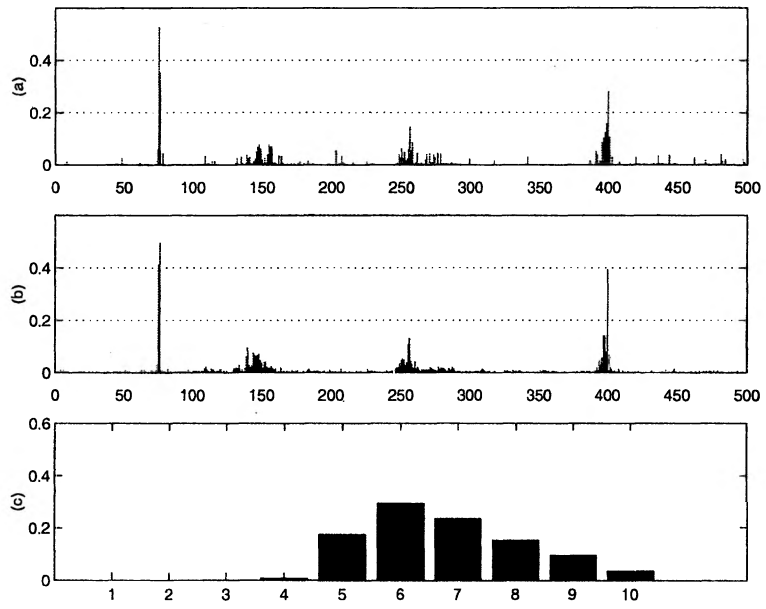


Figure 1.2: The posterior distribution of r estimated with the Hastings-Metropolis algorithm. The marginal distributions $\{\mathbb{P}(r_t = 1|y;\theta), 1 \leqslant t \leqslant n-1\}$ estimated with: (a) 15 000 iterations, (b) 150 000 iterations, (c) the posterior distribution of the number K_r of segments.

of a change-point is very high in a neighborhood of these two instants. For example, the estimated probability to have a change point in the interval [135, 165] (resp. [235, 265]) is 0.85 (resp. 0.77).

In other words, it is not convenient to apply directly a threshold on the sequence of estimated marginal probabilities $\{\mathbb{P}(r_t = 1|y;\theta)\}$, for detecting the change-points. A first solution consists in estimating the probability to have a change-point in an interval, instead of an instant. Of course, the positions of the change-points will not be precisely estimated with this method. To overcome this loss of accuracy, a second approach consists in estimating the configurations of change-points of higher probability. That can easily be done, by using a slight variation of the Hastings-Metropolis algorithm.

1.3.3 Running the Hastings-Metropolis algorithm at a low temperature

For any $T > 0$, we can consider the distribution $p_T(\cdot|y;\theta)$, based on the original distribution $p(\cdot|y;\theta)$, and defined as follows:

$$p_T(r|y;\theta) = C_T(y;\theta) \exp\left\{-\frac{U_\theta(y,r)}{T}\right\} \tag{1.3.25}$$

$$= C_T(y;\theta) \exp\left\{-\frac{\phi}{T}S_r - \frac{\gamma}{T}K_r\right\}. \tag{1.3.26}$$

The role of the parameter T (usually called temperature) is mainly to discriminate the global and the local maxima of the posterior distribution $p(\cdot|y;\theta)$. Indeed, any maximum (local or global) of $p(\cdot|y;\theta)$ is a minimum of $U_\theta(y,r)$, and also a maximum of $p_T(\cdot|y;\theta)$. However, $p_T(r|y;\theta) \to 0$ when $T \to 0$ if r is not a global maximum. Thus, when $T \to 0$, $p_T(\cdot|y;\theta)$ converges to the uniform distribution on the set of global maxima of $p(\cdot|y;\theta)$.

The Hastings-Metropolis algorithm described above can be used for simulating $p_T(\cdot|y;\theta)$. Indeed, from (1.3.26), the only modification to introduce in the algorithm is the replacement of the parameters (ϕ,γ) by $(\phi/T,\gamma/T)$.

The simulated annealing algorithm consists in using a sequence of temperatures (T_i) that decreases at each iteration. Then, the Markov Chain $(r^{(i)})$ is no more homogeneous, and converges to the global maxima of $p(\cdot|y;\theta)$ (the MAP estimator) if T_i behaves like $T_0/\log(i)$ (see [24, 27]). Unfortunately, this schedule of temperature cannot be used in the practice, since it would require a very large number of iterations.

In fact, there exists a very efficient and attractive alternative for detecting the most likely configuration of change-points. It consists merely in running the Hastings Metropolis algorithm at a fixed low temperature T.

We can observe the influence of a low temperature on the chain's construction during the algorithm: consider two states r and r' such that $U_\theta(y,r) < U_\theta(y,r')$. When T tends to 0, $\exp\{-(U_\theta(y,r') - U_\theta(y,r))/T\}$ tends to 0. So $\alpha(r,r')$ tends to 0 and a move from r to r' has a very low probability. Consequently, when T is a low temperature, the Hastings-Metropolis algorithm will favor the configurations of change-points of highest probabilities.

To set $T = 0$ leads to the so-called Iterative Conditional Modes (ICM) algorithm (see [39]). This deterministic procedure usually leads to a local minima of the posterior distribution of r.

On this other hand, for any $T > 0$, the Markov chain $(r^{(i)})$ simulated with this algorithm remains homogeneous and ergodic: its distribution converges to $p_T(\cdot|y;\theta)$.

Figure 3 shows the results obtained with three temperatures: $T = 0.5$, $T = 0.2$ and $T = 0.01$. Looking at these results, we can make two main remarks:

1. The false alarms are removed, and only the main events are left. Even a "high" temperature, such as $T = 0.5$, cleans the results. With $T = 0.5$, the estimated probability to have 5 segments is 0.97. This estimated probability is 1 for $T \leqslant 0.3$. That clearly shows that the most likely configurations are made up of five segments.

2. When the temperature decreases, the posterior distribution becomes more and more concentrated around the MAP estimator of the change-points instants. In this example, the MAP is $\widehat{\tau} = (\widehat{\tau_1}, \widehat{\tau_2}, \widehat{\tau_3}, \widehat{\tau_4}) = (76, 147, 256, 400)$.

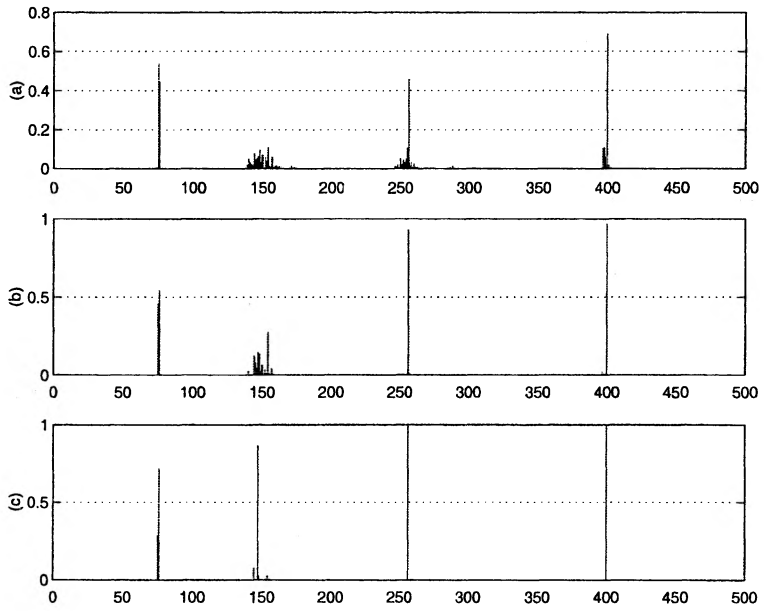


Figure 1.3: Running the Hastings-Metropolis algorithm at a low temperature: (a) $T = 0.5$, (b) $T = 0.2$, (c) $T = 0.01$.

1.4 The estimation of the mean

Assume now that we are interested in the joint posterior distribution of the mean sequence $m = (m_k)$ and the change-points instants $\tau = (\tau_k)$ (or $r = (r_t)$, according to the parametrization), instead of the posterior distribution of τ. One iteration of the Hastings-Metropolis algorithm will consist now in drawing a candidate $(\tilde{m}, \tilde{\tau})$ (or $(\tilde{m}, \tilde{r})$), with a new proposal b, and to accept it with a probability $\alpha((m^{(i)}, \tau^{(i)}), (\tilde{m}, \tilde{\tau}))$. Different approaches can be adopted for choosing a proposal b.

1.4.1 The Reversible Jump MCMC algorithm

A first approach is the so-called Reversible Jump algorithm, proposed by Green [31]. This method is an adaptation of MCMC algorithms, when the dimensionality of the parameter vector is not fixed. Then, the Markov chain can "jump" between models with parameter spaces of different dimensions. As before, different kernels should be used. For example, we can make use of .the four following moves:

1. One of the mean m_k is randomly changed. The proposed mean $\tilde{m}_k$ is such that $\log(\tilde{m}_k/m_k)$ is uniformly distributed on $[-1/2; 1/2]$, in order to avoid big jumps.

2. A change-point is added in segment k. The position $\tilde{\tau}_k$ is drawn uniformly in $[\tau_{k-1} +$

$1, \tau_k - 1]$. The mean m_k is split into two means $\tilde{m}_k$ and $\tilde{m}_{k+1}$ such that

$$(\tilde{\tau}_k - \tau_{k-1})\tilde{m}_k + (\tau_k - \tilde{\tau}_k)\tilde{m}_{k+1} = n_k m_k.$$

This condition is satisfied with

$$\tilde{m}_k = m_k - u\sqrt{\frac{\tilde{n}_{k+1}}{\tilde{n}_k}} \quad \text{and} \quad \tilde{m}_{k+1} = m_k + u\sqrt{\frac{\tilde{n}_k}{\tilde{n}_{k+1}}},$$

where u is uniformly distributed on the interval $[-0.2, 0.2]$, for the same reason as in the first move.

3. A change-point τ_k is removed. Then, the means m_k and m_{k+1} are replaced by a unique $\tilde{m}_k$ such that

$$(n_k + n_{k+1})\tilde{m}_k = n_k m_k + n_{k+1} m_{k+1},$$

where $n_k = \tau_k - \tau_{k-1}$ is the length of segment k.

4. A change-point τ_k is moved. A new position $\tilde{\tau}_k$ is drawn uniformly on $[\tau_{k-1}+1, \tau_{k+1}-1]$ and the means remain unchanged.

At each iteration, an independent random choice is made between these four move types. These have probabilities 0.3 for the moves 1 and 4, and 0.2 for two others.

Following Green [31], the probabilities of acceptance can be computed for these different kernels. The formulae are given in the Appendix B.

We applied this algorithm on the same series y displayed Figure 1-a. Figure 4 presents the estimation of the probabilities $\{\mathbb{P}(r_t = 1|y;\theta)\}$ after 15 000 and 150 000 iterations. Comparing Figure 4-a with Figure 2-a, we remark that the Reversible Jump algorithm converges much more slowly than the Hastings-Metropolis algorithm described in the previous section. Indeed, we are now simulating a pair of variables (m, r) instead of simulating only r. The introduction of a new (continuous) variable to sample slows down the algorithm.

One explanation is the fact that the Reversible Jump algorithm does not take use of the natural hierarchy

$$p(m, r|y;\theta) = p(r|y;\theta)p(m|r, y;\theta) \tag{1.4.27}$$

for its proposal kernels, and many candidate $(\tilde{m}, \tilde{r})$ are rejected. Then, a big amount of iterations are required for estimating correctly the posterior distribution of interest.

1.4.2 An hybrid algorithm

A second approach consists in combining the Hastings-Metropolis algorithm described in Section 1.3.1 for simulating r, with the Gibbs sampler [57] for simulating m.

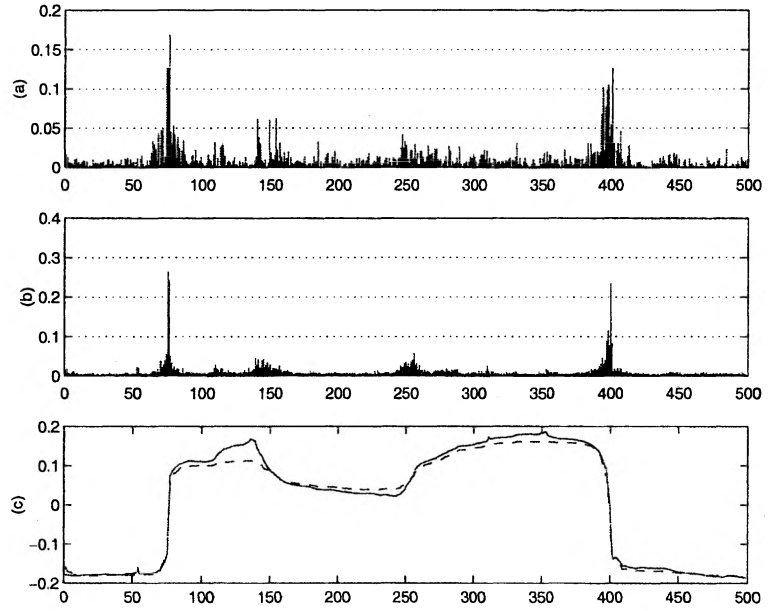


Figure 1.4: The posterior distributions of (r, m) estimated with the Reversible Jump algorithm. The marginal distributions $\{\mathbb{P}(r_t = 1|y;\theta), 1 \leqslant t \leqslant n-1\}$ estimated with: (a) 15 000 iterations, (b) 150 000 iteration, (c) —: the posterior mean of m, ––: the posterior mean of m, conditionnaly to $K_r = 5$.

The proposal kernels are defined by

$$b((m^{(i)}, \tau^{(i)}), (\tilde{m}, \tilde{\tau})) = q_i(\tau^{(i)}, \tilde{\tau}) p(\tilde{m}|\tilde{\tau}, y; \theta) \tag{1.4.28}$$

where q_i is one of the proposal kernels defined in Section 1.3.2. Then, the probability of acceptance is

$$\alpha(m^{(i)}, \tau^{(i)}, (\tilde{m}, \tilde{\tau})) = \min\left\{1, \frac{p(\tilde{m}, \tilde{\tau}|y;\theta)}{p(m^{(i)}, \tau^{(i)}|y;\theta)} \frac{b((\tilde{m}, \tilde{\tau}), (m^{(i)}, \tau^{(i)}))}{b((m^{(i)}, \tau^{(i)}), (\tilde{m}, \tilde{\tau}))}\right\} \tag{1.4.29}$$

$$= \min\left\{1, \frac{p(\tilde{\tau}|y;\theta)}{p(\tau^{(i)}|y;\theta)} \frac{q_i(\tilde{\tau}, \tau^{(i)})}{q_i(\tau^{(i)}, \tilde{\tau})}\right\}. \tag{1.4.30}$$

That means that the probability of acceptance does not depend on the mean vectors $m^{(i)}$ and $\tilde{m}$, but only on the configurations of changes $r^{(i)}$ and $\tilde{r}$. In other words, we use the Hastings-Metropolis algorithm described in Section 1.3.1 for generating the sequence $(r^{(i)})$, while $m^{(i)}$ is drawn at iteration i with the conditional distribution $p(m|r^{(i)}, y; \theta)$. This algorithm was shown to converge much more faster than the Reversible Jump algorithm, since a very good approximation of the marginal posterior probabilities $\{\mathbb{P}(r_t = 1|y;\theta)\}$ is obtained after only 15 000 iterations (see Figure 2-a).

1.4.3 What can we do with a joint distribution?

These algorithms produce ergodic Markov chains $(m^{(i)}, r^{(i)})$ that converge to the joint posterior distribution $p(m, r|y; \theta)$. Once more, this joint distribution cannot be described completely, but should be reduced to some interesting and tractable characteristics. Most of the times, MCMC is only used for estimating the posterior mean of the non observed variable. In our context, that would mean to estimate $\mathbb{E}(m_t|y; \theta)$, $1 \leqslant t \leqslant n$, by the empirical mean $N^{-1} \sum_{i=N_b+1}^{N_b+N} m_t^{(i)}$ (or eventually, by using the Rao-Blackell version described in [17, 57]). This estimated posterior mean is displayed Figure 4.

Unfortunately, this posterior mean is uninteresting in our context. Indeed, let $\Omega = \{0, 1\}^{n-1}$ be the set of possible configurations of change-points. Then, the marginal posterior distribution $p(m|y; \theta)$ is the sum of the joint posterior distributions $p(m, r|y; \theta)$, over all the possible configurations:

$$p(m|y; \theta) = \sum_{r \in \Omega} p(m, r|y; \theta) \tag{1.4.31}$$

and the posterior mean can also be decomposed as weighted sum of conditional means, over all the possible configurations:

$$\mathbb{E}(m|y; \theta) = \sum_{r \in \Omega} \mathbb{E}(m|r, y; \theta)\, p(r|y; \theta). \tag{1.4.32}$$

That means that we are mixing configurations without any change-points with configurations with one, two, ten or more change-points. Then, the meaning of this posterior mean is not obvious at all ...

Green proposes to estimate this posterior mean, conditionally to a given number of segments (or to a given number of change-points). Such an example is presented Figure 4, for four segments. The estimated mean remains smooth, since we are now integrating over all the configurations with four segments. Actually, this curve can be seen as a smooth version of the original data. It is a little bit embarrassing to obtain a smooth function, when we are looking for a step function ...

Another approach consists in estimating m, conditionally to a given configuration of change-points r. That is very easy, since the conditional distribution $p(m|r, y; \theta)$ is a Gaussian distribution with known parameters. For example, it seems natural to consider the most likely configuration of change-points, that is, the MAP estimate of r. Then, conditionally to this particular configuration, $(m_1, m_2, m_3, m_4, m_5)$ is Gaussian with mean $(0.106, 0.534, 0.338, 0.548, 0.114)$ and variance $(13, 14, 9, 7, 10) \times 10^{-3}$.

1.5 Estimation of θ using SAEM algorithm

The implementation of an MCMC algorithm as described above, assumes that the set of parameters of the model is known. Recall that these hyper-parameters are respectively the

prior proportion of change-points λ, the parameters μ and V of the Gaussian distribution for the vector of means m, and σ^2 the variance of the additive noise. Instead of setting the hyper-parameters θ to a particular value, as it is usually done in a Bayesian framework, we propose to estimate θ.

The maximum likelihood estimator (MLE) of θ maximizes the likelihood of the observed data $g(y;\theta)$. Unfortunately, the MLE cannot be computed in a close-form in a context of incomplete data. The SAEM is well suitable for computing the MLE in this kind of situation, see [15, 38] for some examples of application. This stochastic version of the EM (Expectation Maximization, [23]) algorithm just consists in updating the estimate of the hyper-parameters θ at each iteration of the MCMC algorithm described above. This update is based on a stochastic approximation of the exhaustive statistics of the complete data model (r, y). Thus, the first thing to do is to write the complete likelihood $f(r, y; \theta)$ in a standard exponential form. We have the following Lemma:

Lemma 1.5.1. *For any configuration of changes r, let $\bar{y}_k = n_k^{-1} \sum_{t=\tau_{k-1}+1}^{\tau_k} y_t$, $\bar{y} = n^{-1} \sum_{t=1}^{n} y_t$ and $S_r = \sum_{k=1}^{K_r} \sum_{t=\tau_{k-1}+1}^{\tau_k} (y_t - \bar{y}_k)^2$. Then, the likelihood of the complete data is defined by*

$$
\begin{aligned}
f(y, r; \theta) &= (2\pi\sigma^2)^{-\frac{n}{2}} \left(\frac{\sigma^2 + V}{\sigma^2}\right)^{-\frac{K_r}{2}} \lambda^{K_r - 1} (1-\lambda)^{n - K_r + 1} \\
&\quad \times \exp\left\{ -\frac{1}{2(V+\sigma^2)} \left(\sum_{t=1}^{n} (y_t - \mu)^2 + \frac{V}{\sigma^2} S_r \right) \right\} \qquad (1.5.33)
\end{aligned}
$$

and the maximum likelihood estimator of θ (i.e. the value of θ that maximizes the complete likelihood $f(y, r; \cdot)$) is ,for $K_r > 1$, $\tilde{\theta} = (\tilde{\mu}, \tilde{\lambda}, \tilde{\sigma}^2, \tilde{V})$, where

$$\tilde{\mu} = \bar{y}, \qquad (1.5.34)$$

$$\tilde{\lambda} = \frac{K_r - 1}{n - 1}, \qquad (1.5.35)$$

$$\tilde{\sigma}^2 = \frac{S_r}{n - K_r}, \qquad (1.5.36)$$

$$\tilde{V} = \frac{\sum_{t=1}^{n} (y_t - \bar{y})^2 - S_r}{K_r} - \tilde{\sigma}^2. \qquad (1.5.37)$$

(The proof of this Lemma is in the Appendix A)

Remarks:

1. The maximum likelihood of μ is $\bar{y}$ and therefore does not depend on the non observed data r. Thus, $\bar{y}$ is also the value of μ that maximizes the observed likelihood $g(y;\theta)$. The SAEM algorithm will be used for estimating the others parameters λ, σ^2 and V.

2. Just like in an ANalysis Of VAriance (ANOVA) context, the maximum likelihood of σ^2 is the empirical residual variance $(n-K_r)^{-1}\sum_{k=1}^{K_r}\sum_{t=\tau_{k-1}+1}^{\tau_k}(y_t-\bar{y}_k)^2$, that is, the total sum of residual squares S_r divided by the degrees of freedom $n-K_r$ (see [55] p 185-186).

3. Considering the observed series y as a constant, the maximum likelihood estimator of (λ,σ^2,V) only depends on the missing sequence r via the two statistics K_r and S_r. As we shall see just below, the SAEM algorithm is based on this remark.

The proposed SAEM algorithm is an iterative algorithm that requires an initial configuration of change-points $r^{(0)}$ and an initial guess $\theta^{(0)}$. Then, at iteration i, a simulation step and an estimation step are performed as follows:

- Simulation step: a new configuration $r^{(i)}$ is generated with M iterations of the MCMC algorithm, using the current values of the hyper-parameters $\theta^{(i-1)}$ and the current configuration $r^{(i-1)}$.
- Estimation step: $\theta^{(i)}$ is updated by using the new configuration $r^{(i)}$, and according to the two following steps :
  1. Stochastic Approximation: update the approximation of the sufficient statistics as follows:

$$s_1^{(i)} = s_1^{(i-1)} + a_i(K_{r^{(i)}} - s_1^{(i-1)}) \tag{1.5.38}$$

$$s_2^{(i)} = s_2^{(i-1)} + a_i(S_{r^{(i)}} - s_2^{(i-1)}) \tag{1.5.39}$$

  where $K_{r^{(i)}}$ and $S_{r^{(i)}}$ are the sufficient statistics of the complete model, computed at the point $(y,r^{(i)})$ and where (a_i) is a sequence of decreasing stepsizes such that $\sum a_i=\infty$ and $\sum a_i^2\leqslant\infty$.

  2. Maximization step : compute $\theta^{(i)}=(\lambda^{(i)},\sigma^{2(i)},V^{(i)})$ by maximizing the complete likelihood (see (1.5.35),(1.5.36),(1.5.37)):

$$\lambda^{(i)} = \frac{s_1^{(i)}-1}{n-1}, \tag{1.5.40}$$

$$\sigma^{2(i)} = \frac{s_2^{(i)}}{n-s_1^{(i)}}, \tag{1.5.41}$$

$$V^{(i)} = \frac{\sum_{t=1}^{n}(y_t-\bar{y})^2-s_2^{(i)}}{s_1^{(i)}}-\sigma^{2(i)}. \tag{1.5.42}$$

Remarks:

1. We choose a decreasing sequence (a_i) in order to obtain a pointwise convergence of the sequence $(\theta^{(i)})$ to a value $\theta^\star$ (see [24] for results concerning stochastics algorithms

and the many references therein). A satisfactory schedule consists in setting $a_i = 1$ during some iterations (about 10 iterations in the practice), for converging quickly to a neighborhood of $\theta^\star$ and then, (a_i) decreases as $(1/i)$.

2. It was shown by Delyon *et al.* [22] that SAEM converges to a (local or global) maximum of the observed data likehood $g(y;\theta)$ under very general conditions, but assuming exact and independent simulations of the missing data at each iterations. Here, the sequence of missing data $(r^{(i)})$ is a Markov Chain, and this result does not apply directly. Nevertheless, by using the results of Metivier and Priouret [49] for this kind of situation, we can show that the algorithm described above converges to a maximum of $g(y;\theta)$ if the sequence of parameters $(\theta^{(i)})$ belongs to a compact set. Then, the slight technical stabilization device proposed by Delyon *et al.* [22] for the SAEM algorithm ensures the compactness of $(\theta^{(i)})$, and its convergence to a maximum of the observed likelihood.

We propose in Figure 5 a numerical example of this algorithm. A series y of length 1 000 was simulated with the following parameters: $\lambda = 0.01$, $V = 1$ and $\sigma^2 = 0.1$. This series is displayed Figure 5-a together with the sequence of means and the change-points. The number of iterations of MCMC to perform before updating $\theta^{(i)}$ was fixed to $M = 1\,000$. The sequence of stepsizes (a_i) was such that $a_i = 1$ for $1 \leqslant i \leqslant 10$, and $a_i = 1/(i-10)$ for $i \geq 11$. The sequences $(\lambda^{(i)})$, $(V^{(i)})$ and $(\sigma^{2(i)})$ are displayed Figure 5-b, 5-c and 5-d. The algorithm quickly converges, and after 30 iterations, the estimated parameters are $\lambda^{(30)} = 0.007$, $V^{(30)} = 1.469$ and $\sigma^{2(30)} = 0.091$.

Conclusion

We have proposed an attractive methodology for the change-points problem, in a Bayesian context. The probabilistic model makes use of an non-observed sequence r, and a MCMC algorithm can be used for estimating the posterior distribution of this change-points process r. Numerical experiments have clearly shown that this procedure is much more faster than the Reversible Jump algorithm. Furthermore, the hyperparameters of the model are estimated, rather than arbitrary chosen. We have also seen that a slight modification of the sampler allows to select the most likely configurations of change-points.

The main advantage of this method is the ability to perform automatically different tasks. We think that this kind of approach should not be restricted to the problem of detecting change-points in a signal contaminated by an additive (or a multiplicative) noise. Indeed, it should be interesting and useful to extend this approach for detecting changes in the spectrum of a signal, for example.

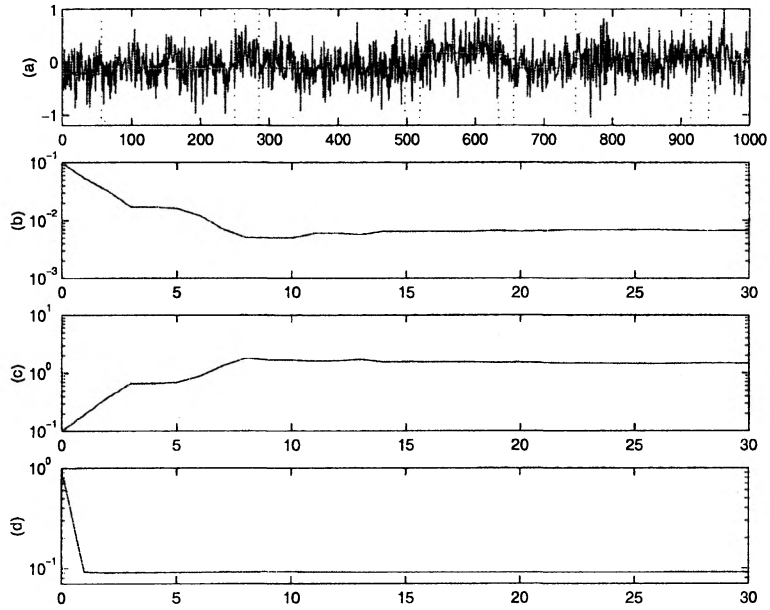


Figure 1.5: Estimation of the hyper-parameters. (a) the signal y, the mean m and the change-point instants τ, simulated with $\lambda = 0.01$, $V = 1.5$ and $\sigma^2 = 0.1$, (b), (c) and (d) the sequences of estimates $(\lambda^{(i)})$, $(V^{(i)})$ and $(\sigma^{2(i)})$.

Appendix A

PROOF OF THE FORMULAE (1.2.11) (1.2.12) and (1.2.13):

Using equations (1.2.6) and (1.2.7), we have

(1.5.43)

$$
\begin{aligned}
p(m|y,r;\theta)h(y|r;\theta) &= h(y|r,m;\sigma^2)\pi(m|r;\mu,V) \\
&= (2\pi\sigma^2)^{-\frac{n}{2}} e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{K_r}\sum_{t=\tau_{k-1}+1}^{\tau_k}(y_t-m_k)^2} \prod_{k=1}^{K_r}\left(\frac{2\pi V}{n_k}\right)^{-\frac{1}{2}} e^{-\frac{n_k}{2V}(m_k-\mu)^2} \\
&= \prod_{k=1}^{K_r}(2\pi V_k)^{-\frac{1}{2}} e^{-\frac{1}{2V_k}(m_k-\mu_k)^2} \\
&\quad \times (2\pi\sigma^2)^{-\frac{n}{2}} \prod_{k=1}^{K_r}\left(\frac{V}{n_k V_k}\right)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\left(\sum_{t=\tau_{k-1}+1}^{\tau_k}\frac{y_t^2}{\sigma^2}+\frac{n_k\mu^2}{V}-\frac{\mu_k^2}{V_k}\right)\right\}
\end{aligned}
$$

where

$$\mu_k = \frac{V\sigma^2}{V+\sigma^2}\left(\frac{\bar{y}_k}{\sigma^2}+\frac{\mu}{V}\right)$$

and

$$V_k = \frac{V\sigma^2}{n_k(V+\sigma^2)}$$

By identification, we obtain the distribution of m conditional on a given configuration r and the observations y

$$p(m|y,r;\theta) = \prod_{k=1}^{K_r} (2\pi V_k)^{-\frac{1}{2}} e^{-\frac{1}{2V_k}(m_k-\mu_k)^2}$$

where μ_k and V_k are respectively the posterior mean and variance of m_k.

PROOF OF LEMMA 1 :

First, remark that according to (1.5.43)

$$\begin{aligned} h(y|r;\theta) &= (2\pi\sigma^2)^{-\frac{n}{2}} \prod_{k=1}^{K_r} \left(\frac{V}{n_k V_k}\right)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\left(\sum_{t=\tau_{k-1}+1}^{\tau_k} \frac{y_t^2}{\sigma^2} + \frac{n_k\mu^2}{V} - \frac{\mu_k^2}{V_k}\right)\right\} \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} \left(\frac{\sigma^2+V}{\sigma^2}\right)^{-\frac{K_r}{2}} \exp\left\{-\frac{1}{2(V+\sigma^2)}\left(\sum_{t=1}^{n}(y_t-\mu)^2 + \frac{V}{\sigma^2}S_r\right)\right\} \end{aligned} \tag{1.5.44}$$

We can then obtain Lemma 1 using (1.2.4) and (1.5.44). Indeed,

$$\begin{aligned} p(r|y;\theta) &= \frac{h(y|r;\theta)\pi(r;\lambda)}{g(y;\theta)} \\ &= C(y;\theta)\ \exp\{-\phi S_r - \gamma K_r\} \end{aligned} \tag{1.5.45}$$

where

$$\phi = \frac{V}{2\sigma^2(\sigma^2+V)} \quad , \quad \gamma = \frac{1}{2}\log\left(\frac{\sigma^2+V}{\sigma^2}\right) + \log\left(\frac{1-\lambda}{\lambda}\right).$$

PROOF OF LEMMA 2:

Since $f(y,r;\theta) = h(y|r;\theta)\pi(r;\lambda)$, we can easily obtain (1.5.33) from (1.2.4) and (1.2.7) :

$$\begin{aligned} f(y,r;\theta) &= (2\pi\sigma^2)^{-\frac{n}{2}} \left(\frac{\sigma^2+V}{\sigma^2}\right)^{-\frac{K_r}{2}} \lambda^{K_r-1}(1-\lambda)^{n-K_r+1} \\ &\quad \times \exp\left\{-\frac{1}{2(V+\sigma^2)}\left(\sum_{t=1}^{n}(y_t-\mu)^2 + \frac{V}{\sigma^2}S_r\right)\right\} \end{aligned} \tag{1.5.46}$$

Then, the expression of the maximum likelihood estimate of θ is directly obtained by maximizing $f(y,r;\theta)$ with respect to θ.

Appendix B

PROBABILITIES OF ACCEPTANCE FOR THE REVERSIBLE JUMP ALGORITHM:

In the following formulae, the means m are assumed to be centered ($\mu = 0$).

Let (m, τ) be the current state and $(\tilde{m}, \tilde{\tau})$ be the proposed candidate. The probability of acceptance is

$$\begin{aligned}
\alpha((m,\tau),(\tilde{m},\tilde{\tau})) &= \min\left\{1, \frac{p(\tilde{m},\tilde{\tau}|y;\theta)}{p(m,\tau|y;\theta)} \times \frac{j((\tilde{m},\tilde{\tau}),(m,\tau))q_2(u')}{j((m,\tau),(\tilde{m},\tilde{\tau}))q_1(u)} \times \left|\frac{\partial(\tilde{m},\tilde{\tau},u')}{\partial(m,\tau,u)}\right|\right\} \\
&= \min\left\{1, \frac{h(y|\tilde{m},\tilde{\tau};\sigma^2)}{h(y|m,\tau;\sigma^2)} \times \frac{\pi(\tilde{m};\tilde{\tau};\theta)}{\pi(m,\tau;\theta)} \times \frac{j((\tilde{m},\tilde{\tau}),(m,\tau))q_2(u')}{j((m,\tau),(\tilde{m},\tilde{\tau}))q_1(u)} \times \left|\frac{\partial(\tilde{m},\tilde{\tau},u')}{\partial(m,\tau,u)}\right|\right\} \\
&= \min\{1, A\}.
\end{aligned}$$

where

- $j((\tilde{m},\tilde{\tau}),(m,\tau))$ (resp $j((m,\tau),(\tilde{m},\tilde{\tau}))$) is the probability of choosing the move from $(\tilde{m},\tilde{\tau})$ to (m,τ) (resp from (m,τ) to $(\tilde{m},\tilde{\tau})$).

- u (resp u') is generated from the proposal density $q_1(u)$ (resp $q_2(u')$) such that $(\tilde{m},\tilde{\tau},u') = f(m,\tau,u)$ where f is a specific inversible fonction.

- the final term is the Jacobian arising from the change of variables from (m,τ,u) to $(\tilde{m},\tilde{\tau},u')$.

We can compute A for the different moves :

1^{st} move

$$A = \frac{h(y|\tilde{m},\tilde{\tau};\theta)}{h(y|m,\tau;\theta)} \times \exp\{-\frac{1}{2V}(\tilde{m}_k^2 - m_k^2)(\tau_k - \tau_{k-1})\} \times \frac{\tilde{m}_k}{m_k}$$

2^{nd} move

$$\begin{aligned} A = & \frac{h(y|\tilde{m},\tilde{\tau};\theta)}{h(y|m,\tau;\theta)} \times \frac{\lambda}{1-\lambda} \times (2\pi)^{-\frac{1}{2}} \left(\frac{(\tau_k - \tilde{\tau}_k)(\tilde{\tau}_k - \tau_{k-1})}{(\tau_k - \tau_{k-1})}\right)^{\frac{1}{2}} \\ & \times \exp\left\{-\frac{1}{2V}\left(\tilde{m}_k^2(\tilde{\tau}_k - \tau_{k-1}) + \tilde{m}_{k+1}^2(\tau_k - \tilde{\tau}_k) - m_k^2(\tau_k - \tau_{k-1})\right)\right\} \\ & \frac{0.4(n-1)}{k+1}\left(\frac{\tau_k - \tau_{k-1}}{((\tilde{\tau}_k - \tau_{k-1})(\tau_k - \tilde{\tau}_k))^{1/2}}\right) \end{aligned}$$

3^{nd} move

$$\begin{aligned} A = & \frac{h(y|\tilde{m},\tilde{\tau};\theta)}{h(y|m,\tau;\theta)} \times \frac{1-\lambda}{\lambda} \times (2\pi)^{\frac{1}{2}} \left(\frac{(\tau_{k+1} - \tau_{k-1})}{(\tau_{k+1} - \tau_k)(\tau_k - \tau_{k-1})}\right)^{\frac{1}{2}} \\ & \times \exp\left\{-\frac{1}{2V}\left(\tilde{m}_k^2(\tau_{k+1} - \tau_{k-1}) - m_{k+1}^2(\tau_{k+1} - \tau_k) - m_k^2(\tau_k - \tau_{k-1})\right)\right\} \\ & \times \frac{k+1}{0.4(n-1)}\left(\frac{((\tau_{k+1} - \tau_k)(\tau_k - \tau_{k-1}))^{1/2}}{\tau_{k+1} - \tau_{k-1}}\right) \end{aligned}$$

4^{th} move

$$A = \frac{h(y|\tilde{m},\tilde{\tau};\theta)}{h(y|m,\tau;\theta)} \times \left(\frac{(\tau_{k+1} - \tilde{\tau}_k)(\tilde{\tau}_k - \tau_{k-1})}{(\tau_{k+1} - \tau_k)(\tau_k - \tau_{k-1})}\right)^{\frac{1}{2}} \exp\{-\frac{1}{2V}(m_k^2 - m_{k+1}^2)(\tilde{\tau}_k - \tau_k)\}$$

Deuxième partie

Détection de ruptures dans la moyenne par méthode de Sélection de Modèle

Chapitre 2

Modèle de détection de ruptures dans la moyenne et méthode de sélection de modèle.

2.1 Introduction

Comme dans la partie I, nous considérons le problème de détection de ruptures dans la moyenne d'un signal Gaussien. Les sauts de moyennes, la localisation et le nombre d'instants de ruptures sont inconnus, et l'objectif est de les estimer.

Dans ce chapitre, nous nous intéressons exclusivement à une approche de sélection de modèle par pénalisation pour la détection de ruptures dans la moyenne. Plusieurs auteurs considèrent cette approche, parmi eux Yao [66], Miao *et. al* [51] et plus récemment Lavielle et Moulines [42]. Ils proposent un estimateur des instants de ruptures des moindres carrés pénalisé : pour un nombre de ruptures fixé, l'estimateur des instants de ruptures minimise le critère des moindres carrés, ensuite le nombre de ruptures est estimé par minimisation du critère pénalisé où la pénalité est proportionnelle au nombre de ruptures. Par exemple, Yao [66] considère une pénalité de la forme $D \log n/n$, correspondant au critère de Schwarz, où n est la taille de l'échantillon observé et D le nombre de segments, c'est-à-dire le nombre de ruptures plus une. Lavielle et Moulines [42] propose une pénalité de la forme plus générale $D\beta_n$ où β_n est une suite dépendant de n convergent vers 0 avec une vitesse appropriée. Ils obtiennent des résultats de consistance pour les estimateurs des instants de ruptures et de leur nombre. Leurs résultats sont asymptotiques, c'est-à-dire pour une taille d'échantillon qui tend vers l'infini et en pratique, β_n doit être fixé et choisi afin d'obtenir des solutions satisfaisantes.

Nous proposons ici une approche qui se place dans un contexte non-asymptotique, c'est-à-dire pour une taille d'échantillon fixée. Nous posons le problème sous un autre angle : les instants de ruptures et les moyennes sont assimilés à une fonction constante par morceaux

notée s et l'objectif est d'estimer la fonction s. Nous considérons la méthode d'estimation non-paramétrique par sélection de modèles proposée par Birgé et Massart [7] dans le cadre des processus linéaires Gaussiens. Cette méthode se décompose en trois parties : dans un premier temps, nous définissons une collection de partitions notée $\mathcal{M}_n$ et associons à chaque partition de $\mathcal{M}_n$, un modèle qui est le sous-espace linéaire des fonctions constantes par morceaux construites sur cette partition. Pour pouvoir bien approcher la fonction s et puisque nous n'avons pas d'information a priori, nous prenons la collection de partitions la plus riche possible, c'est-à-dire l'ensemble de toutes les partitions de la grille $\{1, \ldots, n\}$. Dans un second temps, nous estimons la fonction s sur chacun des modèles en minimisant une fonction de contraste particulière, qui est le critère classique des moindres carrés ([66], [51] ou [42]). Nous obtenons ainsi une collection d'estimateurs. Dans un troisième temps, nous sélectionnons parmi cette collection le meilleur estimateur $\tilde{s}$ en sélectionnant la meilleure partition par la minimisation d'un critère pénalisé construit uniquement sur les observations, un critère des moindres carrés pénalisé. Quand la collection de partitions est très riche, comme c'est le cas ici, une pénalité proportionnelle à la dimension de la partition comme dans le C_p de Mallows [47] ne peut être utilisée (cf [7]). Pour obtenir la bonne forme de la fonction de pénalité, nous appliquons un résultat obtenu par Birgé et Massart dans le cadre des processus linéaires Gaussiens [7]. Elle est choisie de manière à obtenir la meilleure borne de risque de l'estimateur pénalisé, défini par $\mathbb{E}_s[\|\tilde{s} - s\|_n^2]$, possible. Nous verrons qu'elle prend en compte la complexité de la collection de partitions considérée, c'est-à-dire du nombre de partitions de même dimension.
Il est important de souligner que cette méthode ne mène pas à la sélection de la "meilleure" fonction associée au "vrai" nombre de ruptures (c'est-à-dire détecter toutes les ruptures) mais plutôt à sélectionner une fonction qui "approche" le mieux possible s au sens du risque. C'est donc une situation dans laquelle il peut être préférable d'ignorer certaines ruptures correspondant à des sauts de moyennes très faibles.

Dans la section 2.2, nous présentons le modèle de ruptures dans la moyenne. Dans la section 2.3, nous appliquons la méthode de sélection de modèle : dans la sous-section 2.3.1, nous commençons, pour une partition fixée de $\mathcal{M}_n$, par définir le sous-espace linéaire des fonctions constantes par morceaux construites sur cette partition. Nous introduisons ensuite la fonction de contraste, puis estimons la fonction s sur le sous-espace considéré. Nous définissons alors la fonction de perte et calculons le risque de l'estimateur obtenu. Dans la section 2.3.2, nous donnons la collection d'estimateurs. Enfin, dans la section 2.4, nous définissons l'estimateur pénalisé et donnons la forme de la fonction de pénalité et un contrôle du risque de cet estimateur.

2.2 Présentation du modèle

Soit y un processus réel tel que

$$y_t = s(x_t) + \varepsilon_t \quad t = 1, \ldots, n \tag{2.2.1}$$

où $x_t = \frac{t}{n}$ et ε est un bruit blanc gaussien de variance σ^2. La fonction s est supposée constante par morceaux. Ainsi, il existe des instants $\tau_0 = 0 < \tau_1 < \ldots < \tau_{K_r} = 1$ et une suite finie $(s_1, \ldots, s_{K_r})$ tels que

$$s = \sum_{k=1}^{K_r} s_k \mathbb{1}_{I_k} \quad \text{with } I_k =]\tau_{k-1}, \tau_k]$$

avec la convention $\tau_0 = 0$ et $\tau_{K_r} = 1$.
Cela signifie que $K_r - 1$ changements affectent la moyenne de y en des instants inconnus $(t_k, 1 \leqslant k \leqslant K_r - 1)$ où $t_k = [n\tau_k]$. Nous ne parlerons néanmoins que des instants renormalisés $(\tau_k, 1 \leqslant k \leqslant K_r - 1)$. Comme dans la première partie, les sauts de moyennes, la localiation et le nombre des instants de ruptures sont inconnus. A titre d'exemple, supposons qu'il existe deux ruptures aux instants 0.2 et 0.6, soit trois segments. Prenons respectivement comme sauts de moyennes 0, 2 et 1, alors la fonction s associée est définie par :

$$s(x) = \begin{cases} 0 & \text{si } 0 < x \leqslant 0.2 \\ 2 & \text{si } 0.2 < x \leqslant 0.6 \\ 1 & \text{si } 0.6 < x \leqslant 1 \end{cases}$$

La représentation graphique de cette fonction est donnée par la Figure 2.1. La Figure 2.2 représente une réalisation du processus y de taille $n = 500$ simulée à partir de la fonction s et d'un bruit blanc de variance $\sigma^2 = 1$.

L'objectif est l'estimation du vecteur de moyennes $(s_k)_{1 \leqslant k \leqslant K_r}$ et des instants de ruptures $(\tau_k)_{1 \leqslant k \leqslant K_r - 1}$ à partir des n observations $y_1, \ldots, y_n$. Au lieu de chercher à estimer ces deux vecteurs, nous considérons le problème comme un problème non-paramétrique en estimant la fonction s par une méthode de sélection de modèle.

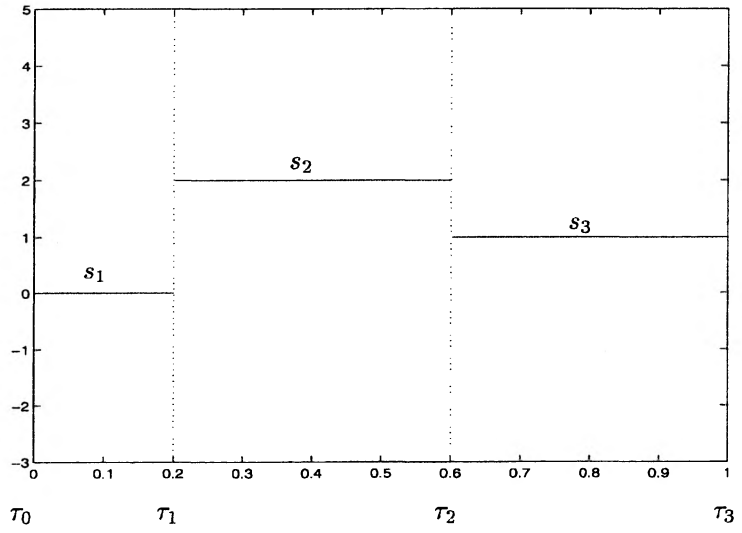


FIG. 2.1: *Représentation de la fonction s.*

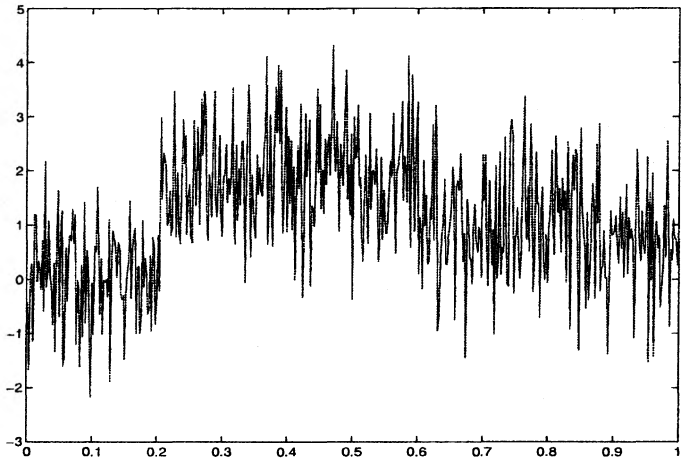


FIG. 2.2: *Une réalisation de y.*

2.3 Méthode de sélection de modèle

Dans cette section, nous décrivons successivement les étapes de la méthode de sélection de modèle. Nous introduisons les différentes notions théoriques et donnons la "bonne" forme de la pénalité.

2.3.1 Estimation sur un modèle

Nous nous fixons une partition m de $]0,1]$ de dimension D_m. Cette partition est associée à la configuration des instants de ruptures renormalisés $\tau = (\tau_1, \ldots, \tau_{D_m})$ de la façon suivante :

$$m = \bigcup_{k=1}^{D_m} I_k,$$

telle que

$$\begin{cases} I_k =]\tau_{k-1}, \tau_k] & k = 1, \ldots, D_m, \\ I_k \cap I_{k'} = \emptyset & \text{pour } k = 1, \ldots, D_m \text{ et } k' = 1, \ldots, D_m. \end{cases}$$

Nous définissons le **modèle** associé à la partition m et noté $\mathcal{S}_m$, comme étant le sous-espace linéaire des fonctions constantes par morceaux construites sur la partition m :

$$\mathcal{S}_m = \left\{ u = \sum_{k=1}^{D_m} u_k \mathbb{1}_{I_k}, (u_k)_{k=1,\ldots,D_m} \in \mathbb{R}^{D_m} \right\}. \tag{2.3.2}$$

La dimension de $\mathcal{S}_m$ est donc celle de la partition m, soit D_m.

Nous notons $\mathcal{S}$ l'espace de fonctions auquel appartient la fonction inconnue s. Dans notre cadre d'étude $\mathcal{S}$ est l'espace de Hilbert $\mathcal{S} = (\mathbb{L}^2([0,1]), \mu_n)$ où μ_n est la mesure empirique sur $(x_t)_{1\leqslant t\leqslant n}$: $\mu_n = \frac{1}{n}\sum_{t=1}^{n} \delta_{x_t}$ avec δ la mesure de dirac. Pour toute fonction $u \in \mathcal{S}$, nous ne considérons que ses valeurs aux points $x_1, \ldots, x_n$. Cette fonction peut donc être identifiée au vecteur $(u(x_t))_{1\leqslant t\leqslant n}$ qui appartient à l'espace $\mathbb{R}^n$. De même le sous-espace linéaire $\mathcal{S}_m$, défini en (2.3.2), peut être considéré comme un sous-espace linéaire de $\mathbb{R}^n$. Nous notons $\|.\|_n^2$ la norme Euclidienne de $\mathbb{R}^n$ renormalisée par n et définie pour $u \in \mathcal{S}$ par :

$$\|u\|_n^2 = \frac{1}{n}\sum_{t=1}^{n} u^2(x_t). \tag{2.3.3}$$

2.3.1.1 Estimateur du minimum de contraste

Dans cette sous-section, nous estimons s sur le sous-espace linéaire $\mathcal{S}_m$ par minimisation d'une fonction de contraste.

Nous considérons le contraste empirique qui est le critère des moindres carrés défini pour tout $u \in \mathcal{S}$ par :

$$\gamma_n(u) \;=\; \frac{1}{n}\sum_{t=1}^{n} [y_t - u(x_t)]^2 \,. \tag{2.3.4}$$

D'après la norme Euclidienne renormalisée définie en (2.3.3), le contraste empirique γ_n s'écrit également :

$$\gamma_n(u) = \|y - u\|_n^2. \tag{2.3.5}$$

L'estimateur de s dans le sous-espace linéaire $\mathcal{S}_m$ est celui qui minimise le contraste empirique défini par l'équation (2.3.4) sur $\mathcal{S}_m$. Cet estimateur, appelé l'estimateur du minimum de contraste, est défini par :

$$\hat{s}_m = \underset{u\in\mathcal{S}_m}{\operatorname{argmin}}\, \gamma_n(u). \tag{2.3.6}$$

D'après la définition (2.3.2), pour une fonction u du sous-espace linéaire $\mathcal{S}_m$, il existe une suite finie $(u_1, \ldots, u_{D_m})$ telle que

$$u = \sum_{k=1}^{D_m} u_k \mathbb{1}_{I_k},$$

où $I_k =]\tau_{k-1}, \tau_k]$ est le k^{eme} segment de la partition m et $\tau_k = \frac{t_k}{n}$ pour tout $k = 1, \ldots, D_m$. Le contraste empirique pris en la fonction u vaut :

$$\gamma_n(u) = \frac{1}{n}\sum_{k=1}^{D_m}\sum_{t=t_{k-1}+1}^{t_k} (y_t - u_k)^2.$$

Nous montrons facilement que le contraste empirique est minimum pour $u_k = \bar{y}_k$ pour $k = 1, \ldots, D_m$, où

$$\bar{y}_k = \frac{1}{n_k} \sum_{t=t_{k-1}+1}^{t_k} y_t,$$

est la moyenne empirique de y sur le segment I_k avec $n_k = t_k - t_{k-1}$. Ainsi, l'estimateur du minimum de contraste de s sur le sous-espace linéaire $\mathcal{S}_m$ est défini par :

$$\hat{s}_m = \sum_{k=1}^{D_m} \bar{y}_k \mathbb{1}_{I_k}. \tag{2.3.7}$$

Par la relation qui existe entre le contraste et la norme Euclidienne renormalisée par n donnée par l'équation (2.3.5), cet estimateur est aussi appelé estimateur par projection de s sur le sous-espace linéaire $\mathcal{S}_m$ au sens de la norme $\|.\|_n^2$.

2.3.1.2 Risque de l'estimateur

Pour mesurer la qualité de l'estimateur, nous cherchons à voir s'il est proche de la fonction s. Nous définissons alors une fonction de perte et une fonction de risque.

Definition 2.3.1. *Etant donnée une fonction de contraste empirique γ_n,* **la fonction de perte** *notée l, est associée à γ_n par la relation suivante :*

$$l(s, u) = \mathbb{E}_s \left[\gamma_n(u) - \gamma_n(s)\right],$$

pour toute fonction $u \in \mathcal{S}$. Cette fonction est positive et s'annule si et seulement si $u = s$.

Definition 2.3.2. Le risque *de l'estimateur $\hat{s}_m$ de s est défini à partir de la fonction de perte l par :*

$$\mathbb{E}_s \left[l(s, \hat{s}_m)\right].$$

Dans notre étude, d'après la fonction de constraste empirique choisie, donnée en (2.3.4), la fonction de perte l est définie pour tout $u \in \mathcal{S}$ par :

$$\begin{aligned} l(s, u) &= \mathbb{E}_s \left[\gamma_n(u) - \gamma_n(s)\right] \\ &= \frac{1}{n} \sum_{t=1}^{n} \left[s(x_t) - u(x_t)\right]^2 \\ &= \|s - u\|_n^2. \end{aligned}$$

La perte de l'estimateur par minimum de contraste $\hat{s}_m$ vaut :

$$l(s, \hat{s}_m) = \|s - \hat{s}_m\|_n^2,$$

et son risque est le risque quadratique :

$$\mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right].$$

Pour le calculer, nous utilisons s_m la projection de s sur le sous-espace linéaire $\mathcal{S}_m$ au sens de la perte, c'est-à-dire au sens de la norme $\|.\|_n^2$, qui est défini par :

$$s_m = \sum_{k=1}^{D_m} \bar{s}_k \mathbb{1}_{I_k}, \tag{2.3.8}$$

où $\bar{s}_k$ est la moyenne empirique de s sur le segment I_k :

$$\bar{s}_k = \frac{1}{n_k} \sum_{t=t_{k-1}+1}^{t_k} s(x_t).$$

Par le théorème de Pythagore, la perte de l'estimateur $\hat{s}_m$ se décompose en la somme de deux termes :

$$\|s - \hat{s}_m\|_n^2 = \|s - s_m\|_n^2 + \|s_m - \hat{s}_m\|_n^2,$$

ce qui permet d'en déduire une évaluation du risque de $\hat{s}_m$

$$\mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right] = \|s - s_m\|_n^2 + \mathbb{E}_s \left[\|s_m - \hat{s}_m\|_n^2 \right].$$

En utilisant les expressions de $\hat{s}_m$ et s_m données respectivement par les égalités (2.3.7) et (2.3.8), nous obtenons :

$$\mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right] = \|s - s_m\|_n^2 + \mathbb{E}_s \left[\| \sum_{k=1}^{D_m} (\bar{s}_k - \bar{y}_k) \mathbb{1}_{I_k} \|_n^2 \right].$$

D'après la définition de la norme Euclidienne renormalisée donnée par l'égalité (2.3.3), nous en déduisons :

$$\mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right] = \|s - s_m\|_n^2 + \frac{1}{n} \sum_{k=1}^{D_m} n_k \mathbb{E}_s \left[(\bar{s}_k - \bar{y}_k)^2 \right].$$

D'après le modèle défini en (2.2.1), $\mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right]$ vaut :

$$\begin{aligned} \mathbb{E}_s \left[\|s - \hat{s}_m\|_n^2 \right] &= \|s - s_m\|_n^2 + \frac{1}{n} \sum_{k=1}^{D_m} n_k \mathbb{E}_s \left[\left(\frac{1}{n_k} \sum_{t=t_{k-1}+1}^{t_k} (s(x_t) - y_t) \right)^2 \right] \\ &= \|s - s_m\|_n^2 + \frac{1}{n} \sum_{k=1}^{D_m} n_k \mathbb{E}_s (\bar{\varepsilon}_k^2). \end{aligned}$$

Par hypothèse, ε_t est une gaussienne centrée de variance σ^2. Par conséquent, $\bar{\varepsilon}_k$ est une gausienne centrée de variance $\frac{\sigma^2}{n_k}$. Le second terme vaut alors :

$$\frac{1}{n}\sum_{k=1}^{D_m} n_k \mathbb{E}_s\left[\bar{\varepsilon}_k^2\right] = \frac{D_m}{n}\sigma^2.$$

Le risque de l'estimateur $\hat{s}_m$ se décompose donc en la somme de deux termes :

$$\mathbb{E}_s\left[\|s-\hat{s}_m\|_n^2\right] = \|s-s_m\|_n^2 + \frac{D_m}{n}\sigma^2. \tag{2.3.9}$$

Le premier terme est un terme de biais : il représente l'erreur d'approximation de s sur le sous-espace linéaire $\mathcal{S}_m$. Le second terme est un terme de variance : il représente l'erreur d'estimation dans $\mathcal{S}_m$. Ce terme est proportionnel à la dimension de $\mathcal{S}_m$ qui est D_m (*i.e.* le nombre de paramètres à estimer).

2.3.2 Collection de modèles

Nous considérons une collection de partitions notée $\mathcal{M}_n$ qui est l'ensemble de toutes les partitions construites sur la grille $\{x_1,\ldots,x_n\} = \{1/n,\ldots,1\}$:

$$\mathcal{M}_n = \mathcal{P}\left(\{1/n,\ldots,1\}\right).$$

Cette collection dépend de la taille de l'échantillon n. Pour chaque partition m de $\mathcal{M}_n$, nous considérons $\mathcal{S}_m$ le sous-espace linéaire des fonctions constantes par morceaux construite sur la partition m. Nous obtenons une collection de sous-espaces linéaires

$$\{\mathcal{S}_m, m \in \mathcal{M}_n\},$$

et nous déduisons la collection d'estimateurs du minimum de contraste

$$\{\hat{s}_m, m \in \mathcal{M}_n\}.$$

L'objectif est de sélectionner le "meilleur" estimateur parmi cette collection. L'idéal serait de sélectionner la partition de $\mathcal{M}_n$ qui fournit l'estimateur qui a le plus petit risque. D'après la forme de ce risque, défini en (2.3.9), nous remarquons que plus la dimension de $\mathcal{S}_m$ est grande, plus le terme de biais diminue mais plus le terme de variance augmente (le nombre de paramètres à estimer augmente). Ainsi, la partition idéale est celle qui fait le meilleur compromis entre ces deux tendances inverses. Malheureusement, le risque, plus précisement le terme de biais, dépend de la fonction s inconnue, et donc la partition idéale aussi. Une telle partition ne peut donc pas être utilisée pour construire un estimateur de s. L'objectif de la sélection de modèle est de construire un critère uniquement à partir des données qui sélectionne une partition qui se comporte aussi bien que la partition idéale en terme de risque.

2.4 Sélection de modèle

Dans cette section, nous donnons la procédure générale de la sélection de modèle qui construit l'estimateur sélectionné à l'aide d'un critère pénalisé, puis nous utilisons un résultat obtenu par Birgé et Massart dans le cadre des processus linéaires Gaussiens [7] pour obtenir la "bonne" forme du critère pénalisé dans notre cadre d'étude.

Soit $pen_n : \mathcal{M}_n \longrightarrow \mathbb{R}^+$ une fonction de pénalité, l'estimateur du minimum de contraste pénalisé est défini par :

$$\tilde{s} = \hat{s}_{\hat{m}} \tag{2.4.10}$$

où $\hat{m}$ minimise sur la collection de partitions $\mathcal{M}_n$ le critère pénalisé suivant :

$$\begin{aligned} crit_n(m) &= \gamma_n(\hat{s}_m) + pen_n(m) \\ &= \frac{1}{n}\sum_{k=1}^{D_m}\sum_{t=t_{k-1}+1}^{t_k}(y_t - \bar{y}_k)^2 + pen_n(m). \end{aligned} \tag{2.4.11}$$

La construction d'un "bon" estimateur de s se réduit alors au choix d'une "bonne" fonction de pénalité pen_n.

Proposition 2.4.1. *Il existe des constantes c_1 et c_2 telles que si la pénalité est définie pour toute partition $m \in \mathcal{M}_n$ par :*

$$pen_n(m) = \frac{\sigma^2}{n} D_m \left(c_1 \log \frac{n}{D_m} + c_2 \right), \tag{2.4.12}$$

alors il existe des constantes $C(c_1, c_2)$ et $C'(c_1, c_2)$ telles que le risque de l'estimateur pénalisé $\tilde{s}$, défini en (2.4.10), est controlé par

$$\mathbb{E}_s\left[\|\tilde{s} - s\|_n^2\right] \leqslant C(c_1, c_2) \inf_{m \in \mathcal{M}_n} \left[\|s - s_m\|_n^2 + pen_n(m)\right] + C'(c_1, c_2)\frac{\sigma^2}{n}. \tag{2.4.13}$$

Preuve. La preuve est basée sur le théorème proposé par Birgé et Massart [7] dans le cadre des processus linéaires Gaussiens que nous rappelons ici.

Théorème 2.4.2. *Soit y un processus linéaire Gaussien défini sur un sous-espace linéaire $\mathcal{S}$ d'un espace de Hilbert $\mathbb{H}$ par :*

$$y(u) = < s, u > + \sigma Z(u) \quad \textit{for all } u \in \mathcal{S}.$$

où $s \in \mathbb{H}$ est la moyenne du processus y, σ sa variance et Z est un processus linéaire isonormal indéxé par $\mathcal{S}$. Soit $\{\mathcal{S}_m, m \in \mathcal{M}_n\}$ une famille finie de sous-espaces linéaires de $\mathcal{S}$ de dimension respective D_m et $\{L_m\}_{m \in \mathcal{M}_n}$ une famille de poids positifs satisfaisant

$$\Sigma = \sum_{\{m \in \mathcal{M}_n | D_m > 0\}} e^{-L_m D_m} < +\infty$$

Considérons alors une fonction de pénalité pen sur $\mathcal{M}_n$ telle que pour tout $m \in \mathcal{M}_n$ et une constante $K > 1$,

$$pen(m) \geq K\sigma^2 \frac{D_m}{n}(1+\sqrt{2L_m})^2, \tag{2.4.14}$$

alors il existe des constantes $C(K)$ et $C'(K)$ indépendantes de s, σ et Σ telles que l'estimateur pénalisé $\tilde{s}$ défini en (2.4.10) satisfait

$$\mathbb{E}_s[l(\hat{s}_{\hat{m}}, s)] \leqslant C(K) \inf_{m\in\mathcal{M}_n} [l(s, s_m) + pen(m)] + C'(K)\frac{\Sigma}{n}\sigma^2. \tag{2.4.15}$$

Nous constatons par ce théroème que le choix de la fonction de pénalité se réduit au choix des poids $\{L_m\}_{m\in\mathcal{M}_n}$ tel que la famille $\{e^{-L_m D_m}\}_{m\in\mathcal{M}_n}$ soit sommable.

Prenons L_m comme étant une fonction de la dimension D, soit L_D, nous obtenons

$$\begin{aligned}
\Sigma &= \sum_{m\in\mathcal{M}} e^{-L_m D_m} \\
&= \sum_{D=1}^{n} e^{-DL_D} \#\{m \in \mathcal{M}_n, D_m = D\} \\
&\leqslant \sum_{D=1}^{n} e^{-DL_D} C_n^D \\
&\leqslant \sum_{D=1}^{n} e^{-DL_D} \left(\frac{en}{D}\right)^D \\
&\leqslant \sum_{D=1}^{n} e^{-D\left(L_D - 1 - \log\left(\frac{n}{D}\right)\right)}.
\end{aligned}$$

Ainsi, en prenant

$$L_D = 1 + \beta + \log\left(\frac{n}{D}\right),$$

avec $\beta > 0$ indépendant de D, nous obtenons directement

$$\Sigma = \Sigma_\beta < +\infty.$$

Plus précisément, $\Sigma \leqslant \sum_{D\geq 1} e^{-D\beta} = (1-e^{-\beta})^{-1} - 1$. Donc pour contrôler Σ, il suffit de prendre $\beta = \log 2$, soit pour $D \geq 1$,

$$L_D = 1 + \log 2 + \log\left(\frac{n}{D}\right).$$

En utilisant l'inégalité suivante :

$$2ab \leqslant \theta a^2 + \theta^{-1} b^2 \quad \forall \ a, b > 0 \text{ et } \forall \ 0 < \theta < 1,$$

nous avons que

$$\begin{aligned}\left(1+\sqrt{2L_{D_m}}\right)^2 &\leqslant \left[2(1+\theta)\log\left(\frac{n}{D_m}\right)\right] \\ &+ \left[(1+\theta^{-1})+2(1+\log 2)(1+\theta)\right].\end{aligned}$$

Donc en prenant

$$pen_n(m) = \frac{\sigma^2}{n} D_m \left(c_1 \log \frac{n}{D_m} + c_2\right),$$

l'inégalité (2.4.14) est vérifiée.

Nous concluons la démonstration de la proposition 2.4.1 en prenant $C(c_1, c_2) = C(K)$ et $C'(c_1, c_2) = C'(K)$.

Remarque 2.1. La pénalité ne dépend de la partition m que via sa dimension D_m. Le facteur $\log \frac{n}{D}$ apparaîssant dans cette pénalité est issu de la complexité de la collection de partitions $\mathcal{M}_n$ choisie, *i.e.* du nombre de partitions de même dimension dans cette collection, qui est ici C_{n-1}^{D-1}. Ce facteur peut paraître inhabituel : il n'apparaît pas dans la pénalité du critère du C_p de Mallows par exemple, donnée en (2.5.17).

Le critère pénalisé défini en (2.4.11) est facile à interpréter : le premier terme est lié à l'ajustement aux observations. Il est clair que plus la dimension de la partition est élevée, plus cet ajustement est meilleur. Le second terme, le terme de pénalité, qui ne dépend que de la dimension de la partition croît avec cette dimension. Son rôle est simple : il consiste à contrôler la dimension de la partition à sélectionner. Cette fonction de pénalité, définie en (2.4.12), dépend de deux constantes c_1 et c_2 dont les valeurs optimales ne sont pas accessibles théoriquement, et de la variance σ^2 qui est un paramètre inconnu. L'objectif des deux chapitres suivants est de calibrer les constantes c_1 et c_2 de façon optimale d'un point de vue sélection de modèle en considérant la variance du bruit connue, puis d'aborder le problème de l'estimation de la variance.

2.5 Annexe : heuristique de Mallows

Mallows [47] propose un critère, le Cp de Mallows dont l'heuristique est la suivante :

La partition idéale minimise le risque quadratique

$$\mathbb{E}_s[\|\hat{s}_m - s\|_n^2],$$

sur l'ensemble de toutes les partitions de la collection $\mathcal{M}_n$.
D'après l'expression du risque quadratique donnée par l'égalité (2.3.9), cette partition minimise sur $\mathcal{M}_n$

$$\|s - s_m\|_n^2 + \frac{D_m}{n}\sigma^2.$$

Par le théorème de Pythagore, nous avons que

$$\|s\|_n^2 = \|s - s_m\|_n^2 + \|s_m\|_n^2.$$

Par équivalence, la partition idéale minimise sur $\mathcal{M}_n$

$$-\|s_m\|_n^2 + \frac{D_m}{n}. \tag{2.5.16}$$

D'après l'expression de s_m donnée en (2.3.8), $\|s_m\|_n^2$ dépend de la fonction inconnue s. Or

$$\begin{aligned} \mathbb{E}_s[-\|\hat{s}_m\|_n^2] &= -\|s_m\|_n^2 - \mathbb{E}_s[\|\hat{s}_m - s_m\|_n^2] \\ &= -\|s_m\|_n^2 - \frac{D_m}{n}\sigma^2, \end{aligned}$$

donc $-\|\hat{s}_m\|_n^2 + \frac{D_m}{n}$ est un estimateur non biaisé de $\|s_m\|_n^2$. Nous remplaçons $\|s_m\|_n^2$ dans l'équation défini en (2.5.16) par son estimateur non biaisé. La recherche de la partition idéale revient alors à minimiser l'expression

$$-\|\hat{s}_m\|_n^2 + 2\frac{D_m}{n}\sigma^2.$$

sur $m \in \mathcal{M}_n$.
De plus, en remarquant que

$$\gamma_n(\hat{s}_m) = \frac{1}{n}\sum_{t=1}^{n} y_t^2 - \|\hat{s}_m\|_n^2,$$

cela mène au critère du Cp de Mallows qui est définie pour toute partition $m \in \mathcal{M}_n$ par :

$$C_p(m) = \gamma_n(\hat{s}_m) + 2\frac{D_m}{n}\sigma^2. \tag{2.5.17}$$

Chapitre 3

Calibration des constantes de la pénalité

3.1 Introduction

Dans le chapitre 1, nous avons proposé une pénalité de la forme

$$pen_n(m) = \frac{\sigma^2}{n} D_m \left(c_1 \log \frac{n}{D_m} + c_2 \right).$$

Elle dépend de la variance du bruit σ^2 qui est inconnue et des constantes c_1 et c_2 dont les valeurs optimales sont elles aussi inconnues. L'estimateur pénalisé $\tilde{s}$ dépend donc du choix de ces constantes et ses propriétés peuvent varier avec ces constantes. Dans ce chapitre, nous supposons que la variance σ^2 est connue et notre objectif est d'obtenir des valeurs pour les constantes c_1 et c_2 qui mènent à de "bons" estimateurs dans une majorité de situations.

Comme nous l'avons vu dans la sous-section 2.3.2, l'estimateur idéal est celui qui réalise le plus petit des risques de la collection $\{\hat{s}_m, m \in \mathcal{M}_n\}$, l'**oracle** que nous identifions et notons :

$$O(s, \mathcal{S}) = \inf_{m \in \mathcal{M}_n} \mathbb{E}_s \left[\|\hat{s}_m - s\|_n^2 \right],$$

où nous rappelons que $\mathcal{S}$ est la collection de modèles choisie. Un tel estimateur ne pouvait être utilisé comme estimateur de s puisqu'il dépend de s mais l'objectif de la sélection de modèle était de construire un estimateur qui a le même comportement que l'oracle. De ce point de vue, l'estimateur sélectionné $\tilde{s}$ est "bon" si son risque est de l'ordre de celui de l'oracle et il est donc naturel d'évaluer sa performance par la mesure du rapport suivant :

$$\frac{\mathbb{E}_s \left[\|\tilde{s} - s\|_n^2 \right]}{O(s, \mathcal{S})}. \tag{3.1.1}$$

Par suite, si l'estimateur sélectionné conduit à un rapport proche de 1, il sera alors un bon estimateur. Les constantes optimales c_1 et c_2 seront celles qui mène à ce résultat. Le point important ici est que nous souhaitons choisir des constantes optimales universelles, c'est-à-dire des constantes optimales pour toute fonction s et toute taille d'échantillon n. L'idée est de choisir les valeurs des constantes c_1 et c_2 qui minimisent le rapport de risques uniformément en s et n. Nous cherchons donc à majorer le rapport de risques par une constante indépendante de n et de s.

Nous verrons dans la sous-section 3.2.1 que le rapport de risques, défini en (3.1.1), est majoré par un terme qui n'est pas indépendant de la taille de l'échantillon n. Dans la sous-section 3.2.2, nous définissons l'oracle adéquat pour notre étude avant de s'attacher à la calibration des constantes. Pour choisir ensuite les constantes optimales c_1 et c_2, nous nous sommes inspirés des travaux de Birgé et Rozenholc [9] sur la calibration des constantes de pénalité dans le cadre de l'estimation de densité par histogrammes. Elles sont obtenues par une étude de simulations. La description de la procédure de simulations et les résultats sont présentés en section 3.3.

3.2 Définition de l'oracle

Cette section est consacrée à la définition de l'oracle. Nous expliquons pourquoi l'oracle défini classiquement en sélection de modèle n'est pas approprié, puis nous définissons un oracle pour notre cadre d'étude.

3.2.1 L'oracle classique

L'estimateur sélectionné est bon si son risque est de l'ordre du plus petit des risques des estimateurs de la collection, appelé oracle.

Definition 3.2.1. *Etant donné une collection de modèles linéaires $\mathcal{S} = \{\mathcal{S}_m\}_{m\in\mathcal{M}_n}$, l'oracle de la collection $\mathcal{S}$ pour la fonction s est défini par:*

$$O(s,\mathcal{S}) = \inf_{m\in\mathcal{M}_n} \mathbb{E}_s\left[\|\hat{s}_m - s\|_n^2\right], \tag{3.2.2}$$
$$= \inf_{m\in\mathcal{M}_n} \|s_m - s\|_n^2 + \frac{D_m}{n}.$$

D'après la majoration du risque, donnée en (2.4.13), il existe une constante C telle que l'estimateur sélectionné est comparable à l'oracle via l'inégalité

$$\mathbb{E}_s\left[\|\tilde{s} - s\|_n^2\right] \leqslant C\log(n)\ O(s,\mathcal{S}). \tag{3.2.3}$$

D'après cette majoration, le plus petit des risques des estimateurs de la collection $\{\hat{s}_m, m \in \mathcal{M}_n\}$ n'est atteint qu'à un $\log n$ près.

Le rapport du risque de l'estimateur pénalisé sur l'oracle est alors majoré par par

$$\frac{\mathbb{E}_s\left[\|\tilde{s}-s\|_n^2\right]}{O(s,\mathcal{S})} \leqslant C\log(n),$$

Nous remarquons que le rapport de risques est majoré par un terme dépendant de la taille de l'échantillon n, issu de la richesse de la collection de partitions que nous considérons. Quand n est grand, la majoration du rapport de risque devient grande, et cette comparaison n'a pas de sens pour obtenir des constantes optimales uniformément en n. Par conséquent, l'oracle classique défini en (3.2.2) ne peut être utilisé dans notre cadre d'étude.

Pour palier au problème de la dépendance en n de la majoration du rapport de risques, nous avons considéré le rapport suivant :

$$\frac{\mathbb{E}_s\left[\|\tilde{s}-s\|_n^2\right]}{\inf_{m\in\mathcal{M}_n}\left[\|s-s_m\|_n^2+pen_n(m)\right]}.$$

En substituant la fonction de pénalité par son expression donnée par l'équation (2.4.12), le rapport s'écrit

$$\frac{\mathbb{E}_s\left[\|\tilde{s}-s\|_n^2\right]}{\inf_{m\in\mathcal{M}_n}\left[\|s-s_m\|_n^2+\frac{D_m}{n}\sigma^2(c_1\log(\frac{n}{D_m})+c_2)\right]}.$$

Le dénominateur dépend des constantes c_1 et c_2. Par conséquent, plus les constantes c_1 et c_2 sont grandes, plus le rapport tend vers 0. En effet, la fonction de pénalité croît avec les constantes c_1 et c_2. Par conséquent, le risque de l'estimateur pénalisé deviendra constant (égal au risque de l'estimateur associé à la dimension 1 puisque nous sur-pénaliserons) et le dénominateur sera grand puisque le terme de biais sera négligeable devant les valeurs de la fonction de pénalité. La minimisation de ce rapport ne peut donc permettre l'accès à des constantes optimales.

3.2.2 L'oracle des modèles regroupés

Nous donnons ici l'oracle que nous avons choisi et la justification de ce choix.

Nous cherchons un oracle adapté à une collection de partitions très riche, dans laquelle le nombre de partitions de même dimension est élevé, ici C_{n-1}^{D-1} pour une dimension D. Par définition (3.2.1), l'oracle dépend de la vraie fonction s. Supposons un instant que s est connue, nous disposons alors de la dimension de la vraie partition. Le problème de sélection de modèles, donc de partitions peut se voir ici comme un problème de sélection de dimensions. Puisque la pénalité ne dépend de la partition que via sa dimension, choisir le meilleur estimateur parmi la collection d'estimateurs $\{\hat{s}_m, m\in\mathcal{M}_n\}$ revient à choisir le meilleur parmi la collection

$$\{\hat{s}_{\hat{m}_D}, D=1,\dots,n\},$$

où $\hat{m}_D$ est la meilleure partition de dimension D, définie par :

$$\hat{m}_D = \underset{m \in \mathcal{M}_n, |m|=D}{\operatorname{argmin}} \gamma(\hat{s}_m). \tag{3.2.4}$$

La meilleure dimension est définie par :

$$\hat{D} = \underset{D \geq 1}{\operatorname{argmin}} \left[\gamma_n(\hat{s}_{\hat{m}_D}) + \sigma^2 \frac{D}{n} \left(c_1 \log \frac{n}{D} + c_2 \right) \right].$$

Et l'estimateur $\tilde{s}$ sélectionné est défini par $\hat{s}_{\hat{m}_{\hat{D}}}$.

Si D est la dimension de la vraie partition, nous souhaitons comparer l'estimateur sélectionné au meilleur estimateur de dimension D, c'est-à-dire celui construit sur la meilleure partition de dimension D. Or le meilleur estimateur de dimension D n'est pas obligatoirement le meilleur sur toutes les dimensions au sens du risque. De ce point de vue, l'oracle qui semble approprié est l'estimateur qui a le plus petit risque "dimension par dimension", oracle appelé oracle des modèles regroupés. Nous donnons sa définition.

Definition 3.2.2. *Soit $\hat{s}_{\hat{m}_D}$, noté abusivement $\hat{s}_D$, l'estimateur du minimum de contraste sur le meilleur modèle de dimension D défini par :*

$$\begin{aligned} \hat{s}_D &= \underset{u \in S_D = \bigcup_{m \in \mathcal{M}_n, |m|=D} \mathcal{S}_m}{\operatorname{argmin}} \gamma(u), \\ &= \underset{m \in \mathcal{M}_n, |m|=D}{\operatorname{argmin}} \gamma(\hat{s}_m). \end{aligned}$$

Nous considérons l'oracle, dit "oracle des modèles regroupés", noté et défini par :

$$O_r(s, \mathcal{S}) = \inf_{D=1,\dots,n} \mathbb{E}_s \left[\|\hat{s}_D - s\|_n^2 \right]$$

Le rapport du risque $\mathbb{E}_s \left[\|\tilde{s} - s\|_n^2 \right]$ sur l'oracle $O_r(s, \mathcal{S})$ est contrôlé par une borne indépendante de s et n. Nous ne disposons pas de justifications théoriques pour cette considération mais elle a pu être vérifiée dans la pratique et s'explique par l'heuristique suivante : le regroupement préalable des partitions de même dimension nous conduit à un problème de sélection de partitions avec une partition par dimension. Dans ce cas, la pénalité est proportionnelle à la dimension (cf Birgé et Massart [7]). Elle n'inclut pas de partie logarithmique et le rapport de risques n'est plus majoré par un terme dépendant de n. Nous savons que la partie logarithmique de la pénalité obtenue résulte du nombre de partitions de même dimension. Donc heuristiquement, le $\log n$ est intrinsèque au regroupement effectué : il n'apparaît plus dans la majoration mais fait partie du terme $\mathbb{E}_s \left[\|\hat{s}_D - s\|_n^2 \right]$.

Remarque 3.2. La question qui se pose est la suivante : pourquoi ne pas regrouper préalablement les modèles de même dimension et considérer le problème de sélection de modèles à partir de la nouvelle famille de modèles $\{S_D = \cup_{m \in \mathcal{M}_n, |m|=D} \mathcal{S}_m, D = 1, \dots, n\}$?
La forme de la fonction de pénalité provient du contrôle du risque quadratique de l'estimateur pénalisé. Plus précisément, l'inégalité de concentration de Cirelson pour les modèles

Gaussiens est utilisée pour contrôler un certain terme autour de sa moyenne sur tous les modèles considérés (c'est de ce contrôle uniforme sur tous les modèles que sort la complexité de la collection de modèles qui apparaît alors dans la pénalité). En posant le problème comme un problème de selection de dimensions, cette démarche ne peut aboutir car les modèles $\{S_D, D = 1, \ldots, n\}$ ne sont pas linéaires et l'inégalité de concentration ne peut pas être utilisée.

3.3 Calibration des constantes de pénalité

Dans cette section, nous proposons une étude de simulations pour calibrer les constantes c_1 et c_2 de façon optimale. Nous reprenons la démarche suivie par Birgé et Rozenholc [9].

Dans la sous-section 3.3.1, nous décrivons la procédure de simulations pour une collection de fonctions $\mathcal{L}$ et de tailles d'échantillon $\mathcal{N}$. Dans la sous-section 3.3.2, nous donnons tout d'abord les paramètres de notre étude de simulations: les deux collections précédentes, et les différentes valeurs des constantes c_1 et c_2 considérées. Nous décrivons ensuite l'algorithme dynamique utilisé pour construire la collection des meilleures partitions $\{\hat{m}_D, D = 1, \ldots, n\}$.

3.3.1 Description de la procédure de simulations

Nous allons commencer par décrire la procédure de simulation qui consiste à calculer pour une fonction s et une taille d'échantillon n données, le rapport de risques défini par

$$F_n(s, c_1, c_2) = \frac{\mathbb{E}_s\left[\|\tilde{s}(c_1, c_2) - s\|_n^2\right]}{\inf_{D=1,\ldots,n} \mathbb{E}_s\left[\|\hat{s}_D - s\|_n^2\right]}$$

Les valeurs exactes des espérances $\mathbb{E}_s\left[\|\tilde{s}(c_1, c_2) - s\|_n^2\right]$ et $\mathbb{E}_s\left[\|\hat{s}_D - s\|_n^2\right]$ n'étant pas accessible analytiquement, elles seront estimées par une méthode de Monte Carlo en moyennant les valeurs $\|\tilde{s}(c_1, c_2) - s\|_n^2$ et $\|\hat{s}_D - s\|_n^2$ sur N_b échantillons simulés. Puis nous considérerons une collection de fonctions s et une collection de taille d'échantillon n, et nous donnerons la procédure qui permet de définir les constantes optimales.

Procédure pour une fonction s et une taille d'échantillon n données

Soit s une fonction et n une taille d'échantillon, nous simulons N_b échantillons de taille n suivant le modèle :

$$y_t = s(x_t) + \varepsilon_t \quad t = 1, \ldots, n$$

où ε est un bruit blanc gaussien de variance σ^2 donnée. Nous estimons s à partir de chaque échantillon :

nous regroupons les partitions de même dimension et nous sélectionnons la meilleure partition pour chaque dimension D, $\hat{m}_D$. Cette opération est réalisée à l'aide d'un algorithme dynamique décrit dans la sous-section 3.3.2 qui permet d'obtenir la collection des meilleurs partitions de dimension D, $D = 1, \ldots, n$, et les valeurs du contraste des estimateurs construits sur les meilleurs partitions :

$$\{\hat{m}_D, D = 1, \ldots, n\} \quad \text{et} \quad \{\gamma_n(\hat{s}_D), D = 1, \ldots, n\}.$$

Nous obtenons ainsi la collection d'estimateurs :

$$\{\hat{s}_D, D = 1, \ldots, n\},$$

où le meilleur estimateur de dimension D est défini par :

$$\hat{s}_D = \sum_{k=1}^{D} \bar{y}_k \mathbb{1}_{\hat{I}_k},$$

avec

$$\hat{m}_D = \bigcup_{k=1}^{D} \hat{I}_k = \bigcup_{k=1}^{D}]\hat{\tau}_{k-1}, \hat{\tau}_k].$$

Étant données deux constantes c_1 et c_2, la fonction de pénalité est définie pour une dimension D par :

$$pen_n(D) = \frac{\sigma^2}{n} D \left(c_1 \log \frac{n}{D} + c_2 \right).$$

Nous minimisons le critère pénalisé suivant sur $D = 1, \ldots, n$

$$crit_n(D) = \gamma_n(\hat{s}_D) + pen_n(D).$$

Nous obtenons ainsi la meilleure partition qui dépend des constantes c_1 et c_2 de la pénalité :

$$\hat{m}(c_1, c_2) = \hat{m}_{\hat{D}(c_1, c_2)}.$$

L'estimateur de la fonction s est alors :

$$\tilde{s}(c_1, c_2) = \hat{s}_{\hat{m}(c_1, c_2)}.$$

Nous obtenons N_b estimateurs $\tilde{s}(c_1, c_2)^{(1)}, \ldots, \tilde{s}(c_1, c_2)^{(N_b)}$ et N_b collections des meilleurs estimateurs de dimension $D \geq 1$, $\{\hat{s}_D^{(1)}, D = 1, \ldots, n\}, \ldots, \{\hat{s}_D^{(N_b)}, D = 1, \ldots, n\}$ à partir desquels nous estimons $\mathbb{E}_s \left[\|\tilde{s}(c_1, c_2) - s\|_n^2 \right]$ et $\mathbb{E}_s \left[\|\hat{s}_D - s\|_n^2 \right]$ pour tout $D \geq 1$ respectivement par :

$$\frac{1}{N_b} \sum_{i=1}^{N_b} \|\tilde{s}(c_1, c_2)^{(i)} - s\|_n^2 \quad , \quad \frac{1}{N_b} \sum_{i=1}^{N_b} \|\hat{s}_D^{(i)} - s\|_n^2.$$

Nous calculons alors le rapport :

$$F_n(s, c_1, c_2) = \frac{\mathbb{E}_s \left[\|\tilde{s}(c_1, c_2) - s\|_n^2 \right]}{\inf_{D=1,\dots,n} \mathbb{E}_s \left[\|\hat{s}_D - s\|_n^2 \right]} \tag{3.3.5}$$

Nous effectuons le calcul numérique du rapport $F_n(s, c_1, c_2)$ pour toutes les constantes c_1 et c_2 considérées dans un même temps.

En pratique, nous ne considérons pas les partitions de trop grande dimension par rapport à la vraie dimension de la partition, connue dans cette étude de simulation et supposée sur des données réelles. Nous notons D_{max} la valeur maximale de la dimension des partitions. Ainsi, par l'algorithme dynamique, nous obtenons la collection des meilleurs partitions et les valeurs du contraste des estimateurs pour les dimensions $D = 1, \dots, D_{max}$.

Procédure finale

Nous cherchons les constantes c_1 et c_2 optimales pour toute fonction s et toute taille d'échantillon n. La procédure de simulations se décompose en deux étapes :

1. nous fixons n et cherchons les valeurs optimales de c_1 et c_2 indépendemment de la fonction s. Pour cela, nous considérons un ensemble fini de fonctions, noté $\mathcal{L}$. Nous appliquons la procédure décrite précédemment pour chacune des fonctions. L'idée est de chercher les constantes c_1 et c_2 faisant au mieux pour toutes les fonctions dans $\mathcal{L}$. C'est pourquoi, nous considérons la valeur du rapport suivant :

   $$F_n(c_1, c_2) = \sup_{s \in \mathcal{L}} F_n(s, c_1, c_2) \tag{3.3.6}$$

   où $F_n(s, c_1, c_2)$ est défini pour toute fonction s et pour des constantes c_1 et c_2 par (3.3.5).

   Un bon choix des constantes c_1 et c_2 devra mener à des valeurs bornées de ce rapport.

2. nous effectuons la même procédure pour différentes valeurs de n. Notons N le vecteur des valeurs de n choisies.

Nous disposons maintenant d'une famille de valeurs

$$\{F_n(c_1, c_2), c_1, c_2 > 0, n \in N\}.$$

Le nombre de valeurs trop important ne permet pas d'extraire facilement les constantes optimales. Par conséquent, nous étudions cet ensemble comme des fonctions de n, c_1 et c_2. Plus précisément, afin de visualiser les résultats et de pouvoir les interpréter, nous fixons la constante c_2 et nous traçons les graphiques des fonctions

$$c_1 \to F_n(c_1, c_2),$$

pour les différentes valeurs de n. Ensuite, nous procédons à cette même étude pour différentes valeurs de c_2.

À c_2 et n fixés, la valeur optimale de c_1 est donnée par :

$$c_1^*(n, c_2) = \underset{c_1>0}{\operatorname{argmin}} F_n(c_1, c_2).$$

L'idée est de prendre comme valeur optimale de la constante c_2, la valeur qui rend stable $c_1^*(n, c_2)$ en n. Soit c_2^* cette valeur. Et la valeur de c_1 optimale sera alors égale à $c_1^* = c_1^*(n, c_2^*)$ quelque soit n.

3.3.2 Présentation des paramètres considérés et de l'algorithme dynamique

Les paramètres

Pour cette étude, nous avons considéré les ensembles de valeurs suivantes :

- différentes tailles d'échantillon $N = \{20, 50, 100, 300, 500, 1000, 5000\}$.
- une variance du bruit $\sigma^2 = 1$.
- 46 valeurs de c_1 et 110 de c_2 :
  - c_1 de 0 à 4 par pas de 0.1 et de 4 à 6 par pas de 0.5.
  - c_2 de 0 à 4 par pas de 0.5 et de 4 à 14 par pas de 0.1.
- 35 fonctions constantes par morceaux simulées aléatoirement. Pour chaque fonction, nous simulons
  - le nombre de segments $nseg = X + 1$ où X suit une loi de poisson de paramètre 5.
  - chaque instant de rupture τ_k, $k = 1, \ldots, nseg - 1$ où τ_k suit une loi uniforme sur l'intervalle $[0, 1]$.
  - chaque moyenne s_k, $k = 1, \ldots, nseg$ où s_k suit une loi normale de moyenne 0 et de variance 1.
- $D_{max} = 40$.
- $N_b = 250$.

Algorithme dynamique pour la construction de la famille $\{\hat{m}_D, D \geq 1\}$.

Pour une dimension D fixée, le nombre de choix possibles de partitions de dimension D sur la grille de $\{1, \ldots, n\}$ est C_{n-1}^{D-1}. Ainsi la recherche de la meilleure partition de dimension D, $\hat{m}_D$ défini par l'égalité (3.2.4) prend algorithmiquement $\mathcal{O}(n^D)$ opérations. Pour réduire le temps de programmation, nous proposons d'employer un algorithme dynamique. Cet algorithme est basé sur le fait que le contraste considéré est additif, ce qui est le cas dans notre étude. Il est décrit explicitement dans [37]. Nous en donnons ici un court résumé. En pratique, nous travaillons sur la grille $\{1, \ldots, n\}$ plutôt que sur $\{1/n, \ldots, 1\}$.

La meilleure partition de dimension D minimise en $t_1, t_2, \ldots, t_{D-1}$

$$\sum_{k=1}^{D} \sum_{t=t_{k-1}+1}^{t_k} (y_t - \overline{y}_k)^2.$$

La programmation de l'algorithme dynamique peut être développée de manière récursive de la façon suivante :
posons

$$\Delta(t_{k-1}+1 : t_k) = \sum_{t=t_{k-1}+1}^{t_k} (y_t - \overline{y}_k)^2,$$

et définissons pour $D < E \leqslant n$

$$I_D(E) = \min_{t_0=0<t_1<t_2<\ldots<t_{D-1}<t_D=E} \sum_{k=1}^{D} \Delta(t_{k-1}+1 : t_k).$$

$\Delta(t_{k-1}+1 : t_k)$ est l'erreur quadratique sur le segment $[t_{k-1}+1, t_k]$ et $I_D(E)$ correspond donc à l'erreur minimale quadratique pour D segments (soit $D-1$ ruptures) entre 1 et E.

$I_D(E)$ se décompose facilement en

$$I_D(E) = \min_{t_{D-1}} \{I_{D-1}(t_{D-1}) + \Delta(t_{D-1}+1 : E)\}.$$

Cet algorithme peut être vu comme la recherche du chemin le moins coûteux pour aller d'un point à un autre. Un chemin correspond à une partition de $\{1, \ldots, n\}$ et son coût total est la valeur du contraste non renormalisé calculé sur cette partition. Les coûts de tous les segments envisagés sont préalablement calculés : les $\Delta(i : j)$ pour $1 \leqslant i \leqslant n$ et $i+1 \leqslant j \leqslant n$. Tous les $I_1(E)$ pour $E = 2, \ldots, n$ sont donc disponibles. Ensuite la grille $\{1, \ldots, n\}$ est balayée : nous calculons $I_2(E) = \min_r \{I_1(r) + \Delta(r+1 : E)\}$ pour $E = 3, \ldots, n$. Les valeurs I_1 étant déjà déterminées, cette recherche s'avère très rapide. Et ainsi de suite, nous calculons à chaque étape les "I_D'' que nous réutilisons à l'étape suivante.

La programmation dynamique ne nécessite donc pas l'évaluation de toutes les configurations considérées et son utilisation permet de réduire considérablement le temps de programmation puisque sa complexité algorithmique est $\mathcal{O}(n^2)$.

3.3.3 Résultats

Dans cette sous-section, nous analysons les résultats des simulations obtenus pour définir les constantes optimales c_1^* et c_2^*.

Les graphes des fonctions

$$c_1 \to F_n(c_1, c_2),$$

sont représentés respectivement sur les Figures 3.1, 3.2 et 3.3 pour $c_2 = 0, 5, 8$ et $n = 20, 50, 100, 300, 500, 1000, 5000$. Par exemple, sur la Figure 3.1 sont dessinées les fonctions $c_1 \to F_n(c_1, 0)$ pour les différentes valeurs de n.
Nous observons différents résultats :

- Nous évaluons le minimum des fonctions $F_n(c_1, c_2)$ à c_2 fixé, $c_1^*(n, c_2)$, et nous discutons selon les différentes valeurs de n :

  - le minimum $c_1^*(n, c_2)$ varie lentement avec c_2 et n.

  - pour $c_2 = 0$, le minimum $c_1^*(n, 0)$ décroît avec n. Nous remarquons que $c_1^*(n, 0)$ dépend clairement de n.

  - pour $c_2 = 8$, le minimum $c_1^*(n, 8)$ a tendance à croître avec n. Nous observons ce phénomène à partir de environ $c_2 = 6.4$.

  - pour des valeurs de c_2 assez proche de 5 ($4.8 \leqslant c_2 \leqslant 6$), le minimum $c_1^*(n, c_2)$ semble assez stable et sa valeur est proche de 2 (respectivement entre 2.1 et 1.7) quelque soit n.

- Nous savons que si le rapport $F_n(s, c_1, c_2)$ est proche de 1, alors l'estimateur considéré sera bon. D'après les graphiques, quelque soit les valeurs c_2 et n, $F_n(c_1^*(n, c_2), c_2)$ est plus petit que 2, et la plupart du temps ($n \geq 50$) par 1.7.

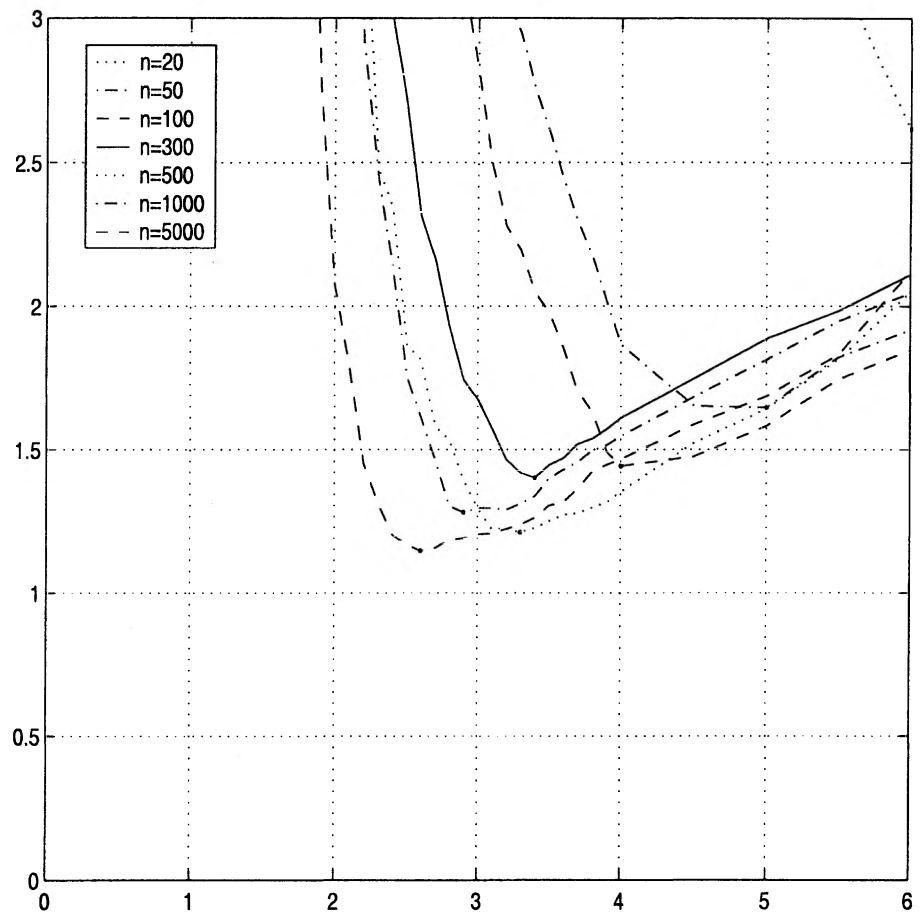


FIG. 3.1: *Graphe de la fonction* $c_1 \to F_n(c_1, 0)$

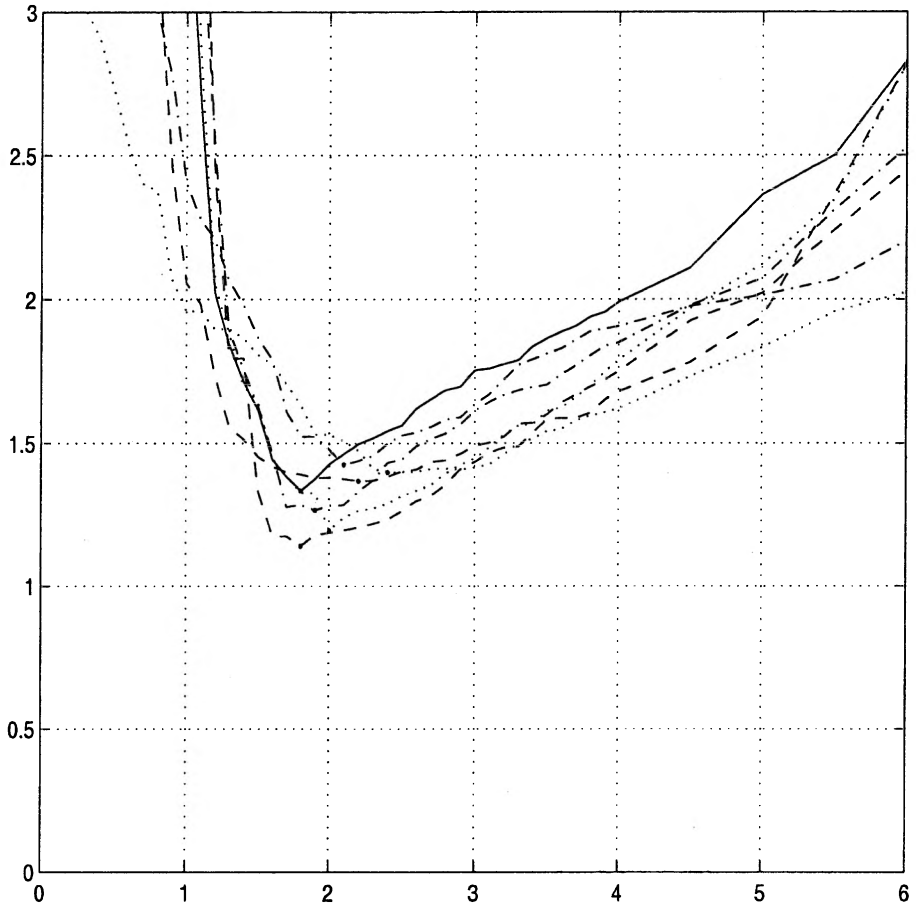


FIG. 3.2: *Graphe de la fonction* $c_1 \to F_n(c_1, 5)$

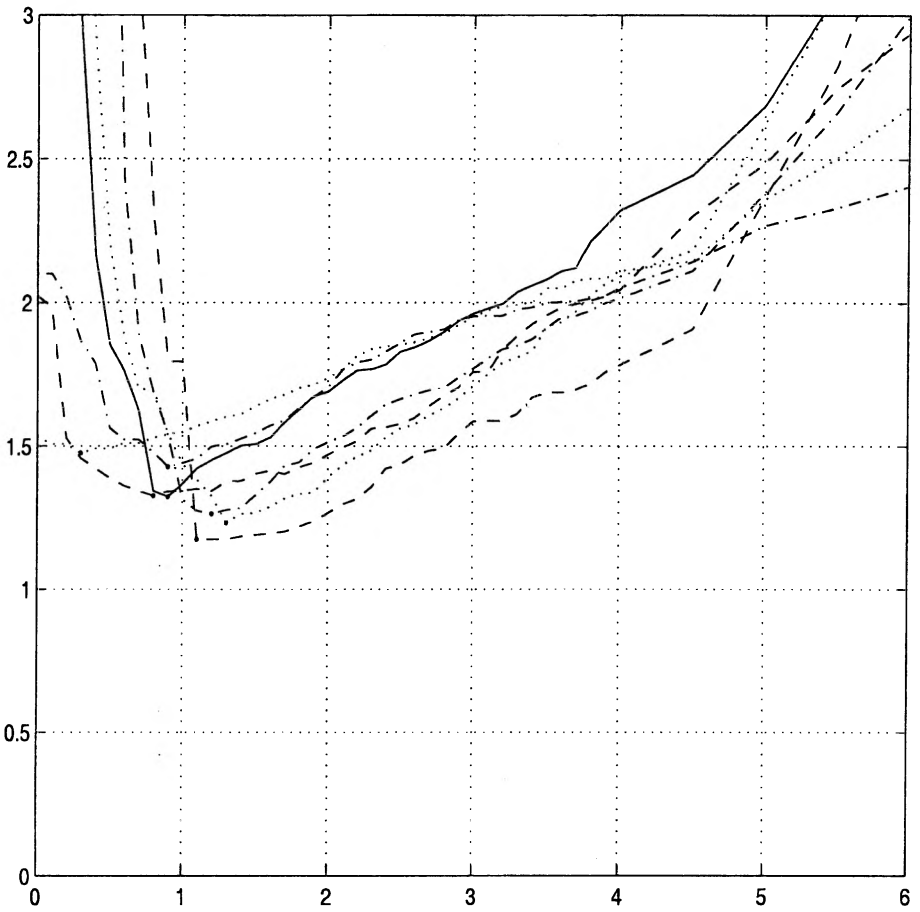


FIG. 3.3: *Graphe de la fonction* $c_1 \to F_n(c_1, 8)$

Un choix précis de c_2 s'avère être un peu plus délicat que le choix de c_1. Plusieurs valeurs de c_2 semble stabiliser le rapport considéré en n : $c_2 \in [5, 6]$.

Nous proposons une lecture graphique différente. En Figures 3.4 et 3.5, nous représentons les fonctions suivantes par lignes de niveaux pour les différentes valeurs de n :

$$(c_1, c_2) \to F_n(c_1, c_2).$$

Par exemple, pour $n = 100$, la zone intérieure associée à 1.38 signifie que la fonction $F_n(c_1, c_2)$ est inférieure à 1.38 pour toutes les valeurs de c_1 (en absisse) et c_2 (en ordonnée) associées. Rappelons que nous cherchons des constantes optimales universelles dans le sens où elles minimisent le rapport $F_n(c_1, c_2)$ pour tout n. Nous remarquons que (2,5) fait un "bon compromis" dans le sens où ce couple de valeurs appartient à toutes les zones "minimales", plus précisément aux ensembles $\{(c_1, c_2)/F_n(c_1, c_2) < 1.6\}$ pour tout n.
Notre choix s'est porté sur les constantes optimales suivantes :

$$(c_1^*, c_2^*) = (2, 5).$$

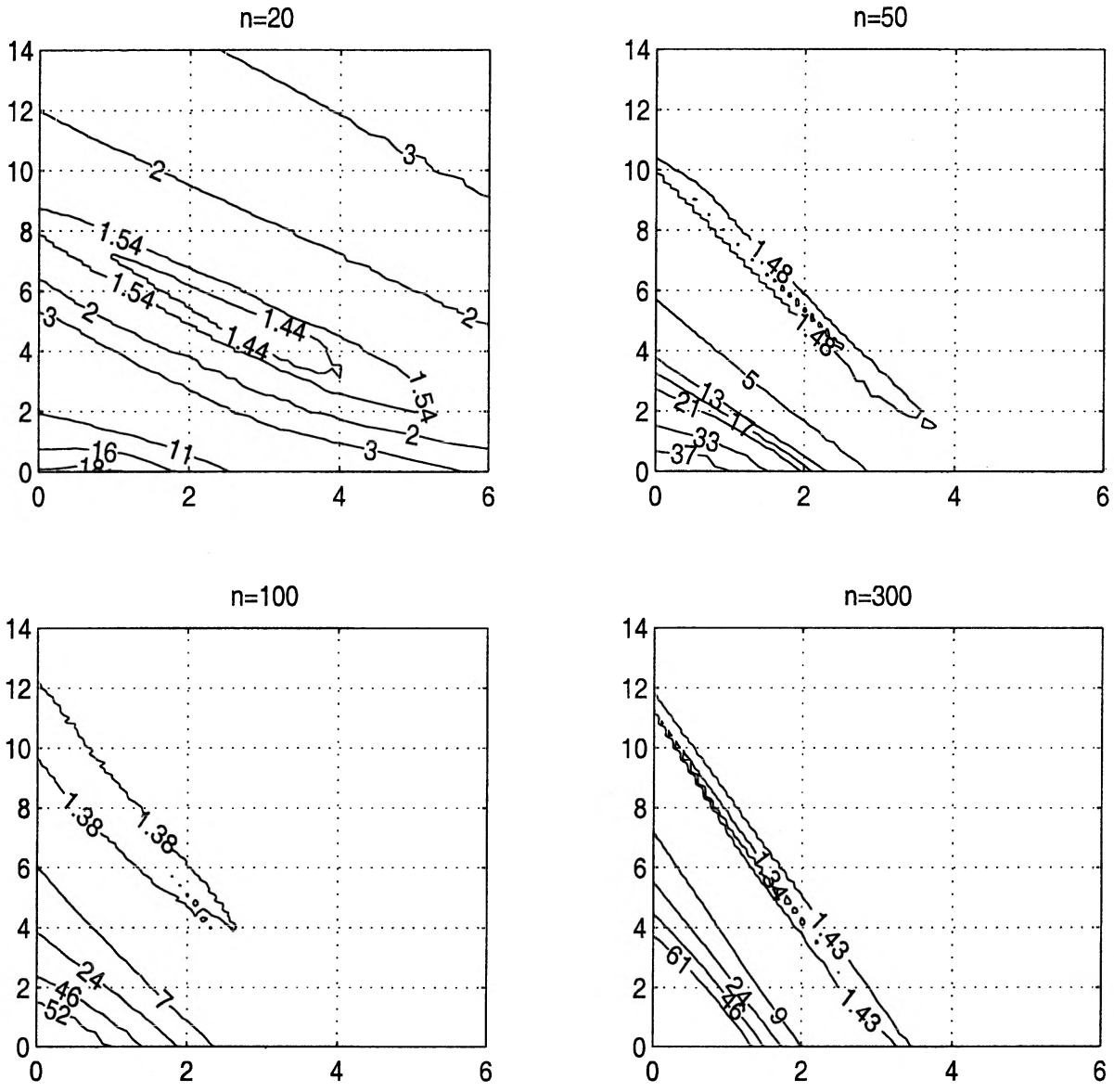


FIG. 3.4: *Graphe de la fonction* $(c_1, c_2) \to F_n(c_1, c_2)$ *pour* $n = 20, 50, 100, 300$.

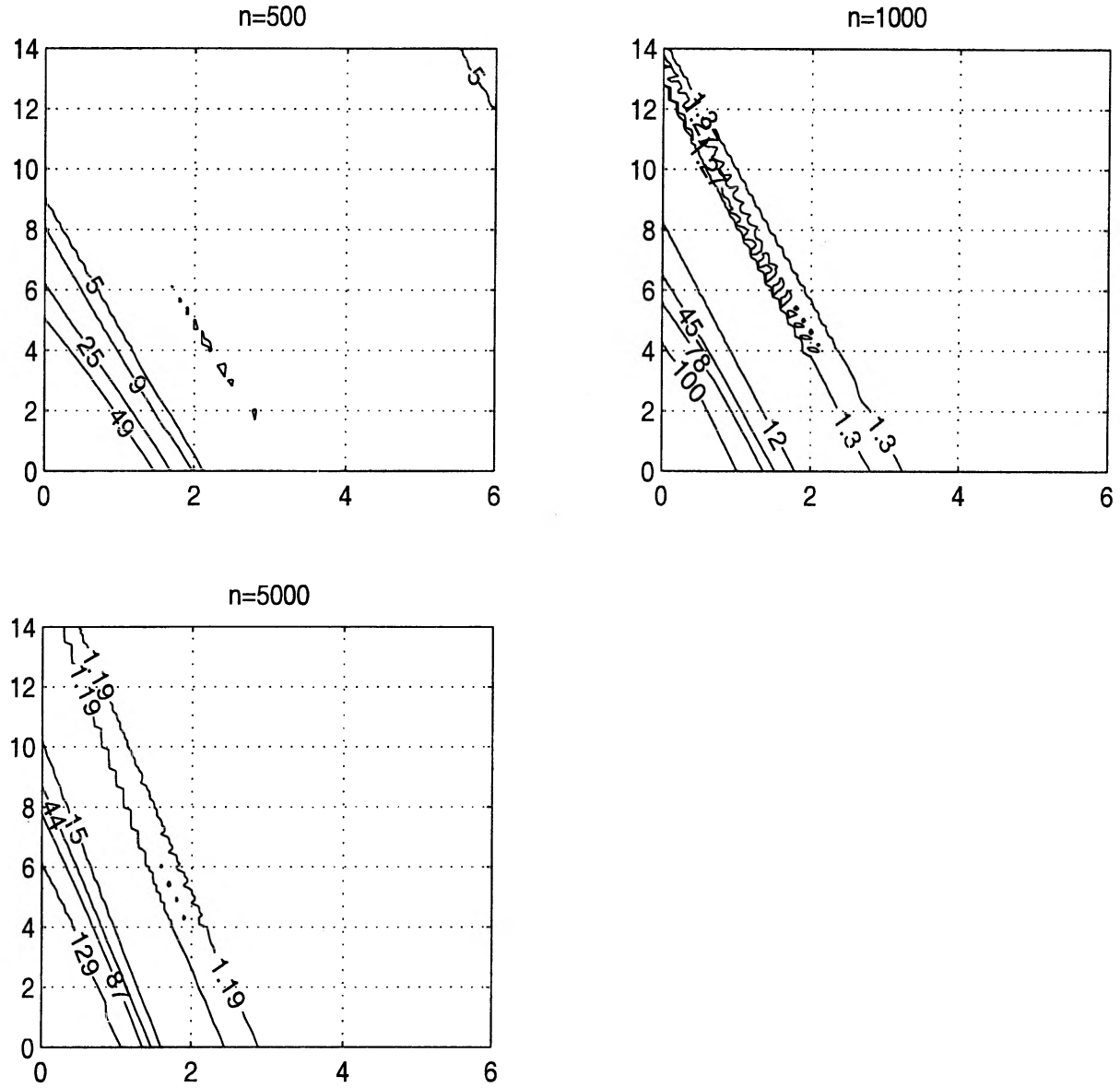


FIG. 3.5: *Graphe de la fonction* $(c_1, c_2) \to F_n(c_1, c_2)$ *pour* $n = 500, 1000, 5000$.

A titre indicatif, nous donnons en Figure 3.6 est une représentation graphique par lignes de niveaux de la fonction

$$(c_1, c_2) \to \max_n |F_n(c_1, c_2) - \min_{c_1, c_2 > 0} F_n(c_1, c_2)|. \tag{3.3.7}$$

A n fixé, les constantes optimales que l'on souhaiterait choisir sont

$$(c_1^*(n), c_2^*(n)) = \underset{c_1,c_2>0}{\operatorname{argmin}} F_n(c_1, c_2).$$

A c_1 et c_2 fixés, la fonction, définie en (3.3.7), donne une idée de l'écart entre $F_n(c_1, c_2)$ et $F_n(c_1^*(n), c_2^*(n))$ sur tous les n. Pour $(c_1^*, c_2^*) = (2, 5)$, la distance est minimale (inférieure à 0.14), très proche de la valeur minimale de la fonction définie en (3.3.7) qui est 0.115. Ce graphe n'est à prendre qu'à titre indicatif car pour les constantes c_1 et c_2 qui nous intéressent, le maximum semble être réalisé que pour les n petits. Ainsi, nous ne prenons pas en considération les grands échantillons.

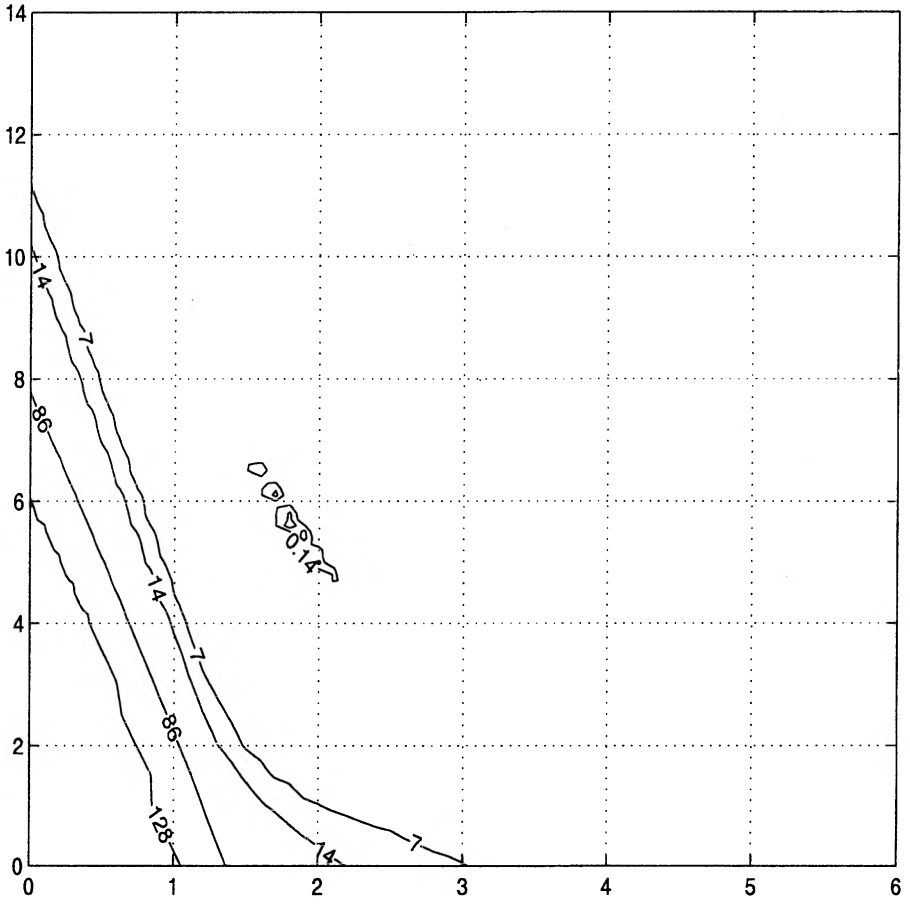


FIG. 3.6: *Graphe de la fonction* $(c_1, c_2) \to \max_n |F_n(c_1, c_2) - \min_{c_1,c_2>0} F_n(c_1, c_2)|$.

3.3.4 Interprétation numérique des résultats

Dans cette sous-section, nous analysons numériquement les résultats des simulations obtenus.

Nous effectuons, à n fixé, les 2 comparaisons suivantes pour mesurer la qualité de notre choix des constantes 2 et 5.

1. Nous considérons le rapport $F_n(c_1, c_2)$, donné en (3.3.6), et nous comparons :
$$F_n(c_1^*(n), c_2^*(n)) \quad \text{et} \quad F_n(2,5).$$

2. Nous étudions $F_n(s, c_1, c_2)$, donné en (3.3.5), en moyenne sur la collection de fonctions $\mathcal{L}$. Soit $N_{\mathcal{L}}$ la taille de la collection $\mathcal{L}$, nous comparons :
$$MF_n = \frac{1}{N_{\mathcal{L}}} \sum_{i=1}^{N_{\mathcal{L}}} F_n(s^{(i)}, c_1^*(n, s^{(i)}), c_2^*(n, s^{(i)})) \quad \text{et} \quad MF_n^{(2,5)} = \frac{1}{N_{\mathcal{L}}} \sum_{i=1}^{N_{\mathcal{L}}} F_n(s^{(i)}, 2, 5).$$
où pour une fonction s,
$$(c_1^*(n, s), c_2^*(n, s)) = \underset{c_1, c_2 > 0}{\operatorname{argmin}} F_n(s, c_1, c_2),$$
avec $F_n(s, c_1, c_2)$ défini par (3.3.5).

Nous proposons ensuite d'estimer le rapport $F_n(s, c_1, c_2)$ en enlevant 5% des valeurs les plus abérantes pour l'estimation de $\mathbb{E}_s\left(\|\hat{s}_{\hat{m}(c_1,c_2)} - s\|_n^2\right)$:
$$F_{(n,q)}(s, c_1, c_2) = \frac{\mathbb{E}_s\left[\|\hat{s}_{\hat{m}(c_1,c_2)} - s\|_n^2 \mathbb{1}_{\{\|\hat{s}_{\hat{m}(c_1,c_2)} - s\|_n^2 < q_{0.95}\}}\right]}{\inf_{D=1,\dots,n} \mathbb{E}_s\left[\|\hat{s}_D - s\|_n^2\right]}$$
où $q_{0.95}$ est le quantile empirique d'ordre 0.95 des réalisations $\|\hat{s}_{\hat{m}(c_1,c_2)} - s\|_n^2$.

Nous réalisons la même étude comparative que précédemment : nous considérons $F_{(n,q)}$ et nous calculons les différentes valeurs $F_{(n,q)}(c_1^*(n), c_2^*(n))$, $F_{(n,q)}(2,5)$, $MF_{(n,q)}$ et $MF_{(n,q)}^{(2,5)}$ pour tout n.

Les tableaux 3.1 et 3.2 donne respectivement les résultats des deux rapports considérés pour les différentes valeurs de n.

	$F_n(c_1^*(n), c_2^*(n))$	$F_n(2,5)$	MF_n	$MF_n^{(2,5)}$
$n = 20$	1.397	1.53	1.0654	1.213
$n = 50$	1.424	1.477	1.0588	1.17
$n = 100$	1.3	1.379	1.0657	1.142
$n = 300$	1.31	1.426	1.0588	1.127
$n = 500$	1.185	1.193	1.022	1.082
$n = 1000$	1.26	1.28	1.026	1.077
$n = 5000$	1.132	1.186	1.024	1.048

TAB. 3.1: *Estimation de $F_n(c_1^*(n), c_2^*(n))$, $F_n(2,5)$, MF_n et $MF_n^{(2,5)}$ pour les différentes valeurs de n.*

	$F_{(n,q)}(c_1^*(n), c_2^*(n))$	$F_{(n,q)}(2,5)$	$MF_{(n,q)}$	$MF_{(n,q)}^{(2,5)}$
$n = 20$	1.227	1.266	0.935	1.064
$n = 50$	1.2	1.3	0.958	1.047
$n = 100$	1.186	1.264	0.959	1.028
$n = 300$	1.198	1.36	0.958	1.017
$n = 500$	1.073	1.106	0.917	0.964
$n = 1000$	1.13	1.19	0.926	0.965
$n = 5000$	1.043	1.104	0.919	0.939

TAB. 3.2: *Estimation de* $F_{(n,q)}(c_1^*(n), c_2^*(n))$, $F_{(n,q)}(2,5)$, $MF_{(n,q)}$ *et* $MF_{(n,q)}^{(2,5)}$ *pour les différentes valeurs de* n.

- Les différents rapports étudiés tendent vers 1 avec n. Cela signifie que plus n est grand plus l'estimateur sélectionné est proche de l'oracle.
- Les rapports ont tendance à être très proches de ceux estimés en $(2,5)$. Par exemple, la différence des rapports $F_n(c_1^*(n), c_2^*(n))$ et $F_n(2,5)$ est inférieure à 0.132 pour tout n.
- Le rapport $F_{(n,q)}(s, c_1^*(s,n), c_2^*(s,n))$ est, en moyenne sur les fonctions s, inférieure à 1. Cela sous-entend que pour une fonction s donnée, l'estimateur sélectionné peut faire mieux que l'oracle en terme de risque.

Tous ces résultats nous conforte dans notre choix des constantes optimales $c_1^* = 2$ et $c_2^* = 5$. De plus, ils confirment que l'oracle des modèles regroupés est l'oracle adéquat dans notre cadre d'étude.

Chapitre 4

Utilisation et calibration d'une méthode heuristique

4.1 Introduction

Dans le chapitre 2, nous avons supposé que la variance du bruit était un paramètre connu afin de calibrer les constantes de la fonction de pénalité de façon optimale. Nous supposons maintenant que cette variance est inconnue. Plutôt que d'utiliser un estimateur de la variance, nous considérons la variance comme une constante, et nous cherchons à estimer la fonction de pénalité à partir des données. Nous écrivons la fonction de pénalité sous la forme générale suivante :

$$pen(D) = \beta f_n(D) \quad \text{pour tout } D \geq 1, \tag{4.1.1}$$

où f_n est une fonction convenablement choisie. Le critère pénalisé est alors défini par :

$$crit_n(D) = \gamma_n(\hat{s}_D) + \beta f_n(D).$$

L'objectif est d'obtenir une bonne valeur de β pour pénaliser correctement.

Il existe des méthodes de type $L - curve$ qui sont particulièrement étudiées depuis quelques années dans les problèmes inverses pour le choix d'un paramètre de régularisation. Nous renvoyons aux travaux de [34], [35], [25], [33], [64], [61]. La méthode $L - curve$ est une méthode graphique. Son nom dérive du comportement du graphe de $(f_n(D), \gamma_n(\hat{s}_D))$ pour différentes valeurs de D en forme de "L". Heuristiquement, le point près du "coin" du "L" représente la dimension à partir de laquelle $\gamma_n(\hat{s}_D)$ varie peu en fonction de $f_n(D)$. Cela signifie que choisir une dimension supérieure n'est pas "nécessaire". Quand le graphe ne montre pas une telle forme, il devient impossible de choisir la "bonne" dimension.

Dans ce chapitre, nous considérons une méthode proposée par Birgé et Massart [8] qui consiste à trouver la bonne pénalité à partir des données. Elle repose sur une heuristique

et sur des résultats théoriques obtenus exclusivement dans un contexte de sélection de modèle. Elle est employée dans [45] pour l'ajustement d'une fonction de régression dans le modèle de Cox. Notre objectif est de mettre en oeuvre cette méthode et de la calibrer dans notre cadre d'étude.

Dans la section 4.2, nous présentons l'heuristique sur laquelle se base la méthode. L'idée est d'utiliser $\gamma_n(\hat{s}_D)$ dans les grandes dimensions, là où il est de l'ordre de l'opposé du terme de pénalité. Dans la section 4.3, nous montrons à l'aide d'une étude de simulation qu'il peut être difficile d'appliquer cette heuristique. Dans [8], Birgé et Massart proposent une extension dont nous présentons l'heuristique dans la section 4.4. Nous décrivons dans la section 4.5 les différents types de problèmes qui se sont présentés lors de l'application de la méthode sur des données simulées. Notre objectif final étant de calibrer la méthode afin d'en obtenir une version automatique et exploitable dans une majorité de situations. Une étude de simulation est proposée dans la section 4.6 pour mesurer la performance de la méthode calibrée. Nous la comparons avec la méthode de sélection de modèle à variance connue et estimée (il suffit de considérer la pénalité dans laquelle la variance est substituée par un de ses estimateurs). Nous présentons dans les sous-sections 4.6.3 et 4.6.4 respectivement les résultats obtenus avec des processus non constants par morceaux, puis des processus non Gaussiens. Nous discutons ensuite dans la sous-section 4.6.5 de la méthode non calibrée. Enfin, dans la sous-section 4.7 nous appliquons la méthode pour détecter des ruptures dans le nombre mensuel de tests HIV en France sur plusieurs années.

4.2 Heuristique

Supposons que nous disposons d'une collection de partitions $\mathcal{M}_n$ avec une partition par dimension : nous pouvons identifier $\mathcal{M}_n$ à l'ensemble d'entiers $\{1, 2, \ldots, n\}$. L'heuristique est basée d'une part sur celle du C_p de Mallows rappelée dans la section 2.5 du chapitre 1, et d'autre part sur le fait que le contraste d'un estimateur se décompose en la somme de deux termes : un premier qui représente une erreur d'approximation du modèle associé, un terme de biais, et un second qui représente une erreur d'estimation, un terme de variance, qui représente l'opposé de la moitié de la fonction pénalité. L'idée est de considérer que le terme de biais s'annule dans les modèles de grandes dimensions. Ainsi le contraste de l'estimateur d'un tel modèle représentera une estimation du terme de pénalité.

L'heuristique se résume par les deux étapes suivantes :

1. L'heuristique de Mallows mène au critère défini pour tout $D \in \mathcal{M}_n$ par :

   $$crit_{Cp}(D) = \gamma_n(\hat{s}_D) + 2\mathbb{E}_s\left[\|\hat{s}_D - s_D\|_n^2\right]. \tag{4.2.2}$$

   D'autre part, nous écrivons la pénalité sous la forme générale suivante

   $$pen(D) = \beta f_n(D) \quad \text{pour tout } D \geq 1,$$

   où f_n est une fonction de la dimension de la partition convenablement choisie.

Le critère pénalisé est défini pour tout $D \in \mathcal{M}_n$ par :

$$\begin{aligned} crit_2(D) &= \gamma_n(\hat{s}_D) + \beta f_n(D) \\ &= \gamma_n(\hat{s}_D) + 2\alpha f_n(D). \end{aligned} \tag{4.2.3}$$

Par identification des deux critères précédents donnés respectivement en (4.2.2) et (4.2.3), nous avons

$$\mathbb{E}_s\left[\|\hat{s}_D - s_D\|_n^2\right] = \alpha f_n(D). \tag{4.2.4}$$

2. La constante α est inconnue et l'objectif de cette seconde étape est de l'estimer.

   Pour une partition fixée, notée abusivement $D \in \mathcal{M}_n$, nous avons

$$\begin{aligned} \mathbb{E}_s\left[\gamma_n(\hat{s}_D) - \gamma_n(s)\right] &= \mathbb{E}_s\left[\gamma_n(s_D) - \gamma_n(s)\right] + \mathbb{E}_s\left[\gamma_n(\hat{s}_D) - \gamma_n(s_D)\right] \\ &= \|s_D - s\|_n^2 - \mathbb{E}_s\left[\|\hat{s}_D - s_D\|_n^2\right]. \end{aligned}$$

   D'après l'égalité (4.2.4), nous obtenons

$$\mathbb{E}_s\left[\gamma_n(\hat{s}_D)\right] = \|s_D - s\|_n^2 - \alpha f_n(D) + \mathbb{E}_s\left[\gamma_n(s)\right].$$

   L'idée de l'heuristique à dire que dans les modèles de grandes dimensions, le terme de biais, *i.e.* $\|s_D - s\|_n^2$, est proche de zéro. Donc puisque le terme $\mathbb{E}_s\left[\gamma_n(s)\right]$ est constant, si $\gamma_n(\hat{s}_D)$ est concentré autour de son espérance sur tous les modèles, *i.e.* si $\gamma_n(\hat{s}_D)$ est de l'ordre de $\mathbb{E}_s\left[\gamma_n(\hat{s}_D)\right]$, alors $\gamma_n(\hat{s}_D)$ sera de l'ordre de $-\alpha f_n(D)$. L'estimation de la pente $-\hat{\alpha}$ de la courbe de $\gamma_n(\hat{s}_D)$ en fonction de $f_n(D)$ seraît un estimateur de $-\alpha$ et donneraît $\hat{\alpha} f_n(D)$ comme estimateur de $\alpha f_n(D)$.

Finalement, d'après l'équation (4.2.3), la bonne pénalité est définie pour tout $D \in \mathcal{M}_n$ par :

$$pen_n(D) = 2\hat{\alpha} f_n(D), \tag{4.2.5}$$

et le critère est défini pour tout $D \in \mathcal{M}_n$ par :

$$crit_n(D) = \gamma_n(\hat{s}_D) + 2\hat{\alpha} f_n(D).$$

Remarque 4.3. Si la collection de partitions $\mathcal{M}_n$ possède plus d'une partition par dimension et si la fonction de pénalité dépend de la partition via sa dimension, alors nous obtenons facilement la collection de partitions désirée en construisant $\{\hat{m}_D, D \geq 1\}$ (comme vu dans la sous-section 3.2.2 du chapitre précédent). Cette collection est aléatoire et $\mathbb{E}_s\left[\gamma_n(s_D) - \gamma_n(s)\right]$ ne peut être égal à $\|s_D - s\|_n^2$. En pratique, il suffit de vérifier par simulation que $\mathbb{E}_s\left[\gamma_n(s_D) - \gamma_n(s)\right]$ est de l'ordre de $\mathbb{E}_s\left[\|s_D - s\|_n^2\right]$.

4.3 Application

Dans cette section, nous appliquons l'heuristique précédente pour savoir si elle permet d'obtenir une bonne pénalité. Pour cela, il faut tout d'abord vérifier que $\gamma_n(\hat{s}_D)$ est une fonction affine de $f_n(D)$ dans les grandes dimensions. Nous montrons dans la sous-section 4.3.1 à l'aide de trois configurations différentes que $\gamma_n(\hat{s}_D)$ n'est pas toujours une fonction affine de $f_n(D)$ et que le problème pour appliquer l'heuristique est le choix des dimensions pour estimer la pente. Au vu de ces constatations, dans la sous-section 4.3.2, nous proposons une étude de simulation pour étudier l'influence des dimensions choisies (pour estimer la pente) sur l'estimateur sélectionné. Une seconde étude est effectuée en même temps pour savoir si l'heuristique fonctionne dans le cas où $f_n(D)$ possède des constantes à estimer.

4.3.1 Trois configurations différentes

Dans cette sous-section, nous étudions le comportement du contraste $\gamma_n(\hat{s}_D)$ en fonction de $f_n(D)$ sur trois cas très différents qui reflètent trois configurations bien particulières.

Nous choisissons tout d'abord la fonction $f_n(D)$ dans notre cadre d'étude. Comme nous l'avons vu dans le chapitre précédent, la pénalité est définie pour tout $D \geq 1$ par :

$$pen_n(D) = \sigma^2 \frac{D}{n}\left(2\log\frac{n}{D} + 5\right).$$

Nous posons :

$$\alpha = \sigma^2,$$

et

$$f_n(D) = \frac{D}{n}\left(\log\frac{n}{D} + 2.5\right) \quad \text{pour tout } D \geq 1.$$

Nous définissons ensuite les paramètres que nous avons choisi pour simuler les trois configurations. Nous prenons $\sigma^2 = 1$. Considérons une fonction g appartenant au modèle de dimension 6 de la famille :

$$g(x) = \begin{cases} -0.87 & \text{si } 0 < x \leqslant 0.083 \\ 0.26 & \text{si } 0.083 < x \leqslant 0.266 \\ -0.87 & \text{si } 0.266 < x \leqslant 0.433 \\ 0.79 & \text{si } 0.433 < x \leqslant 616 \\ -0.04 & \text{si } 0.616 < x \leqslant 0.8 \\ -0.65 & \text{si } 0.8 < x \leqslant 1 \end{cases}$$

La taille de l'échantillon et la fonction s sont choisis afin de rencontrer des cas où :

- (a) : les ruptures sont difficiles à détecter et le nombre d'observations est petit : $n = 60$ et $s = g$;
- (b) : les ruptures sont difficiles à détecter et le nombre d'observations est grand : $n = 300$ et $s = g$;

- (c) : les ruptures sont détectables à l'oeil et le nombre d'observations est petit : $n = 60$ et $s = 3g$.

Nous simulons trois réalisations $y = (y_1, \ldots, y_n)$ à partir des paramètres considérés dans les trois cas, notées respectivement $y(a)$, $y(b)$ et $y(c)$. Les fonctions s choisies et ces réalisations sont représentées Figure 4.1, et seront utilisées tout au long de l'étude à titre d'exemple. Pour chaque réalisation, nous construisons la collection $\{\hat{m}_D, D \geq 1\}$ par l'algorithme dynamique (décrit dans la sous-section 3.3.2 du chapitre précédent) qui donne les valeurs des constrates $\gamma_n(\hat{s}_D)$ pour $D \geq 1$. Le graphe des valeurs $(f_n(D), \gamma_n(\hat{s}_D))$ pour $D = 1, \ldots, 40$ obtenues pour les trois réalisations est dessiné respectivement Figures $4.2 - (a)$, $4.2 - (b)$ et $4.2 - (c)$.

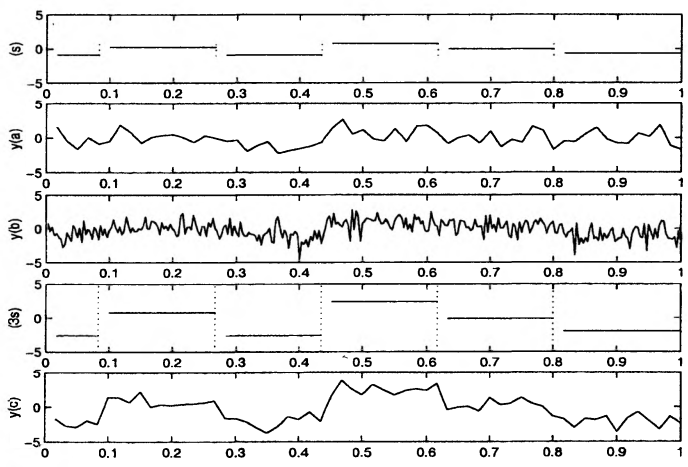


FIG. 4.1: *Représentations des fonctions s choisies et des trois réalisations simulées.*

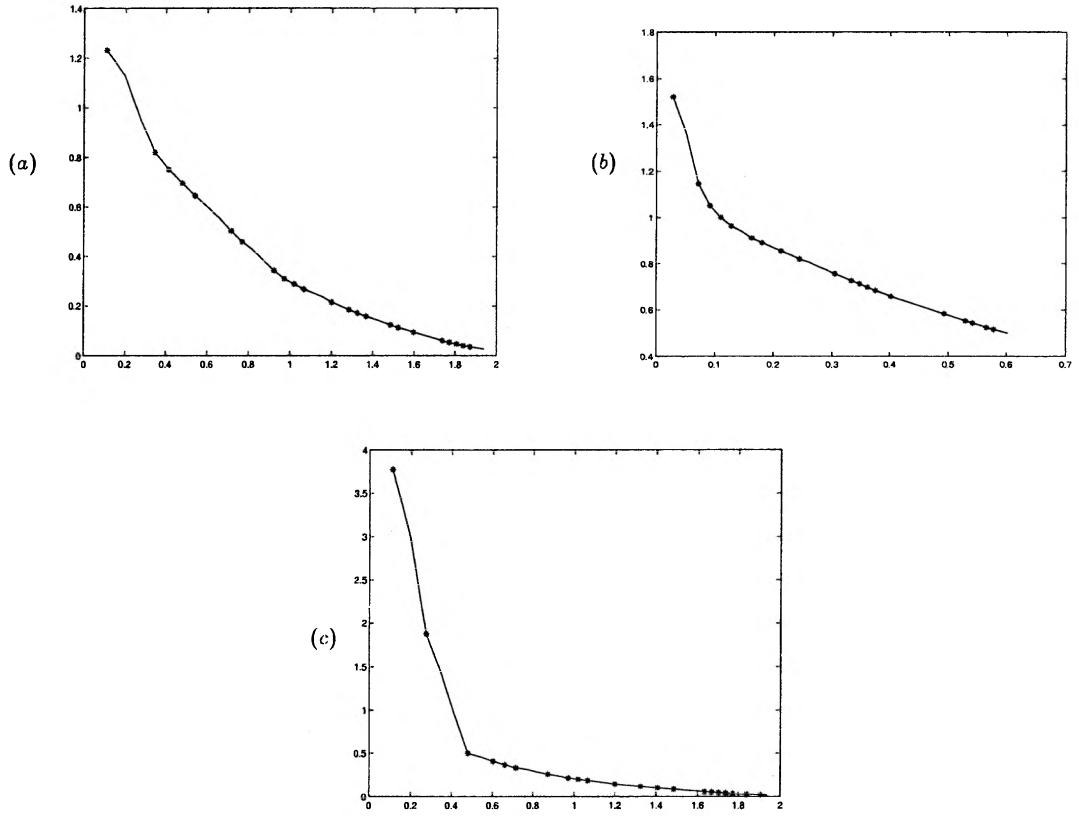


FIG. 4.2: *Représentation de* $(f_n(D), \gamma_n(\hat{s}_D))$ *pour* $D = 1, \ldots, 40$ *pour les trois réalisations.*

Nous déduisons de ces graphiques les remarques suivantes :

(*a*) Lorsque les ruptures sont difficilement détectables, pour des petits échantillons, $\gamma_n(\hat{s}_D)$ a plutôt un comportement logarithmique en $f_n(D)$. Ce comportement peut s'expliquer par le fait que le bruit se sépare difficilement du signal. Il est clair que la pente de la courbe $\gamma_n(\hat{s}_D)$ en fonction de $f_n(D)$ entre les dimensions 5 et 15 ($f_n(5) \simeq 0.34$ et $f_n(15) \simeq 1$) n'a pas la même valeur que celle entre les dimensions 16 et 38 ($f_n(16) \simeq 1$ et $f_n(38) \simeq 1.9$). En appliquant l'heuristique avec ces deux pentes, les partitions sélectionnées pourront être de dimensions différentes.

(*b*) Pour un nombre d'observations élevé, $\gamma_n(\hat{s}_D)$ est une fonction affine de $f_n(D)$ pour des dimensions supérieures à 6.

(*c*) Lorsque les ruptures sont très nettes, le graphe présente une "cassure" nette (associée à la dimension 5). Le contraste à partir de cette dimension varie peu : nous ne "gagnons" pas à choisir une dimension supérieure. Notons que dans ce type de configuration, un choix graphique tel que la méthode $L - curve$ [34] mène à la sélection d'un bon estimateur (celui de dimension 5).

En conclusion, pour une grande taille d'échantillon, $\gamma_n(\hat{s}_D)$ est une fonction affine de $f_n(D)$ pour des dimensions supérieure à la vraie dimension. Mais ce n'est pas le cas pour une petite taille d'échantillon et il est clairement visible que pour appliquer l'heuristique le point délicat tient dans le choix des dimensions pour estimer la pente. Même beaucoup moins marqué, ce phénomène est aussi observé pour les cas (b) et (c) : nous constatons que la pente de la courbe $\gamma_n(\hat{s}_D)$ en fonction de $f_n(D)$ dans les grandes dimensions n'a pas la même valeur que celle dans les moins grandes dimensions. Dans les grandes dimensions, $\gamma_n(\hat{s}_D)$ varie peu, la pente estimée sera plus petite et une partition de trop grande dimension pourra être sélectionnée. La question est alors de savoir sur quelles dimensions il faut estimer la pente pour pouvoir pénaliser correctement.

4.3.2 Étude de simulation

Dans cette sous-section, nous proposons deux études de simulation : la première consiste à étudier l'influence des dimensions choisies pour estimer la pente $-\alpha$ sur le calcul de la pénalité et donc sur la sélection de l'estimateur pénalisé. La seconde consiste à répondre à une question naturelle que nous nous sommes posée : la forme générale de la pénalité obtenue dans le chapitre 1 est

$$pen_n(D) = \sigma^2 \frac{D}{n}\left(c_1 \log \frac{n}{D} + c_2\right).$$

Les constantes c_1 et c_2 ont été calibrées à variance σ^2 connue dans le chapitre précédent par une étude de simulation. Pourquoi alors ne pas considérer les constantes c_1 et c_2 comme inconnues et effectuer une régression non linéaire de $\gamma_n(\hat{s}_D)$ en fonction de $\frac{\sigma^2}{2}\frac{D}{n}\left(c_1 \log \frac{n}{D} + c_2\right) = \frac{D}{n}\left(d_1 \log \frac{n}{D} + d_2\right)$? Cela permettrait une estimation directe de toutes les constantes de pénalité.

Tout d'abord, nous donnons les paramètres de l'étude et les simulations effectuées, puis les résultats observés.

Paramètres et simulations

Nous réalisons 100 échantillons de taille $n = 60, 100, 200, 400$ et 800 simulés à partir d'une variance $\sigma^2 = 1$ et de la fonction s suivante :

$$s(x) = \begin{cases} 0 & \text{si } 0 < x \leqslant 0.1 \\ 1 & \text{si } 0.1 < x \leqslant 0.35 \\ 0 & \text{si } 0.35 < x \leqslant 1 \end{cases}$$

Nous considérons les deux fonctions suivantes définies pour tout $D \geq 1$:

$$\begin{cases} f_{1,n}(D) = \alpha\frac{D}{n}(\log \frac{n}{D} + 2.5), \\ f_{2,n}(D) = \frac{D}{n}\left(d_1 \log \frac{n}{D} + d_2\right), \end{cases}$$

et les deux pénalités obtenues par les méthodes suivantes :

1. Nous supposons que $\gamma_n(\hat{s}_D)$ est une fonction affine de $f_{1,n}(D)$ de pente $-\alpha$ dans les modèles de grandes dimensions. Nous estimons la pente par une simple méthode de régression. Nous obtenons un estimateur $-\alpha$, noté $-\hat{\alpha}$, et nous définissons la pénalité par :
$$pen_{1,n}(D) = 2\hat{\alpha}\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right).$$
L'estimation de α représente l'estimation de σ^2.

2. Nous considérons que $\gamma_n(\hat{s}_D)$ est de l'ordre de $-f_{2,n}(D)$. Nous estimons les constantes de pénalité d_1 et d_2 en utilisant une régression non linéaire de $\gamma_n(\hat{s}_D)$ en fonction de $f_{2,n}(D)$. Nous obtenons $-\hat{d}_1$ et $-\hat{d}_2$ et nous définissons la pénalité par :
$$pen_{2,n}(D) = 2\frac{D}{n}\left(\hat{d}_1 \log\frac{n}{D} + \hat{d}_2\right).$$
L'estimation de d_1 et d_2 représente l'estimation de $\sigma^2 c_1/2$ et $\sigma^2 c_2/2$.

Nous cherchons à étudier l'influence des dimensions choisies pour effectuer les régressions sur la sélection des deux estimateurs. Nous estimons α, d_1 et d_2 en régressant $\gamma_n(\hat{s}_D)$ par les deux fonctions considérées sur les dimensions comprises entre :

- 7 et 15,
- 10 et 20,
- 10 et 40.

Nous notons par exemple $-\hat{\alpha}^{\{7-15\}}$ la valeur de la pente estimée par une régression de $\gamma_n(\hat{s}_D)$ par $f_{1,n}(D)$ sur les dimensions comprises entre 7 et 15, et $pen_{1,n}^{\{7-15\}}$ la pénalité obtenue.

Nous cherchons de plus à comparer la méthode heuristique avec la méthode de sélection de modèle à variance estimée et connue en considérant $f_{1,n}$.

1. Nous substituons la variance σ^2 par un estimateur dans la fonction de pénalité. Nous utilisons l'estimateur proposé par Hall *et. al* et décrit dans la sous-section 4.5.2. Nous définissons la pénalité par :
$$pen_{3,n}(D) = 2\hat{\sigma}^2\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right).$$

2. Nous supposons connue la variance σ^2 (en l'occurence ici $\sigma^2 = 1$). Nous définissons la pénalité par :
$$pen_{4,n}(D) = 2\frac{D}{n}\sigma^2\left(\log\frac{n}{D} + 2.5\right).$$

Remarque 4.4. Pour appliquer l'heuristique, il faut restreindre la collection de partitions $\mathcal{M}_n$ à une collection possédant une partition par dimension. Nous construisons la collection de partitions $\{\hat{m}_D, D = 1, \ldots, n\}$ qui est donc aléatoire. D'après la remarque de la section précédente 4.2, nous avons vérifié par simulations que $\mathbb{E}_s\left[\gamma_n(s_D) - \gamma_n(s)\right]$ était de l'ordre de $\mathbb{E}_s\left[\|s_D - s\|_n^2\right]$ pour différentes tailles d'échantillon n.

Résultats des simulations

Les Figures 4.3, 4.4 et 4.5 montrent les valeurs de $\hat{\alpha}^{\{7-15\}}$, $\hat{\alpha}^{\{10-20\}}$, $\hat{\alpha}^{\{10-40\}}$, $\hat{d}_1^{\{7-15\}}$, $\ldots$, $\hat{d}_2^{\{10-40\}}$ pour les différentes valeurs de n.

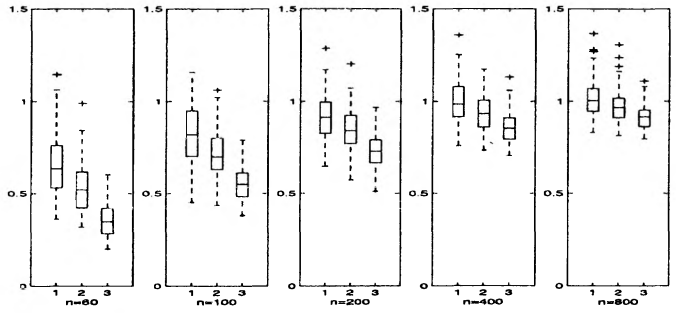


FIG. 4.3: $\hat{\alpha}^{\{7-15\}}$, $\hat{\alpha}^{\{10-20\}}$, $\hat{\alpha}^{\{10-40\}}$ *pour différentes valeurs de* n.

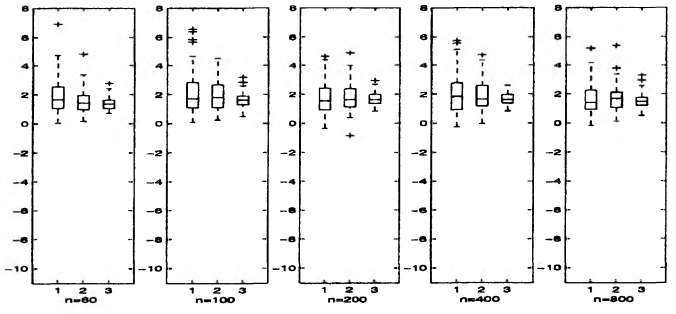


FIG. 4.4: $\hat{d1}^{\{7-15\}}$, $\hat{d1}^{\{10-20\}}$, $\hat{d1}^{\{10-40\}}$ *pour différentes valeurs de* n.

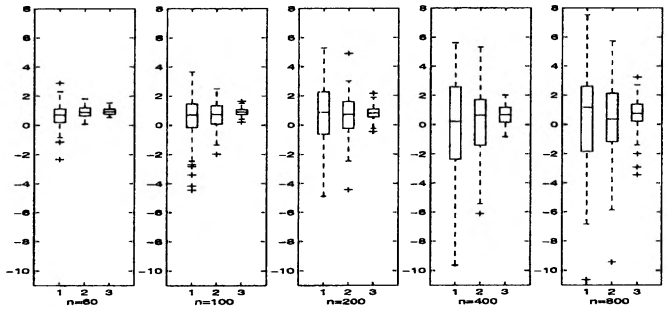


FIG. 4.5: $\hat{d}2^{\{7-15\}}$, $\hat{d}2^{\{10-20\}}$, $\hat{d}2^{\{10-40\}}$ *pour différentes valeurs de n.*

Nous constatons que :

- $\hat{\alpha}$ augmente et se stabilise avec n, et converge vers 1. Nous pénalisons beaucoup moins en utilisant la pénalité calculée à partir de $\hat{\alpha}^{\{10-40\}}$ que celle calculée à partir de $\hat{\alpha}^{\{7-15\}}$ surtout pour des petits échantillons.

- $\hat{d}_1$ reste stable avec n. Il est cependant plus variable quand nous utilisons une régression sur les dimensions comprises entre 7 et 15.

- $\hat{d}_2$ a tendance à être de plus en plus variable au fur et à mesure que n croît surtout quand nous utilisons une régression sur les dimensions comprises entre 7 et 15.

L'estimation des constantes de pénalité dépend donc des dimensions choisies pour les calculer quelque soit n. Et cela va entraîner des changements importants dans la dimension de la partition sélectionnée, comme le montre la Figure 4.6 qui représente le nombre de partitions de dimension D sélectionnées avec les pénalités considérées pour $n = 60$.

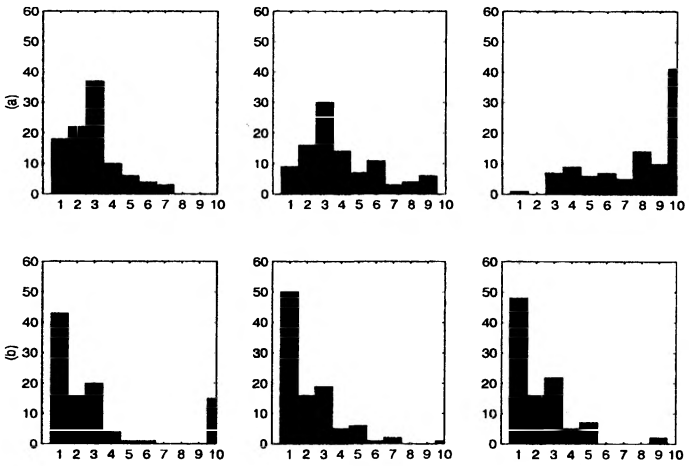


FIG. 4.6: *Nombre de partitions de dimension D sélectionnées sur* 100 *simulations à partir des pénalités* $pen_{1,n}^{\{7-15\}}$, $pen_{1,n}^{\{10-20\}}$, $pen_{1,n}^{\{10-40\}}$ *(a) et des pénalités* $pen_{2,n}^{\{7-15\}}$, $pen_{2,n}^{\{10-20\}}$, $pen_{2,n}^{\{10-40\}}$ *(b).*

Avec $pen_{1,n}^{\{10-40\}}$, nous ne pénalisons pas assez et nous avons tendance à sélectionner une partition de grande dimension, alors qu'avec $pen_{1,n}^{\{7-15\}}$, la partition de la vraie dimension est souvent sélectionnée. Avec la seconde pénalité, c'est le contraire : c'est avec $pen_{2,n}^{\{10-40\}}$ que la partition de la vraie dimension est souvent sélectionnée. Nous constatons qu'avec cette pénalité, nous avons, quelque soit les dimensions choisies pour effectuer la régression, tendance à sélectionner la partition de dimension 1.

Rappelons que pour des n petits, $\gamma_n(\hat{s}_D)$ a un comportement logarithmique en $f_n(D)$:

- Nous approchons mieux $\gamma_n(\hat{s}_D)$ en utilisant $f_{2,n}$ plutôt que $f_{1,n}$ de part le poids qui peut être mis sur le terme $\log \frac{n}{D}$ (soit $\hat{d}_1$). Mais la pénalité est le double de la fonction $f_{2,n}$ estimée et nous aurons alors tendance à sur-pénaliser et la partition sélectionnée sera de petite dimension, notamment de dimension 1.

- Avec la fonction $f_{1,n}$, le rapport des constantes égal à 2.5 oblige une certaine linéarité en $\frac{D}{n}$. Ainsi, $\hat{\alpha}^{\{10-40\}}$ sera trop petit et nous sélectionnerons une partition de trop grande dimension.

Nous nous intéressons maintenant au nombre de fois où la partition sélectionnée a la dimension de la vraie partition. Les résultats sont donnés dans le tableau 4.1 :

n	60	100	200	400	800
$pen_{1,n}^{\{7-15\}}$	**37**	**47**	**77**	**95**	**99**
$pen_{1,n}^{\{10-20\}}$	30	46	76	95	97
$pen_{1,n}^{\{10-40\}}$	7	29	70	92	97
$pen_{2,n}^{\{7-15\}}$	20	31	56	91	99
$pen_{2,n}^{\{10-20\}}$	19	30	61	90	98
$pen_{2,n}^{\{10-40\}}$	**22**	**32**	**67**	**94**	**99**
$pen_{3,n}$	**22**	**35**	**69**	**96**	**98**
$pen_{4,n}$	**18**	**37**	**77**	**96**	**99**

TAB. 4.1: *Nombre de fois où la partition sélectionnée a la même dimension que la vraie partition pour chaque pénalité sur* 100 *simulations.*

De plus, d'un point de vue sélection de modèle, c'est l'estimateur qui réalise le plus petit risque que l'on souhaiterait sélectionner. Nous estimons $\mathbb{E}_s\left[\|\tilde{s}-s\|_n^2\right]$ et $\mathbb{E}_s\left[\|\hat{s}_D-s\|_n^2\right]$ en moyennant $\|\tilde{s}-s\|_n^2$ et $\|\hat{s}_D-s\|_n^2$ sur les 100 simulations, même si le nombre de simulations est faible pour estimer les risques. Notons que l'estimateur qui réalise le plus petit risque sur la collection d'estimateurs $\{\hat{s}_D, D \geq 1\}$ est celui associé à la partition de dimension 3 pour $n \geq 100$ et 2 pour $n = 60$. Le tableau 4.2 donne le rapport du risque de l'estimateur pénalisé sur l'oracle définie par :

$$\frac{\mathbb{E}_s\left[\|\tilde{s}-s\|_n^2\right]}{\inf_{D\geq 1}\mathbb{E}_s\left[\|\hat{s}_D-s\|_n^2\right]}$$

des estimateurs sélectionnés à partir de chaque pénalité et pour chaque valeur de n.

n	60	100	200	400	800
$pen_{1,n}^{\{7-15\}}$	**1.28**	**1.16**	**1.09**	**1.057**	**1.013**
$pen_{1,n}^{\{10-20\}}$	1.61	1.36	1.14	1.06	1.041
$pen_{1,n}^{\{10-40\}}$	2.79	2.06	1.23	1.12	1.041
$pen_{2,n}^{\{7-15\}}$	1.89	1.64	1.44	1.16	1.013
$pen_{2,n}^{\{10-20\}}$	1.35	1.28	1.36	1.16	1.03
$pen_{2,n}^{\{10-40\}}$	**1.31**	**1.23**	**1.19**	**1.08**	**1.013**
$pen_{3,n}$	**1.21**	**1.166**	**1.16**	**1.0545**	**1.02**
$pen_{4,n}$	**1.19**	**1.157**	**1.07**	**1.055**	**1.014**

TAB. 4.2: *Estimation du rapport de risque de l'estimateur pénalisé sur l'oracle à partir de chaque pénalité et pour chaque valeur de n.*

Nous obtenons de meilleurs résultats avec $pen_{1,n}^{\{7-15\}}$ et $pen_{2,n}^{\{10-40\}}$. L'heuristique semble fonctionner puisque par exemple pour des échantillons de grandes tailles ($n \geq 200$), elle permet de sélectionner la bonne partition dans 67 à 98 % des cas avec ces deux pénalités. De plus, les rapports des risques des estimateurs pénalisés sur l'oracle sont plus petits et proche de ceux des estimateurs obtenus par la méthode de sélection de modèle à variance

connue et estimée. La pénalité qui semble cependant être la plus adaptée pour appliquer l'heuristique est $pen_{1,n}^{\{7-15\}}$, celle obtenue en effectuant une régression de $\gamma(\hat{s}_D)$ par $f_{1,n}(D)$ sur les dimensions comprises entre 7 et 15, assez proche de la vraie dimension qui est 3. Pour obtenir cette pénalité, il faut estimer un seul paramètre alors que pour obtenir la pénalité $pen_{2,n}$ il faut estimer deux paramètres.

Nous choisissons donc de fixer les constantes $c_1 = 2$ et $c_2 = 5$ et de considérer la fonction $f_n = f_{1,n}$. La principale difficulté alors est de choisir correctement les dimensions de la régression (surtout pour des petits échantillons) pour obtenir une bonne pénalité, et donc le bon estimateur.

4.4 Méthode

La pénalité que l'on souhaiterait obtenir est appelée **pénalité optimale**. La régression utilisée dans l'heuristique permet d'estimer $\alpha f_n(D)$. Cette grandeur est appelée **pénalité minimale** est vaut d'après l'égalité (4.2.5), la moitié de la pénalité optimale. Pour résoudre la difficulté du choix des dimensions soulevée par les études de simulations précédentes, nous allons utiliser une heuristique dévelopée par Birgé et Massart [8] qui s'appuie sur des résultats théoriques concernant la pénalité minimale. Nous évoquons ces résultats et donnons l'heuristique qui s'en dégage. Après l'avoir vérifiée en pratique, nous donnons la méthode de sélection.

Dans l'étude des processus linéaires Gaussiens, la fonction de pénalité satisfait $pen(m) > K\sigma^2\frac{D_m}{n}(1+\sqrt{2L_m})^2$ avec une constante K supérieure strictement à 1 et une famille convenable de poids $\{L_m\}_{m\in\mathcal{M}_n}$ (cf Théorème 2.4.2). Le choix de la constante K est discuté dans [7] : une trop grande valeur de K mène à une majoration du risque qui tend vers l'infini. D'autre part, si K est plus petit que 1, la dimension de la partition sélectionnée a tendance à être très élevée et les estimateurs qui en résultent n'ont alors pas de bonnes propriétés. La pénalité $K\sigma^2\frac{D_m}{n}(1+\sqrt{2L_m})^2$ avec K proche de 1 par valeurs supérieures est appelé **pénalité minimale** : c'est la plus petite pénalité avec laquelle la partition sélectionnée est de dimension raisonnable. Cela sous-entend que lors du passage à la pénalité minimale, la dimension de l'estimateur associé devrait chuter subitement. Pour estimer la pénalité minimale, l'idée consiste donc à faire varier la constante de pénalité, sélectionner la dimension à partir de chaque pénalité et essayer de repérer le passage à la pénalité minimale en nous intéressant à ces dimensions. Il suffit ensuite de prendre le double de cette pénalité pour obtenir la pénalité optimale.

Considérons la pénalité définie pour tout $D \geq 1$ par :

$$pen_{\alpha,n}(D) = \alpha f_n(D).$$

Le critère correspondant est

$$crit_{\alpha,n}(D) \;=\; \gamma_n(\hat{s}_D) + pen_{\alpha,n} = \gamma_n(\hat{s}_D) + \alpha\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right).$$

A α fixé, la partition sélectionnée est $\hat{D}(\alpha)$, où $\hat{D}(\alpha)$ minimise le critère $crit_{\alpha,n}(D)$ selon D. La procédure consiste à faire varier α, appelée température, par petit pas en partant de la valeur 0 et à sélectionner pour chaque α, le $\hat{D}(\alpha)$ correspondant.

Lemme 4.4.1. *Il existe une suite croissante finie de températures*

$$\alpha_1 = 0 > \alpha_2 > \ldots > \alpha_K,$$

et une suite décroissante finie de dimensions associées à chaque température

$$D_1 = n > D_2 > \ldots > D_K = 1,$$

où pour $i = 2, \ldots, K$

$$\alpha_i = \min_{j<D_{i-1}} \frac{\gamma_n(\hat{s}_{\hat{m}(j)}) - \gamma_n(\hat{s}_{\hat{m}(D_{i-1})})}{f_n(D_{i-1}) - f_n(j)} = \frac{\gamma_n(\hat{s}_{\hat{m}(D_i)}) - \gamma_n(\hat{s}_{\hat{m}(D_{i-1})})}{f_n(D_{i-1}) - f_n(D_i)},$$

et $D_i = \hat{D}(\alpha_i)$. *De plus, étant donné* $i \in \{1, \ldots, K-1\}$, *si* $\alpha \in [\alpha_i, \alpha_{i+1}[$, *alors* $\hat{D}(\alpha) = \hat{D}(\alpha_i) = D_i$.

Preuve. Préalablement, nous démontrons le lemme suivant :

Lemme 4.4.2. *si* $\alpha_1 \leqslant \alpha_2$ *alors* $\hat{D}(\alpha_1) \geq \hat{D}(\alpha_2)$.

Preuve. Par définition, si $\alpha_1 \leqslant \alpha_2$ nous avons

$$\begin{aligned}
crit_{\alpha_1,n}(\hat{D}(\alpha_1)) &\leqslant crit_{\alpha_1,n}(\hat{D}(\alpha_2)) \\
&\leqslant \gamma_n(\hat{s}_{\hat{D}(\alpha_2)}) + \alpha_1 f_n(\hat{D}(\alpha_2)) \\
&\leqslant \gamma_n(\hat{s}_{\hat{D}(\alpha_2)}) + \alpha_2 f_n(\hat{D}(\alpha_2)) + (\alpha_1 - \alpha_2) f_n(\hat{D}(\alpha_2)) \\
&\leqslant crit_{\alpha_2,n}(\hat{D}(\alpha_2)) + (\alpha_1 - \alpha_2) f_n(\hat{D}(\alpha_2)) \\
&\leqslant crit_{\alpha_2,n}(\hat{D}(\alpha_1)) + (\alpha_1 - \alpha_2) f_n(\hat{D}(\alpha_2)).
\end{aligned}$$

En remplaçant le critère par son expression, nous obtenons

$$\gamma_n(\hat{s}_{\hat{D}(\alpha_1)}) + \alpha_1 f_n(\hat{D}(\alpha_1)) \leqslant \gamma_n(\hat{s}_{\hat{D}(\alpha_1)}) + \alpha_2 f_n(\hat{D}(\alpha_1)) + (\alpha_1 - \alpha_2) f_n(\hat{D}(\alpha_2)),$$

d'où

$$(\alpha_1 - \alpha_2)(f_n(\hat{D}(\alpha_1)) - f_n(\hat{D}(\alpha_2))) \leqslant 0.$$

Or, par hypothèse, $\alpha_1 \leqslant \alpha_2$, donc on a que $f_n(\hat{D}(\alpha_1)) \geq f_n(\hat{D}(\alpha_2))$. Et de plus, la fonction f_n est strictement croissante, donc $\hat{D}(\alpha_1) \geq \hat{D}(\alpha_2)$.

La preuve du lemme 4.4.1 résulte de la construction simultanée des suites $(\alpha_i)_{1\leqslant i\leqslant K}$ et $(D_i)_{1\leqslant i\leqslant K}$.

Hypothèses et notations :

- $\alpha_1 = 0$ et α_K est tel que $\hat{D}(\alpha_K) = 1$.
- $\hat{D}(\alpha_1) = D_1 = n$ et $\hat{D}(\alpha_K) = D_K = 1$.
- posons $g(i,j) = \dfrac{\gamma_n(\hat{s}_i) - \gamma_n(\hat{s}_j)}{f_n(j) - f_n(i)}$.

Le principe de la construction est d'itérer la température α de α_1 à une certaine valeur de α qui est ici α_K par petits pas et de sélectionner pour chaque valeur la meilleure partition $\hat{D}(\alpha)$. Nous avons $\alpha_1 \leqslant \alpha \leqslant \alpha_K$. Nous connaissons α_1 par hypothèse et nous cherchons à construire α_2.

$\alpha \geq \alpha_1$, donc d'après le lemme 4.4.2, $\hat{D}(\alpha) \leqslant \hat{D}(\alpha_1) = D_1$. Nous considérons les dimensions strictement inférieures à D_1. Par définition, $\hat{D}(\alpha) = D_1$ si et seulement si pour $j < D_1$, nous avons

$$\begin{aligned} crit_{\alpha,n}(D_1) &< crit_{\alpha,n}(j) \\ \gamma_n(\hat{s}_{D_1}) + \alpha f_n(D_1) &< \gamma_n(\hat{s}_j) + \alpha f_n(j) \\ \alpha &< g(j, D_1). \end{aligned}$$

Donc $\hat{D}(\alpha) = D_1$ si et seulement si $\alpha < \min_{j<D_1} g(j, D_1)$. Prenons

$$\alpha_2 = \min_{j<D_1} g(j, D_1),$$

et

$$D_2 = \hat{D}(\alpha_2).$$

Nous avons construit α_2 tel que $\alpha_1 < \alpha_2$ et par le lemme 4.4.2, $D_1 > D_2$. Les valeurs suivantes des deux suites sont construites de la même façon.

Remarque 4.5. Il est facile de montrer que $\alpha_i = g(D_i, D_{i-1})$ pour $i = 2, \ldots, K-1$.

Remarques 4.6.

- La valeur α_i est associée à la suite décroissante de dimensions $D_{i-1} - D_i$ pour $i = 2, \ldots, K$.
- La fonction $\alpha \to \hat{D}_\alpha$ est une fonction constante par morceaux et s'écrit $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$.

- La valeur $-\alpha_i$ correspond à l'estimation de α obtenu en régressant $\gamma_n(\hat{s}_{(D)})$ par $f_n(D)$ entre les dimensions D_{i-1} et D_i. La suite $(-\alpha_i)_{1\leqslant i\leqslant K}$ peut être identifiée à l'enveloppe convexe des points $(f_n(D), \gamma_n(\hat{s}_D))$ pour $D \in \{D_1, \ldots, D_K\}$. Ou encore les sauts de dimensions correspondent aux dimensions entre lesquelles le graphe $(f_n(D), \gamma_n(\hat{s}_D))$ est concave.

La fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ pour les trois réalisations considérées dans la sous-section 4.3.1, est dessinée respectivement Figures 4.7−(*a*), 4.7−(*b*) et 4.7−(*c*). L'enveloppe convexe des points $(f_n(D), \gamma_n(\hat{s}_D))$ pour $D \in \{D_1, \ldots, D_K\}$ est représentée sur les Figures 4.2−(*a*), 4.2 − (*b*) et 4.2 − (*c*) par des $*$.

En pratique, nous observons un phénomène assez marqué concernant la suite des dimensions $(D_i)_{1\leqslant i\leqslant K}$ qui semble correspondre au passage à la pénalité minimale : pour des petites valeurs de α, les dimensions des partitions sélectionnées restent assez élevées, puis chutent brusquement vers une plus petite valeur quand α atteint un certain seuil, notée $\hat{\alpha}$. Par conséquent, $\hat{\alpha}$ est la valeur de la suite $(\alpha_i)_{1\leqslant i\leqslant K}$ qui est associée au plus grand saut de dimensions observé dans la suite $(D_i)_{1\leqslant i\leqslant K}$. La pénalité minimale est alors définie par :

$$\hat{\alpha}\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right),$$

le critère pénalisé est défini par :

$$crit_n(D) = \gamma_n(\hat{s}_D) + 2\hat{\alpha}\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right), \tag{4.4.6}$$

et l'estimateur final est $\tilde{s} = \hat{s}_{\hat{D}}$ où $\hat{D}$ minimise $crit_n(D)$ en D.

4.5 Calibration de la méthode

En pratique, nous constatons des difficultés pour estimer le paramètre α. Dans cette section, nous discutons de ces problèmes puis nous proposons une calibration de la méthode afin qu'elle soit applicable dans la majorité des situations.

4.5.1 Problèmes et choix

Rappelons qu'en pratique, nous construisons la collection de partitions $\{\hat{m}_D, D = 1, \ldots, D_{max}\}$, où $D_{max} << n$.
Deux questions concernant l'estimation de α se sont posées :

1. si le saut maximal de dimension de la suite $(D_i)_{i=1,\ldots,K}$ est atteint par plusieurs valeurs de la suite $(\alpha_i)_{i=1,\ldots,K}$, laquelle choisir pour l'estimation de α?
2. la dimension maximale des partitions considérées, D_{max}, influence-t-elle l'estimation de α, et donc le choix de l'estimateur final?

La valeur $\hat{\alpha}$ est associée au plus grand saut de dimensions reflétant le passage à la pénalité minimale. Par conséquent, nous nous intéressons au premier moment où il y a un changement brusque de dimensions, c'est-à-dire, à la plus petite valeur parmi les différentes valeurs de α retenues. Des simulations effectuées ont permis de confirmer ce choix.

Il se peut qu'en pratique, des sauts maximaux soient observés pour des valeurs de α très petites. Dans cette situation, la pénalité sera trop petite pour pénaliser correctement le critère et une partition de trop grande dimension sera sélectionnée. Ce problème est illustré par le cas (a) : sur la Figure $4.7-(a)$, les valeurs a_1 et a_2 représentent les deux valeurs de la suite de températures associées respectivement aux deux plus grands sauts de dimensions. Si nous prenons $D_{max} = 40$, alors $\hat{\alpha} = a_1$ et la partition sélectionnée est de dimension 15,

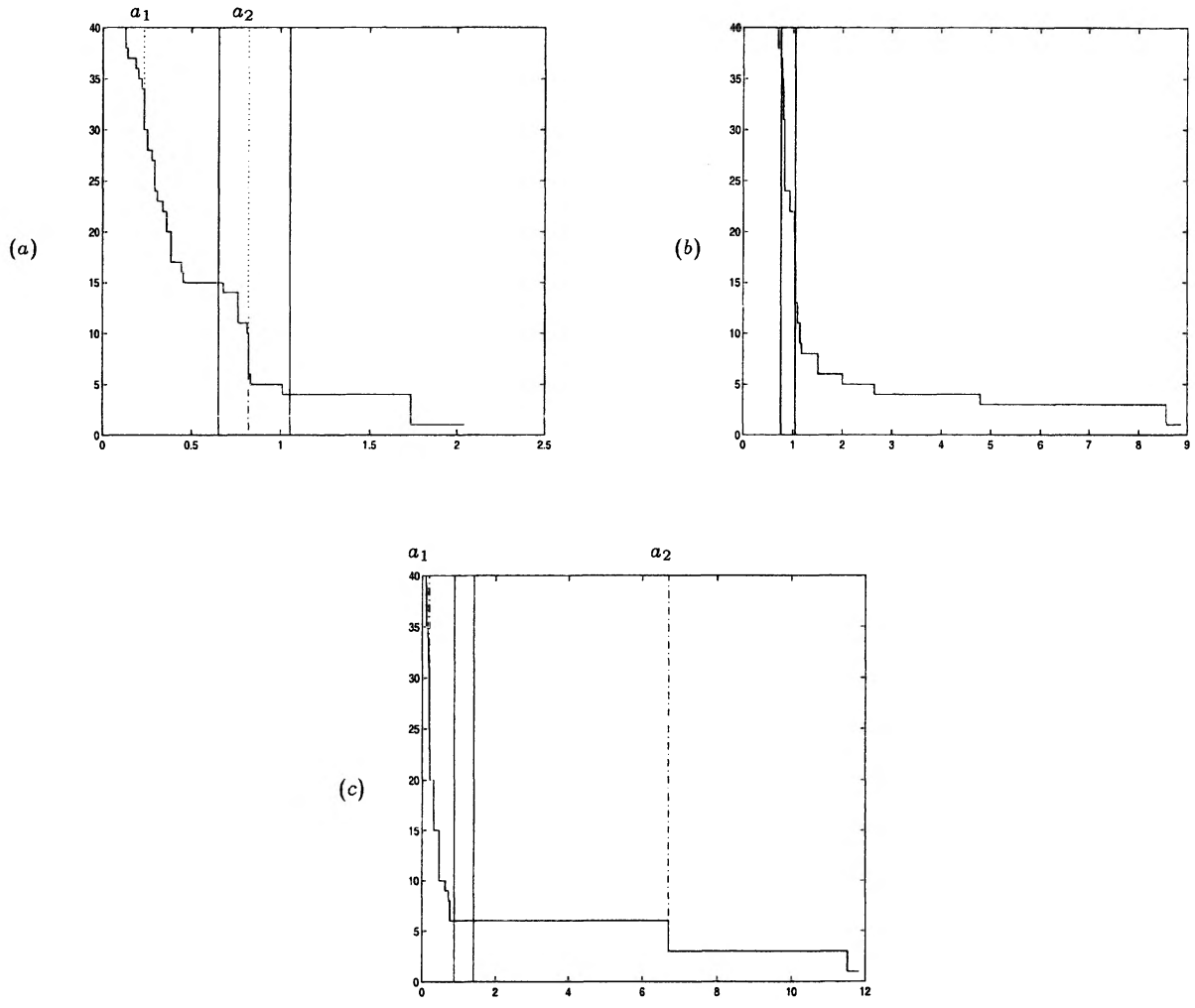


FIG. 4.7: *Graphe de la fonction* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$ *pour* $i = 1, \ldots, K$ *obtenue pour les trois réalisations.*

tandis que si $D_{max} = 25$, alors $\hat{\alpha} = a_2$ et la partition sélectionnée est de dimension 4. Pour répondre à la question soulevée, D_{max} peut jouer un rôle dans l'estimation de α et donc dans la sélection de l'estimateur pénalisé.
En pratique, l'utilisateur devra choisir la dimension maximale D_{max} selon les informations dont il dispose. Cependant, notre objectif est de proposer une méthode automatique. Une valeur universelle de D_{max} n'est pas concevable puisqu'elle dépend du problème posé. Nous proposons alors d'éliminer les partitions de trop grandes dimensions en contraignant α à être supérieure à un seuil noté α_{seuil}. Dans notre cadre d'étude, α joue le rôle de la variance σ^2, il est donc impossible de choisir un α_{seuil} universel. En posant $\alpha_i = \sigma^2\beta_i$, la suite $(\beta_i)_{1\leqslant i\leqslant K}$ ne dépend pas de σ^2. Il suffit donc de choisir un β_{seuil} et de poser $\alpha_{seuil} = \sigma^2\beta_{seuil}$. La variance σ^2 étant bien sûr inconnue, nous proposons de la substituer par un estimateur $\hat{\sigma}^2$. Le nombre de ruptures ainsi que leurs localisations étant inconnus, l'estimateur classique de régression ne peut être utilisé. Dans la première partie, l'estimateur du maximum de vraisemblance est obtenu par l'algorithme SAEM. Un bon résultat est obtenu en quelques secondes. Pour savoir si la méthode de détection de ruptures proposée ici fonctionne bien, nous avons besoin d'estimer le risque de l'estimateur de s pénalisé, par une méthode de Monte Carlo. Plutôt que d'utiliser l'estimateur du maximum de vraisemblance avec SAEM, ce qui serait très long, nous avons préféré utiliser un autre estimateur moins optimal mais obtenu très rapidement.

4.5.2 Estimateur de la variance

Nous avons choisi d'utiliser l'estimateur de la variance proposé par Hall *et. al* [32]. Nous ne disposons pas de justifications théoriques sur le choix particulier de cet estimateur, mais dans la pratique, il donne de bonnes estimations de la variance. Nous détaillons cet estimateur. Nous considèrons un modèle de régression général :

$$y_t = f(x_t) + \varepsilon_t \quad (t = 1, \ldots, n),$$

où f est une fonction inconnue et les erreurs ε_t sont indépendantes et identiquement distribuées de loi $\mathcal{N}(0, \sigma^2)$, et les x_t sont ordonnés. Hall *et. al* proposent l'estimateur de la variance suivant :

$$\hat{\sigma}^2 = (n - M)^{-1} \sum_{k=1}^{n-M} (\sum_{j=0}^{M} d_j y_{j+k})^2,$$

où quelque soit M, la suite $(d_j)_{j=1,\ldots,M}$ est une suite de nombres réels telle que

$$\sum_{j=0}^{M} d_j = 0, \quad \sum_{j=0}^{M} d_j^2 = 1.$$

La suite $(d_j)_{j=0,\ldots,M}$ est obtenue par minimisation de la variance asymptotique. Hall *et. al* présentent trois manières de construire la suite $(d_j)_{j=0,\ldots,M}$ et discute de la performance des trois suites obtenues en comparant l'efficacité de l'estimateur $\hat{\sigma}^2$ qui en résulte relativement

à l'estimateur classique $\hat{\sigma}_0^2 = \frac{1}{n}\sum_{t=1}^{n}\varepsilon_t^2$. Les performances des estimateurs sont optimales pour $2 \leqslant M \leqslant 5$.

Nous avons choisi $M = 3$ et la suite la plus performante est donnée par :

$$(d_j)_{j=0,\dots,3} = (0.1942, 0.2809, 0.3832, -0.8582).$$

L'efficacité de cet estimateur relative à $\hat{\sigma}_0^2$ est alors donnée par $\frac{var(\hat{\sigma}^2)}{var(\hat{\sigma}_0^2)} = \frac{2M}{2M+1} = 0.86$.

4.5.3 Choix de la valeur seuil

Dans cette sous-section, nous proposons de choisir une valeur β_{seuil} à partir d'un test d'hypothèse nulle H_0 : "qu'aucune rupture n'est présente" contre l'hypothèse alternative H_1 : "qu'il existe au moins une rupture". Heuristiquement, ce test correspond aussi au test de l'hypothèse nulle qu'il n'existe aucune rupture supplémentaire à la "bonne" partition contre l'hypothèse alternative qu'il en existe.

Nous cherchons une valeur de β_{seuil} telle qu'en appliquant la méthode pour $\alpha \geq \hat{\sigma}^2\beta_{seuil}$, nous obtenons sous H_0 la partition de dimension 1, avec une grande probabilité. Donc nous voulons tester :

$$"\alpha \geq \hat{\sigma}^2\beta_{seuil}\ "\ \text{ contre }\ "\alpha < \hat{\sigma}^2\beta_{seuil}.\ "$$

Soit α_v la valeur de la suite $(\alpha_i)_{i=1,\dots,K}$ à partir de laquelle la méthode mène à la partition de dimension 1. Posons $\alpha_v = \hat{\sigma}^2\beta_v$, nous avons que

si $\beta_{seuil} \geq \beta_v$ alors pour $\alpha \geq \hat{\sigma}^2\beta_{seuil}$ la partition de dimension 1 est sélectionnée.

Ainsi le test devient

$$H_0 : "\beta_{seuil} \geq \beta_v" \text{ contre } H_1 : "\beta_{seuil} < \beta_v".$$

Notons z le niveau du test, nous décidons d'accepter la partition de dimension 1 quand $\beta_v \leqslant q_{1-z}(\beta_v)$ où $q_{1-z}(\beta_v)$ est le quantile empirique d'ordre $1-z$ de β_v. Nous prenons

$$\beta_{seuil} = q_{1-z}(\beta_v).$$

Suite à la présentation de la construction du test, une question se pose : comment choisir le niveau z? En pratique nous remarquons un changement de comportement de la fonction constante par morceaux $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$ à partir d'une taille d'échantillon $n = 200$: plus la taille de l'échantillon est grande, plus la suite $(\alpha_i)_{i=1,\dots,K}$ décroit lentement et montre une certaine stabilité. Nous décidons de prendre deux niveaux de tests selon la taille de l'échantillon en autorisant plus de souplesse pour des petits échantillons.

La loi de β_v n'étant pas connue, nous allons estimer empiriquement le quantile considéré. Nous utilisons la procédure de simulations suivante : nous prenons $s = 0$ (pour se placer

sous l'hypothèse nulle), $\sigma^2 = 1$ et différentes valeurs du niveau z. Étant donné n, nous simulons $n_{ite} = 2000$ échantillons. Nous obtenons alors un n_{ite} échantillon (β_v). Nous notons $\tilde{\beta}_v = (\beta_{v,(1)}, \ldots, \beta_{v,(n_{ite})})$ le vecteur obtenu en ordonnant les composantes de β_v. L'estimateur empirique de $q_{1-z}(\beta_v)$ est donné par :

$$\bar{q}_{1-z}(\beta_v) = \frac{\beta_{v,([n_{ite}(1-z)])} + \beta_{v,([n_{ite}(1-z)]+1)}}{2}.$$

Nous effectuons cette opération 10 fois. Nous disposons alors d'un échantillon $(\beta^{(j)}_{seuil}(n))_{j=1,\ldots,10}$ et nous considérons la valeur moyenne :

$$\beta_{seuil}(n) = \frac{1}{10}\sum_{j=1}^{10} \beta^{(j)}_{seuil}(n)$$

Cette procédure est effectuée pour différentes valeurs de n allant de 20 à 1000. Les résultats sont donnés dans le tableau 4.3 suivant :

n	$\beta_{seuil}(n)$	z
20	0.611	0.15
40	0.625	0.15
60	0.62	0.15
100	0.603	0.15
200	0.762	0.05
300	0.743	0.05
500	0.7	0.05
1000	0.714	0.05

TAB. 4.3: *Estimations des $\beta_{seuil}(n)$ pour les différentes valeurs de n.*

Maintenant le problème consiste à choisir la valeur seuil pour chaque cas ($n < 200$ et $n \geq 200$) : nous proposons de prendre le seuil proche de la valeur maximale. En effet, dans ce cas, en appliquant la méthode avec $\beta > \beta_{seuil} > \beta_{seuil}(n)$, la dimension de la partition sélectionnée sera 1. Nous décidons de prendre

$$\beta_{seuil} = \begin{cases} 0.62 & \text{si } n < 200 \\ 0.76 & \text{si } n \geq 200 \end{cases}$$

Nous estimons maintenant la puissance du test, que nous considérons comme une fonction du saut de moyenne, uniquement pour $n = 60$. Nous simulons 200 échantillons de taille n à partir d'une variance $\sigma^2 = 1$ et d'une fonction s appartenant au modèle de dimension 3 de la famille :

$$s_\delta(x) = \begin{cases} 0 & \text{si } 0 \leqslant x \leqslant 0.3 \\ \delta & \text{si } 0.3 < x \leqslant 0.7 \\ 0 & \text{si } 0.7 < x \leqslant 1 \end{cases}$$

où δ varie de 0 à 2.5 par pas de 0.04. La valeur de la fonction puissance du test en un δ donné est estimée par :

$$\begin{aligned} \mathcal{L}_{\beta_s}(\delta) &= P_\delta(\beta_v > \beta_{seuil}) \\ &= \frac{1}{200}\sum_{j=1}^{200} \mathbb{1}_{(\beta_{v,j} > \beta_{seuil})} \end{aligned}$$

Cette fonction puissance est représentée Figure 4.8.

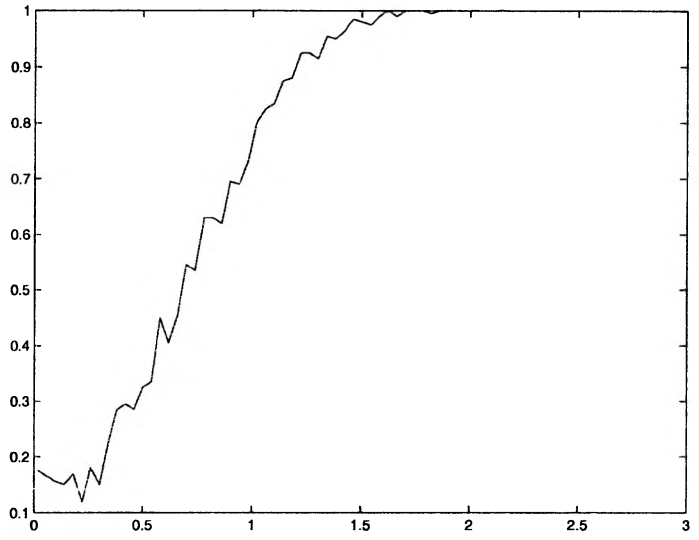


FIG. 4.8: *Graphe de la fonction* $\delta \to \mathcal{L}_{\beta_s}(\delta)$.

4.5.4 Problèmes et calibration finale

Dans cette sous-section, nous discutons du rôle de β_{seuil} et de certains problèmes observés en pratique. Nous nous appuyons sur les trois réalisations considérées tout au long de l'étude.

Les Figures 4.9 – (a), 4.9 – (b) et 4.9 – (c) représentent respectivement le graphe de la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$ obtenue pour les trois réalisations. Tout d'abord, le rôle de β_{seuil} est clairement visible sur la Figure 4.9 – (a) : nous remarquons que $a_1 < \hat{\sigma}^2\beta_{seuil} < a_2$. Ainsi, en appliquant la méthode sans seuil, $\hat{\alpha} = a_1$, nous sur-pénalisons et sélectionnons la partition de dimension 15, tandis que si nous forçons α à être plus grand que $\hat{\sigma}^2\beta_{seuil}$, alors $\hat{\alpha} = a_2$, nous sélectionnons la partition de dimension 4. Par conséquent, la performance de l'estimateur peut être vraiment différente.

En pratique, il se peut que le seuil α_{seuil} soit trop grand : soit parce que la variance est sur-estimée, soit parce que le β_{seuil} est lui même trop grand. Dans ce cas, nous ne pouvons

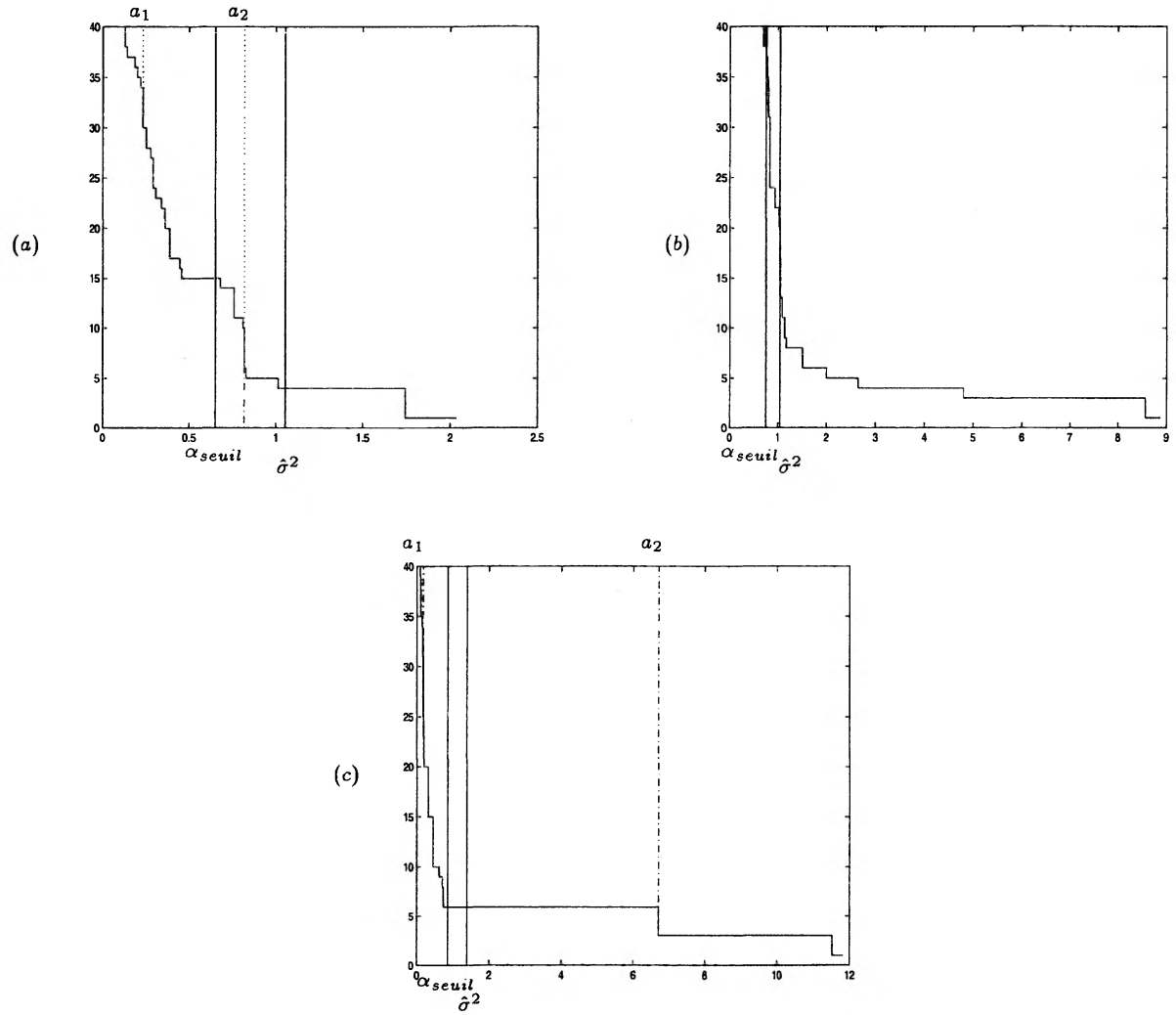


FIG. 4.9: *Graphe de la fonction* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ *pour* $i = 1, \ldots, K$ *obtenue pour les trois réalisations.*

espérer repérer la pénalité minimale. Par exemple, soit α_{min}, la valeur de α correspondant à la pénalité minimale ($pen_{min}(D) = \alpha_{min} f_n(D)$). Si $\alpha_{seuil} > \alpha_{min}$, il est clair que nous ratons la pénalité minimale. Prenons l'exemple (c) (Figure 4.7 – (c)) : nous remarquons que la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ présente un large plateau $I_v = [\alpha_v, \alpha_{v+1}[$ associé à la meilleure partition au sens de la perte de dimension 6, celle que l'on cherche à sélectionner. Notons que $\alpha_{v+1} = a_2$. Le seuil α_{seuil} appartient à cet intervalle I_v. Ainsi, en appliquant la méthode avec seuil, $\hat{\alpha} = \alpha_{v+1}$ et en prenant $2\hat{\alpha}$, nous obtenons $\hat{D} = 1$. Nous ne repérons pas la pénalité minimale et surpénalisons. Pour palier à ce problème, l'idée est de contrôler α par $\hat{\sigma}^2$.

En conclusion, nous proposons la calibration suivante : nous prenons α tel que

$$\beta_{seuil}\hat{\sigma}^2 \leqslant \alpha \leqslant \hat{\sigma}^2.$$

Nous extrayons de la suite $(\alpha_i)_{i=1,\dots,K}$, la suite

$$(\alpha_{i_k})_{k=1,\dots,b}, \tag{4.5.7}$$

et la suite de dimensions correspondantes

$$(D_{i_k})_{k=1,\dots,b}, \tag{4.5.8}$$

telles que

- $\alpha_{i_1} = \beta_{seuil}\hat{\sigma}^2$,
- $\alpha_{i_b} = \hat{\sigma}^2$,
- $(\alpha_{i_k})_{i=2,\dots,b-1} = \{(\alpha_i)_{i=1,\dots,K} | \beta_{seuil}\hat{\sigma}^2 < \alpha_i < \hat{\sigma}^2\}$.

$\hat{\alpha}$ est la valeur de la suite $(\alpha_{i_k})_{k=1,\dots,b}$ qui est associée au plus grand saut de dimension observé dans la suite $(D_{i_k})_{k=1,\dots,b}$. Nous avons alors deux cas :

- il existe au moins un saut de dimension,

$$\hat{\alpha} = \underset{i=1,\dots,b-1; D_{i_{k+1}} - D_{i_k} > 0}{\operatorname{argmin}} D_{i_{k+1}} - D_{i_k}, \tag{4.5.9}$$

- sinon

$$\hat{\alpha} = \hat{\sigma}^2. \tag{4.5.10}$$

L'estimateur pénalisé est défini par

$$\tilde{s} = \hat{s}_{\hat{m}_{\hat{D}}},$$

où $\hat{D} = \hat{D}(2\hat{\alpha})$.

4.5.5 Résumé de la méthode calibrée

Les grandes étapes de la méthode finale calibrée sont résumées sous la forme d'un organigramme représenté Figure 4.10.

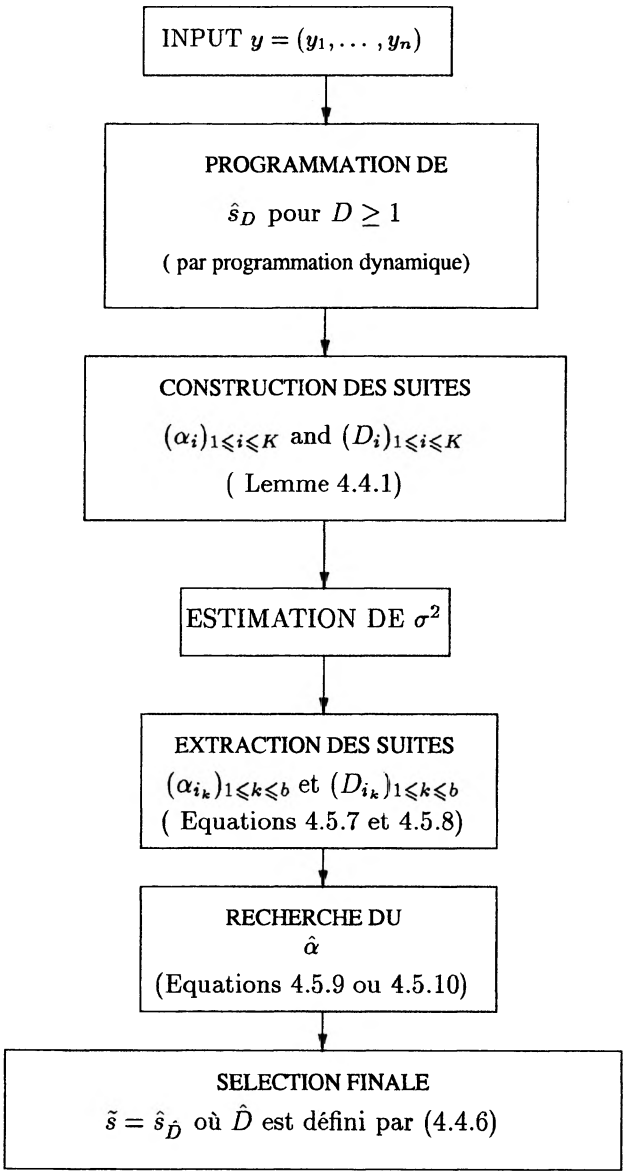


FIG. 4.10: *Organigramme de la méthode calibrée.*

4.5.6 Applications

Nous présentons ici les estimateurs obtenus par l'application de la méthode calibrée sur les trois réalisations considérées tout au long de cette étude, que nous comparons à ceux obtenus à variance connue et estimée.

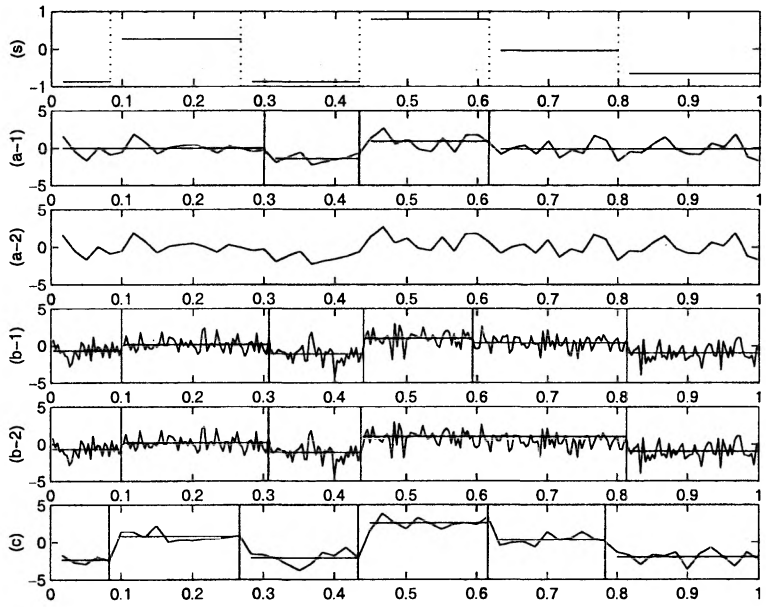


FIG. 4.11: *Graphiques des estimateurs pénalisés obtenus par la méthode calibrée* $(a-1)$, $(b-1)$ *et* (c), *et ceux obtenus par la méthode de sélection de modèle à variance connue et estimée* $(a-2)$, $(b-2)$ *et* (c), *pour les trois réalisations.*

Les résultats sont les suivants :

(a) L'estimateur pénalisé appartient au modèle de dimension 4, $\tilde{s} = \hat{s}_4$. Il est représenté Figure 4.11 – $(a-1)$. Notons que trois des ruptures estimées sont proches des "vraies" et que deux ne sont pas détectées. À variance connue et estimée, les estimateurs sont les mêmes : $\tilde{s} = \hat{s}_1$ (cf Figure 4.11 – $(a-2)$). Les fonctions de pertes associées aux deux estimateurs sont respectivement : $\|s - \hat{s}_4\|_n^2 = 0.196$ et $\|s - \hat{s}_1\|_n^2 = 0.386$. La méthode calibrée sélectionne donc un meilleur estimateur en terme de perte. De plus, l'estimateur de perte minimale est celui de dimension 4 : $\inf_{D \geq 1} \|s - \hat{s}_D\|_n^2 = \|s - \hat{s}_4\|_n^2 = 0.196$.

(b) De la même façon, la dimension de l'estimateur sélectionné par la méthode calibrée est 6, $\tilde{s} = \hat{s}_6$. L'estimateur associé est représenté Figure 4.11 – $(b-1)$. Ceux obtenus avec $\sigma^2 = 1$ et $\hat{\sigma}^2$ sont de dimension 5, $\tilde{s} = \hat{s}_5$ (cf Figure 4.11 – $(b-2)$). De plus, $\|s - \hat{s}_6\|_n^2 = \inf_{D \geq 1} \|s - \hat{s}_D\|_n^2 = 0.165$ et $\|s - \hat{s}_5\|_n^2 = 0.208$

(*c*) Les estimateurs obtenus par les trois méthodes considérées sont de même dimension : $\hat{D} = 6$ (cf Figure 4.11 – (*c*)).

4.6 Étude de simulations

4.6.1 Simulations et résultats

Dans cette sous-section, nous comparons la performance de la méthode calibrée avec celle de la méthode de sélection de modèle à variance σ^2 connue et estimée. Notons :

- m_1 : la méthode calibrée. La pénalité est définie pour tout $D \geq 1$ par :

$$\hat{\alpha}\frac{D}{n}\left(2\log\frac{n}{D}+5\right).$$

- m_2 : la méthode de sélection de modèle à variance connue. La pénalité est définie pour tout $D \geq 1$ par :

$$\sigma^2\frac{D}{n}\left(2\log\frac{n}{D}+5\right).$$

- m_3 : la méthode de sélection de modèle pour laquelle la variance a été substituée par $\hat{\sigma}^2$. La pénalité est définie pour tout $D \geq 1$ par :

$$\hat{\sigma}^2\frac{D}{n}\left(2\log\frac{n}{D}+5\right).$$

La performance de ces méthodes est évaluée par :

1. le rapport du risque de l'estimateur pénalisé sur l'oracle :

$$F_n(s,2,5) = \frac{\mathbb{E}_s\left[\|s-\tilde{s}\|_n^2\right]}{\inf_{D\geq 1}\mathbb{E}_s\left[\|s-\hat{s}_D\|_n^2\right]}. \tag{4.6.11}$$

   Les risques $\mathbb{E}_s\left[\|s-\tilde{s}\|_n^2\right]$ et $\mathbb{E}_s\left[\|s-\hat{s}_D\|_n^2\right]$ sont estimés par une méthode de Monte Carlo en moyennant respectivement les valeurs $\|s-\tilde{s}\|_n^2$ et $\|s-\hat{s}_D\|_n^2$ sur N_b simulations. Nous renvoyons à la sous-section 3.3.1 du chapitre précédent.

2. $\%p_{min}$: le pourcentage d'estimateurs sélectionnés qui réalisent la perte minimale, c'est-à-dire $\inf_{D\geq 1}\|s-\hat{s}_D\|_n^2$ sur toutes les simulations effectuées pour estimer les risques précédents.

Nous prenons les paramètres suivants :

- $N_b = 500$.

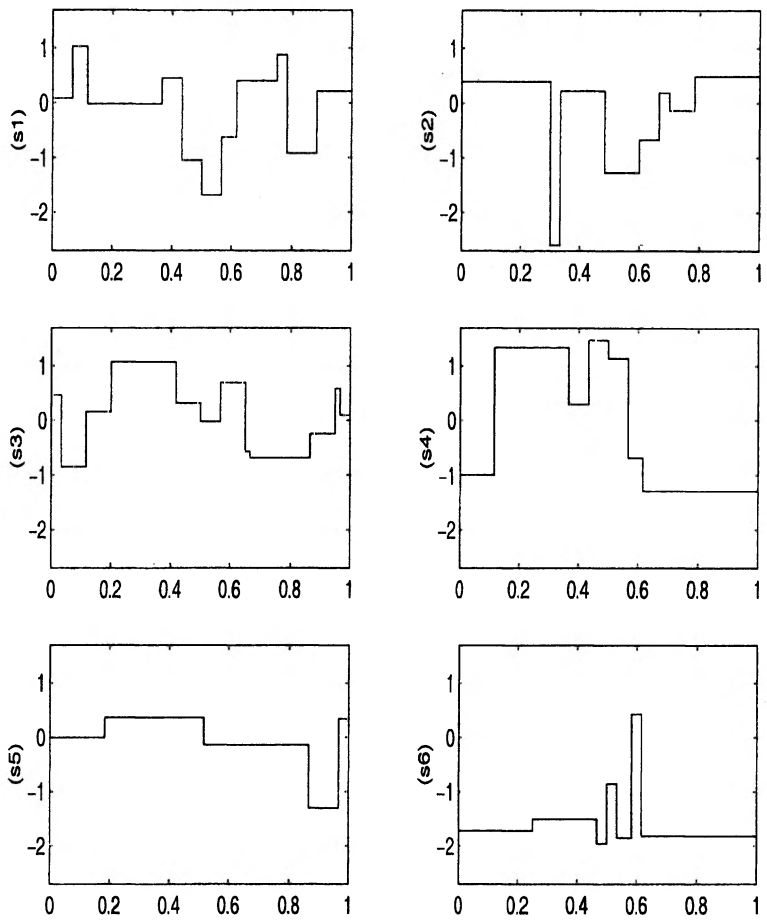


FIG. 4.12: *Les fonctions* s_i $i = 1, \ldots, 6$.

- $\sigma^2 = 1$.
- 2 valeurs différentes de n : $n = 60$ et $n = 300$.
- 6 fonctions représentées Figure 4.12.

Les résultats des simulations sont donnés dans le tableau 4.4.

- La première remarque générale est que les rapports des risques des estimateurs sur l'oracle sont proches, et tendent vers 1 avec n : les estimateurs s'approchent de l'oracle avec n. Pour $n = 300$ et la fonction s_2, la méthode à variance connue mène à un rapport de risques inférieur à 1, ce qui signifie que l'estimateur est meilleur que l'oracle, ce qui peut paraître étonnant puisque l'oracle correspond à celui que l'on choisiraît si on connaissait la fonction s_2.

		$n = 60$			$n = 300$		
		m_1	m_2	m_3	m_1	m_2	m_3
s_1	$F_n(s_1, 2, 5)$	1.1	1.15	1.176	1.005	1.008	1.007
	$\%p_{min}$	21.6	18.2	16.6	19	20	20
s_2	$F_n(s_2, 2, 5)$	1.21	1.27	1.47	1.01	0.99	1.01
	$\%p_{min}$	24	21	7.4	27	28	28
s_3	$F_n(s_3, 2, 5)$	1.018	1.1064	1.1429	1.0271	1.0615	1.0671
	$\%p_{min}$	25	25.2	21.2	27	20.8	22.2
s_4	$F_n(s_4, 2, 5)$	1.139	1.1138	1.0606	1.052	1.0656	1.0727
	$\%p_{min}$	55.4	60.6	68	31.8	27.2	25.4
s_5	$F_n(s_5, 2, 5)$	1.203	1.1879	1.1739	1.084	1.1077	1.1074
	$\%p_{min}$	28.6	22.6	22.4	34.2	30.6	30.2
s_6	$F_n(s_6, 2, 5)$	1.1062	1.0529	1.0306	1.149	1.132	1.14
	$\%p_{min}$	35.4	41.4	40.8	60	65.4	66.6

TAB. 4.4: *Estimation du rapport du risque sur l'oracle pour les fonctions s_1, s_2, s_3, s_4, s_5 et s_6, et les méthodes m_1, m_2 et m_3.*

- La méthode calibrée peut faire "mieux" en terme de risque que les méthodes de sélection de modèle à variance connue et estimée, même si les risque sont très proches. Cette constatation est renforcée par la valeur de $\%p_{min}$. Nous avons choisi des constantes universelles égales à 2 et 5. Mais pour un n et une fonction s fixés, il se peut que ces constantes ne soient pas les plus adéquates même si elles sont proches des optimales. La méthode semble rectifier la constante multiplicative 2 (qui permet le passage de la pénalité minimale à la pénalité optimale) afin de sélectionner l'estimateur de perte minimale.

L'objectif de cette étude était d'appréhender, de tester et de calibrer la méthode proposée par Birgé et Massart sur des signaux Gaussiens, cadre classique pour lequel des résultats théoriques précis existent. D'après les résultats obtenus, la méthode mène à des résultats concluants. Cela permet d'envisager son utilisation pour certaines problèmes plus complexes. En effet, dans notre cadre d'étude, nous savons que la constante de pénalité α représente la variance σ^2. Cependant, pour d'autres problèmes, si le terme de pénalité est de la forme $\beta f_n(D)$ avec f_n une fonction bien définie d'un point de vue selection de modèle, il se peut que le paramètre β soit théoriquement indéfini. Dans un contexte de sélection de modèle, cette méthode permet alors d'obtenir facilement une bonne valeur de β, et donc une bonne pénalité à partir des données.

4.6.2 Comparaison avec MCMC

Nous appliquons la méthode calibrée sur l'exemple proposé dans la première partie de la thèse concernant la détection de ruptures par méthodes MCMC. Nous donnons la fonction s associée en Figure 4.13 – (a). L'estimateur pénalisé est représenté Figure 4.13 – (b).

Les instants de ruptures non renormalisés estimés sont $\hat{t}_1 = 76$, $\hat{t}_2 = 147$, $\hat{t}_3 = 256$ et $\hat{t}_4 = 400$. Nous obtenons le même estimateur que l'estimateur MAP obtenu par l'algorithme d'Hasting-Metropolis.

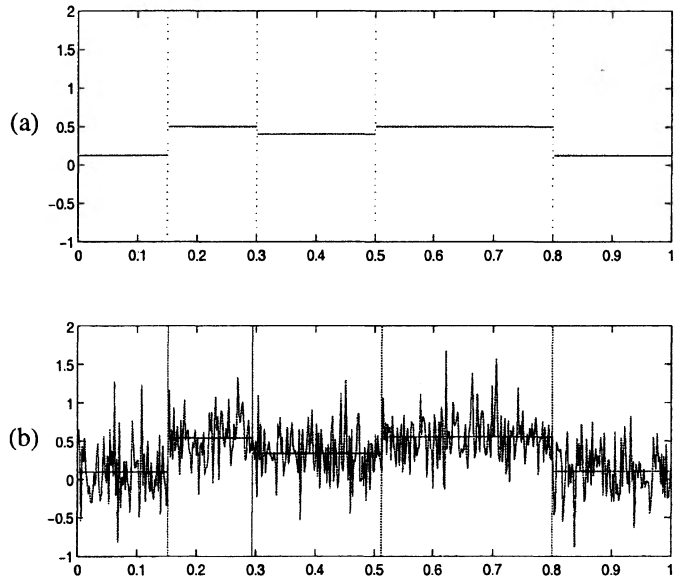


FIG. 4.13: *La fonction s (a) et la représentation graphique de l'estimateur pénalisé (b).*

4.6.3 Cas d'une fonction s non constante par morceaux

Dans cette sous-section, nous souhaitons répondre à la question suivante : que se passe-t'il quand la fonction s n'appartient à aucun modèle $\mathcal{S}_m$, c'est-à-dire quand elle n'est pas constante par morceaux?

Nous nous intéressons à deux fonctions ayant des profils différents : la première fonction, notée s_7, a un comportement constant par morceaux à partir de l'instant environ de 0.45 et la seconde, notée s_8, possède "deux bosses" d'amplitudes différentes. Ces deux fonctions sont représentées respectivement Figures $4.14 - (a)$ et $4.15 - (a)$. Pour la fonction s_7, nous espérons repérer au moins les deux instants de ruptures 0.45 et 0.7, et pour la fonction s_8, une partition composée de deux sous-partitions régulières, avant et après l'instant 0.5.

Nous considérons deux réalisations du processus y avec $n = 1000$, les deux fonctions s_7 et s_8 et les variances respectives $\sigma^2 = 1$ et $\sigma^2 = 0.1$. Les réalisations et les estimateurs pénalisés obtenus sont représentés sur un même graphique : Figure $4.14 - (b)$ pour la première (s_7) et Figure $4.15 - (b)$ pour la seconde (s_8). Les estimateurs que nous avons obtenus sont les estimateurs de perte minimale.

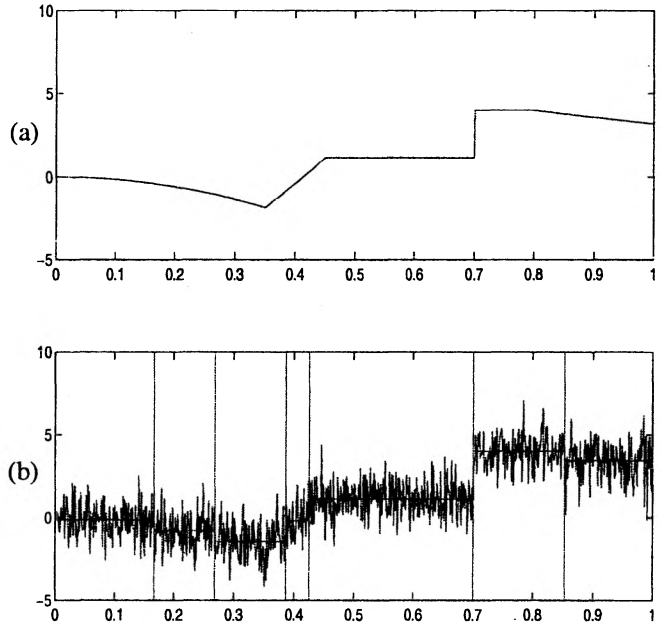


FIG. 4.14: *Fonction s_7 (a) et représentation graphique de l'estimateur pénalisé (b).*

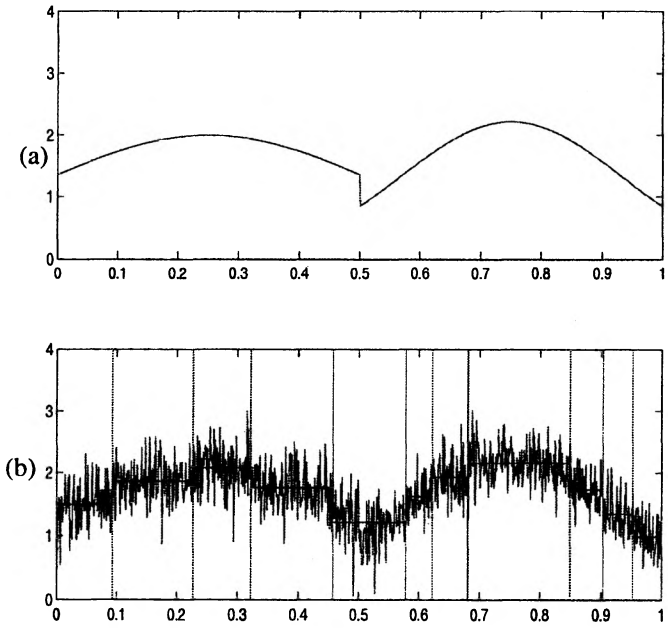


FIG. 4.15: *Fonction s_8 (a) et représentation graphique de l'estimateur pénalisé (b).*

L'impression laissée par ces résultats est que la méthode mène à des estimateurs ayant de bonnes qualités d'approximation.

4.6.4 Cas d'un bruit non Gaussien

Dans cette sous-section, nous présentons les résultats obtenus de l'application de la méthode calibrée sur des processus non Gaussiens :

- ε est un bruit blanc Exponentiel symétrique ($Expo$).

- ε est un bruit blanc Bernouilli de paramètre $1/2$ (Ber).

Nous comparons aux résultats obtenus pour des processus Gaussiens de variance $\sigma^2 = 1$.

En pratique, nous obtenons la même qualité de résultat pour les bruits blancs Gaussien et Bernoulli : l'estimateur de perte minimale est sélectionné. Le bruit Bernoulli de paramètre $1/2$ est faible et les ruptures sont très nettes. Par contre, pour un certain nombre de réalisations simulées à partir du bruit Exponentiel, la méthode sous-pénalise (la pénalité est trop petite pour pénaliser correctement) et une partition de trop grande dimension est sélectionnée. Le nombre important de fois où nous observons ce phénomène laisse penser que les constantes de la pénalité ne sont pas adaptées.

Nous proposons d'illuster ces constatations à l'aide de trois réalisations choisies du processus y avec $n = 500$, une fonction s simulée aléatoirement, et représentée Figure $4.16-(s)$, et les différents bruits. Nous présentons en Figures $4.16-(Gaus)$, $4.16-(Expo)$ et $4.16-(Ber)$ respectivement les réalisations et les estimateurs pénalisés obtenus.

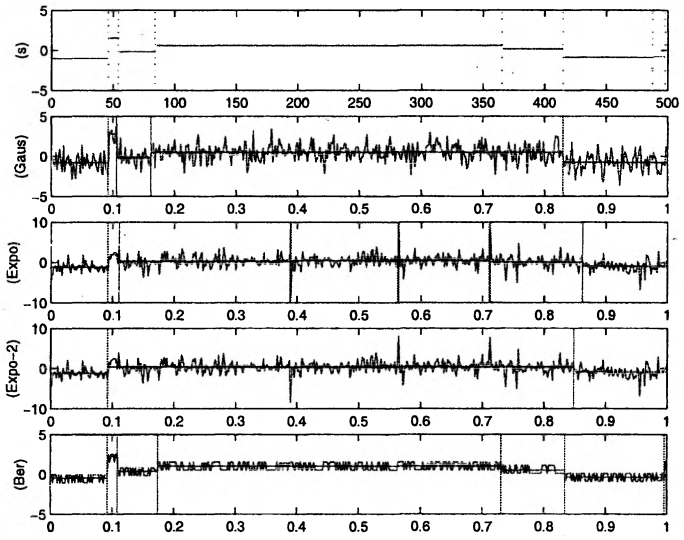


FIG. 4.16: *Fonction s (s) et représentation graphique des estimateurs pénalisés dans le cas d'un bruit blanc Gaussien (Gaus), Exponentiel (Expo), Exponentiel avec double pénalité (Expo − 2) et Bernoulli (Ber).*

Portons notre attention sur le résultat obtenu pour la réalisation simulée à partir du bruit Exponentielle. La partition de dimension 10 est sélectionnée et la partition associée à l'estimateur de perte minimale est 3. Nous remarquons que 6 des ruptures obtenues sont très proches et donc pas toutes nécessaires.

Considérons la pénalité définie pour tout $D \geq 1$ par :

$$pen_{\mu,n}(D) = \mu\hat{\alpha}\frac{D}{n}\left(\log\frac{n}{D} + 2.5\right),$$

et $\hat{D}_\mu$ la dimension qui minimise le critère définie pour tout $D \geq 1$ par :

$$crit_n(D) = \gamma_n(\hat{s}_D) + pen_{\mu,n}(D).$$

Le tableau 4.5 regroupe les dimensions sélectionnées pour différentes valeurs de μ.

μ	2	2.5	3	4	$2\sigma^2$	$4\sigma^2$
$\hat{D}_\mu$	10	9	5	3	9	3

TAB. 4.5: *Les dimensions sélectionnées à partir de différentes pénalités pour la réalisation issue du bruit Exponentiel.*

La méthode est normalement appliquée avec $\mu = 2$. Prendre $\mu = 4$ signifie que nous doublons la pénalité optimale obtenue par la méthode calibrée. Nous indiçons les résultats

obtenus à partir de cette double pénalité par $(Expo-2)$. L'estimateur pénalisé est représenté Figure 4.16 – $(Expo-2)$. Il appartient au modèle de dimension 3. Cette pénalité semble donc plus appropriée.

Nous réalisons une étude de simulation plus complète. Nous considérons cinq fonctions simulées aléatoirement et estimons pour chacune le rapport du risque de l'estimateur sur l'oracle $F_n(s,2,5)$ défini en (4.6.11). Les résultats sont donnés dans le tableau 4.6.

	s_1 (8)	s_2 (9)	s_3 (6)	s_4 (7)	s_5 (8)
$F_n(s,2,5)^{(Gaus)}$	1.18	1.05	1.08	1.15	1.21
$F_n(s,2,5)^{(Expo)}$	2.56	2.15	2.63	2.4	1.89
$F_n(s,2,5)^{(Expo-2)}$	1.29	1.34	1.5	1.13	1.1
$F_n(s,2,5)^{(Ber)}$	1.0008	1.004	1.002	1.0005	1.0003

TAB. 4.6: *Risque sur oracle pour les différentes fonctions s et les différents bruits. s(.) représente la taille de la partition sur laquelle a été construite la fonction s, i.e. le nombre de ruptures +1.*

Nous obtenons les mêmes conclusions que pour l'exemple présenté : la méthode fonctionne bien pour un bruit de Bernouilli de paramètre 1/2, mais ne donne pas de bons résultats pour un bruit Exponentiel.

4.6.5 Deux extensions de la méthode non calibrée

Nous avons tenté de calibrer la méthode de façon à obtenir une méthode complètement automatique qui donne de "bons" résultats dans une majorité de situations. Mais il est toujours très difficile d'obtenir une calibration qui fonctionne dans toutes les situations.

Dans la sous-section 4.6.5.1, nous donnons des configurations rencontrées en pratique dans lesquelles la méthode ne sélectionne pas l'estimateur de perte minimale, dont certaines sont liées à la calibration. Dans la sous-section 4.6.5.2, nous proposons deux extensions de la méthode non calibrée qui tentent de répondre aux problèmes évoqués dans la sous-section 4.6.5.1. Nous revenons aux problèmes qui nous ont motivé à calibrer la méthode. Ces extensions donneront à l'utilisateur une aide quand au choix de l'estimateur face à certaines configurations.

4.6.5.1 Discussion sur certaines configurations

En pratique, nous avons pu observer les phénomènes suivants :

1. La calibration peut s'avérer être trop "stricte" dans le sens où la bonne constante de pénalité minimale n'appartient pas à l'intervalle de calibration $[\beta_{seuil}\hat{\sigma}^2, \hat{\sigma}^2]$. Deux

raisons à cela: soit une trop grande ou trop petite valeur de β_{seuil}, soit une sur-estimation ou une sous-estimation trop importante de la variance σ^2. Dans ce cas, la méthode peut soit sélectionner l'estimateur de perte minimale car le $\hat{\alpha}$ n'est pas "loin" de la bonne constante de pénalité minimale, soit ne pas sélectionner le bon estimateur. Pour illustrer cette dernière configuration, nous considérons une réalisation du processus y de taille $n = 300$ simulée à partir d'une fonction s simulée aléatoirement et d'une variance $\sigma^2 = 0.5$. L'estimateur de la variance donné par Hall *et. al* est $\hat{\sigma}^2 = 0.82$. En Figure 4.17 est représentée la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$. La valeur de α associé au plus grand saut de dimension n'appartient pas à l'intervalle de calibration $[\beta_{seuil}\hat{\sigma}^2, \hat{\sigma}^2]$. Cette valeur est notée $\hat{\alpha}$ sur la Figure 4.17. En appliquant alors la méthode calibrée, la partition de dimension 7 est sélectionnée tandis que sans calibration c'est la partition de dimension 10 qui est sélectionnée, partition associée à l'estimateur de perte minimale.

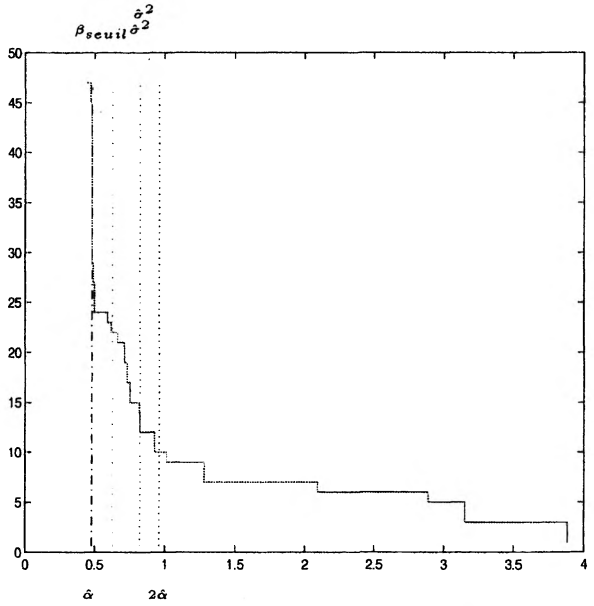


FIG. 4.17: *Graphe de la fonction* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$.

2. Le saut maximal de dimensions de la suite $(D_i)_{1 \leqslant i \leqslant K}$ peut être très faible (par exemple 2 ou 3, voir 4). Ce saut n'étant pas très marqué, comment savoir si il correspond bien au passage à la pénalité minimale? Ces configurations sont souvent rencontrées pour des petits échantillons car dans ce cas, le contraste $\gamma_n(\hat{s}_D)$ est très souvent logarithmique en $f_n(D)$.

3. L'estimation de la constante de la pénalité optimale $2\hat{\alpha}$ peut "tomber" sur un tout petit intervalle $[\alpha_i, \alpha_{i+1}[$ de α ou "juste à côte" du bon intervalle. Nous illustrons cette dernière configuration à l'aide d'une réalisation simulée. La fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$ est représentée Figure 4.18. La quantité $2\hat{\alpha}$ vaut 0.98 et la dimension sélectionnée est

7. Cependant, $2\hat{\alpha}$ est proche du $\alpha_i = 0.99$ qui est associé à la dimension 6, dimension de l'estimateur de perte minimale. Il suffirait de considérer $2.1\hat{\alpha}$ sur cet exemple pour obtenir le bon estimateur.

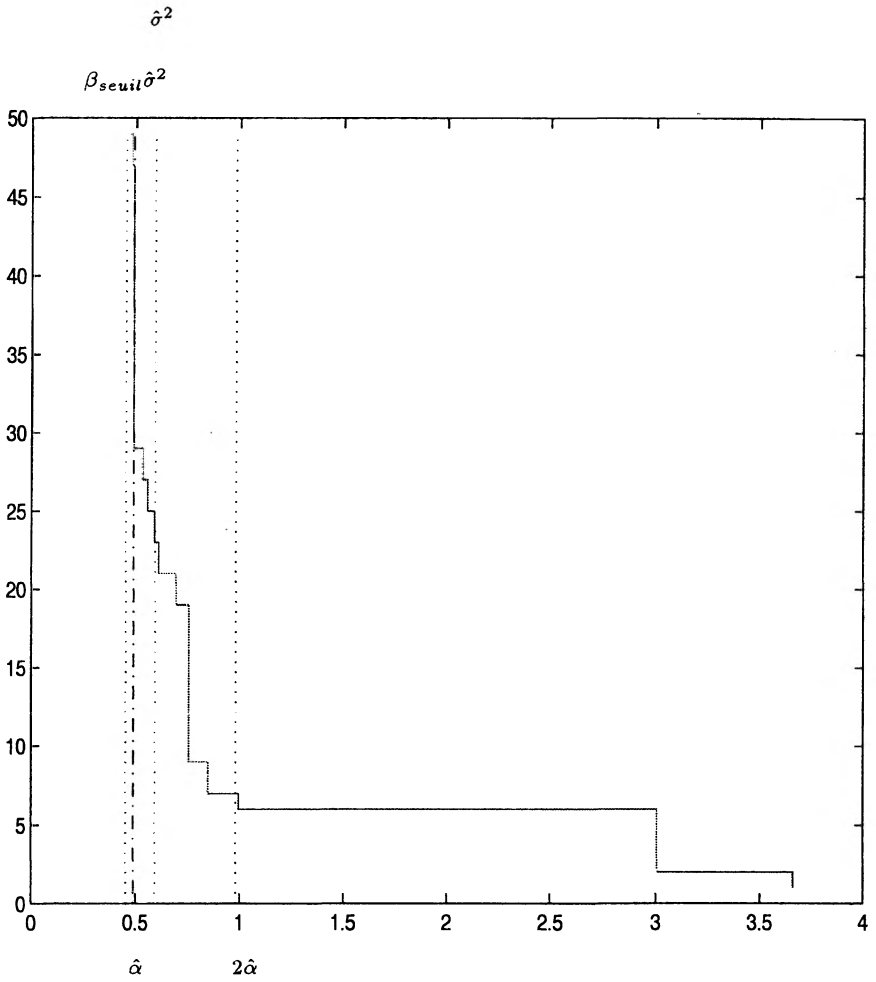


FIG. 4.18: *Graphique de la fonction* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i,\alpha_{i+1}[}$.

4.6.5.2 Discussion sur la méthode non calibrée

1. • La raison principale de la calibration de la méthode était que la valeur de la dimension maximale D_{max} peut fortement influencer la sélection de la partition. L'idée est de faire varier D_{max} par pas de 1 jusqu'à une valeur choisie par l'utilisateur, et pour chaque valeur, d'appliquer la méthode non calibrée : une dimension est sélectionnée pour chaque valeur. Nous illustrons cette procédure sur l'exemple considéré dans le premier point de la sous-section précédente en Figure 4.19. Nous donnons de plus une représentation des sauts de dimensions

de la suite $(D_i)_{k=1,\dots,K}$ en Figure 4.20. Nous pouvons lire sur ce graphique qu'il existe un grand saut de dimension qui passe de la dimension 45 à la dimension 29. Il faut prendre une valeur de D_{max} suffisamment grande pour pouvoir considérer ce saut. L'estimateur associée à la partition de dimension 10 est sélectionnée. C'est une situation idéale.

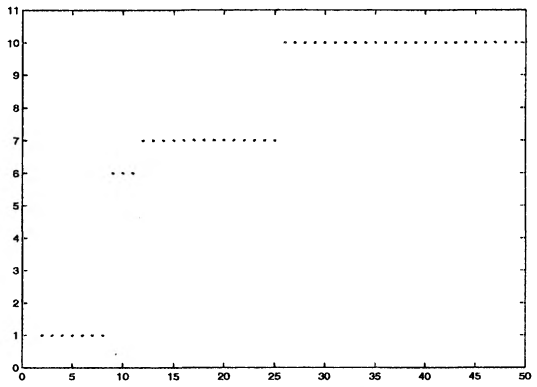


FIG. 4.19: *Représentation graphique de $\hat{D}$ en fonction du D_{max}.*

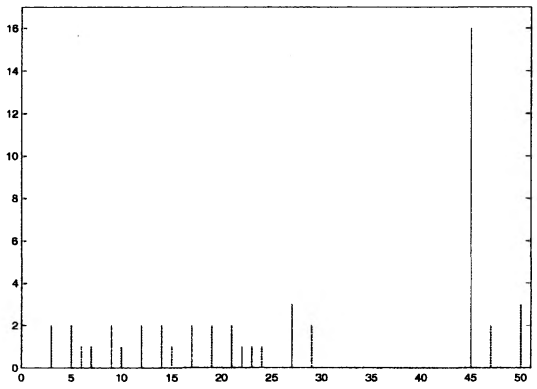


FIG. 4.20: *Représentation des sauts de dimensions de la suite $(D_i)_{k=1,\dots,K}$.*

S'il existe maintenant deux grands sauts de dimensions dans la suite $(D_i)_{k=1,\dots,K}$, lequel choisir? Nous illustrons cette configuration par les Figures 4.21 et 4.22 qui sont issues de l'application de la méthode sur l'exemple de la sous-section 4.6.2. Notons que sur cet exemple, l'estimateur de perte minimale est associé à la partition de dimension 3. Si l'utilisateur choisit $D_{max} < 30$, le plus grand saut de dimensions est égal à 9 et le partition sélectionnée est de dimension 3. Pour $D_{max} \geq 30$ (D_{max} élevé par rapport à la dimension 3), le plus grand saut de dimensions est égal à 13 et la partition sélectionnée est de dimension 5. Notons que les pertes des deux estimateurs associés sont très proches. Les deux estimateurs sont donc envisageables.

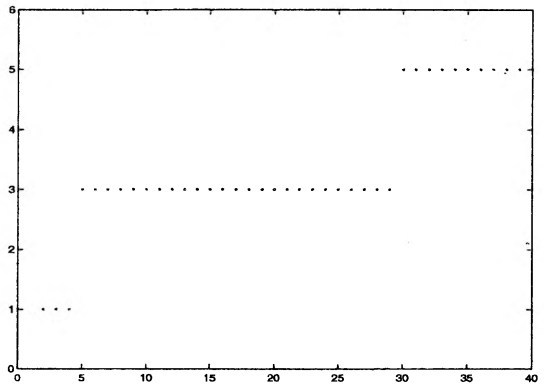


FIG. 4.21: *Représentation graphique de $\hat{D}$ en fonction du D_{max}.*

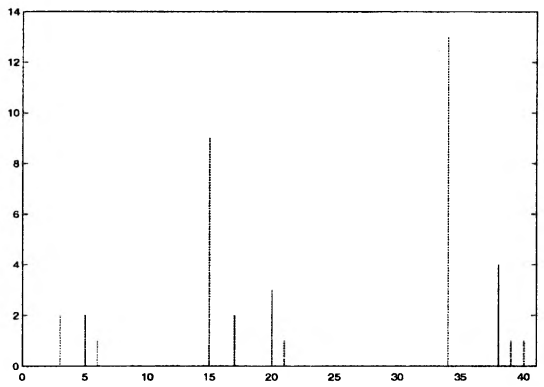


FIG. 4.22: *Représentation des sauts de dimensions de la suite $(D_i)_{k=1,\dots,K}$.*

Le choix sera bien sûr plus difficile si la taille maximale des sauts est très faible et atteinte par plusieurs sauts.

- La méthode prend en compte le premier grand saut de dimension rencontré. Nous donnons ici deux cas où il existe deux grands sauts de dimensions dont un seul est pris en compte par la méthode alors qu'il peut être intéressant de considérer le deuxième.

  - Dans le premier cas, le plus grand saut de dimension est le deuxième rencontré. Les Figures 4.23 et 4.24 illustrent ce cas. La partition de dimension 2 est sélectionnée alors qu'avec le second plus grand saut, c'est la partition de dimension 25, partition la plus proche de la partition associée à l'estimateur de perte minimale. Notons que la dimension de cette partition n'appartient pas à la suite $(D_i)_{k=1,\dots,K}$.

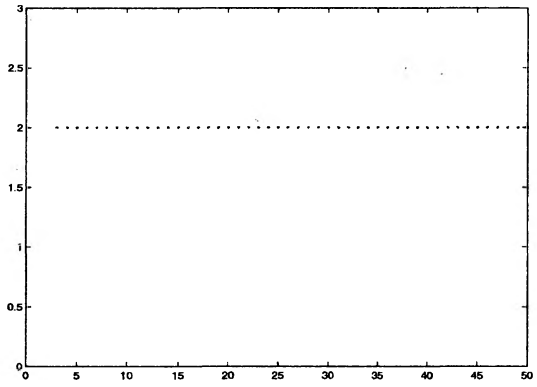


FIG. 4.23: *Représentation graphique de $\hat{D}$ en fonction du K_{max}.*

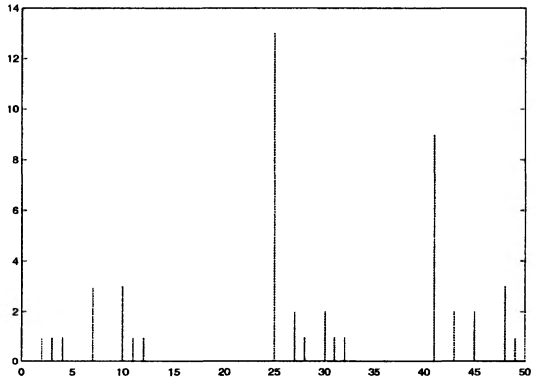


FIG. 4.24: *Représentation des sauts de dimensions de la suite $(D_i)_{k=1,\ldots,K}$.*

- Dans le second cas, les deux plus grands sauts de dimensions sont de même dimension. Les Figures 4.25 et 4.26 illustrent ce cas. La partition de dimension 9 est sélectionnée avec le premier saut alors qu'avec le second plus grand saut, c'est la partition de dimension 6, partition la plus proche de la partition associée à l'estimateur de perte minimale.

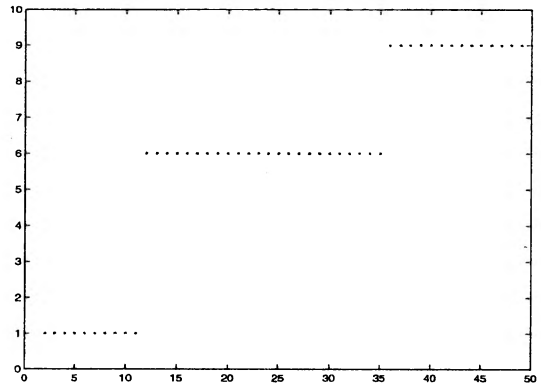


FIG. 4.25: *Représentation graphique de $\hat{D}$ en fonction du K_{max}.*

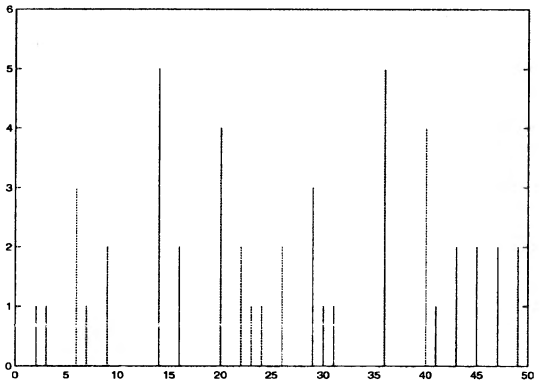


FIG. 4.26: *Représentation des sauts de dimensions de la suite $(D_i)_{k=1,\dots,K}$.*

2. Si l'utilisateur obtient des configurations comme présentées dans le troisième point de la sous-section précédente, il peut être interessant de considérer la partition la plus proche au sens de α (voir les deux plus proches), *i.e.* la partition obtenue en considérant dans la pénalité le α_i de la suite $(\alpha_i)_{1 \leqslant i \leqslant K}$ le plus proche de $2\hat{\alpha}$.

4.7 Application: détection des changements dans le nombre mensuel de tests HIV en France

Notre objectif est d'identifier des changements dans le comportement des français face au virus HIV.

Nous disposons du nombre de tests HIV effectués chaque mois en France entre le mois de Février 1987 et le mois d'Octobre 1991. Un nombre de tests constant sur un intervalle de temps signifie que le comportement des français face au virus est le même durant la période considérée.

Nous considérons les données comme Gausiennes et indépendantes, et nous cherchons à détecter des ruptures dans la moyenne de cette suite de données, chaque rupture déterminant les débuts et fins des intervalles de temps durant lesquels le nombre de tests HIV reste constant.

Nous appliquons la méthode sans calibration. Le graphe de la dimension sélectionnée $\hat{D}$ obtenue par la méthode pour différentes valeurs de D_{max} est donné Figure 4.27 et les sauts de dimensions sont représentés Figure 4.28.

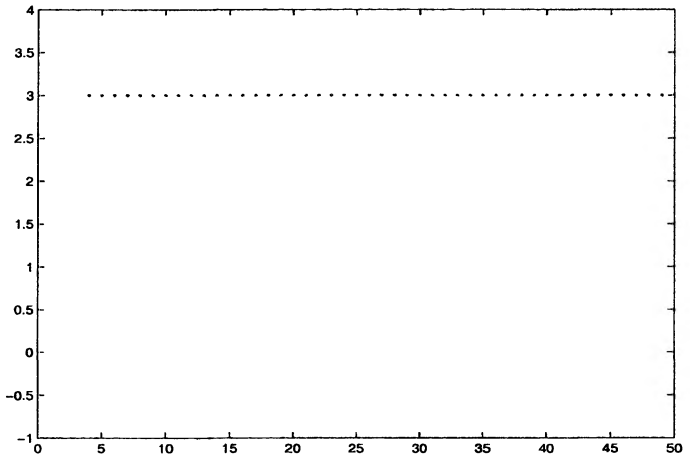


FIG. 4.27: *Représentation graphique de $\hat{D}(\alpha)$ en fonction du D_{max}.*

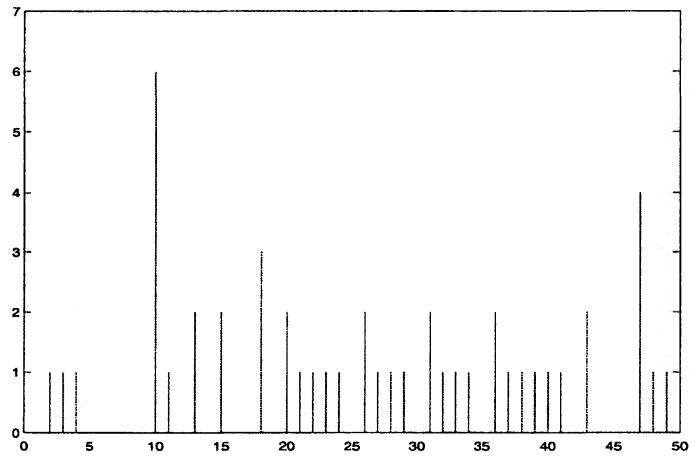


FIG. 4.28: *Représentation des sauts de dimensions de la suite $(D_i)_{k=1,\dots,K}$.*

D'après ces graphes et le second point du paragraphe 4.6.5.2, nous considérons tout

d'abord la partition de dimension 3, puis celle associée au deuxième saut de dimension qui est de dimension 27. Les deux estimateurs respectifs sont représentés Figure 4.29.

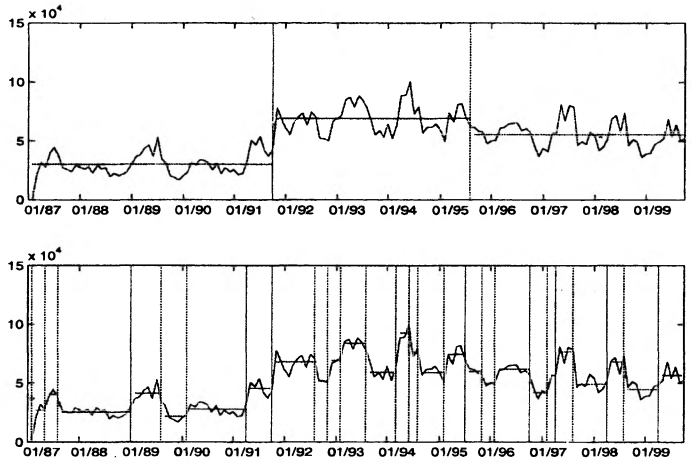


FIG. 4.29: *Représentation graphique des estimateurs pénalisés obtenus associés respectivement aux deux plus grands sauts.*

La partition de dimension 3 montre trois plages constantes du nombre de tests. Il apparaît qu'un plus grand nombre de tests a été effectué entre le mois d'Octobre 1991 et le mois d'Août 1995. Cette période semble correspondre à la prise de conscience de la gravité du virus et sa progation. Nous observons ensuite une légère baisse. La partition de dimension 27 semble révèler l'existence d'un cycle annuel du nombre de tests : il y a plus de tests effectués entre le debut du printemps et la fin de l'été que le reste de l'année. Cela peut s'expliquer par le nombre de mariages qui sont effectués essentiellement pendant cette période.

Chapitre 5

A CART based Algorithm for Detection of Multiple Change Points in the Mean for Large Samples .

Ce chapitre présente un travail en collaboration avec Servane Gey. Ce travail a fait l'objet d'un article qui a été soumis au journal Computational Statistics and Data Analysis.

Résumé :

Nous proposons ici un algorithme basé sur les méthodes de sélection de modèles pour la détection de ruptures multiples dans la moyenne d'un signal gaussien de grande taille. D'un point de vue algorithmique, cette méthode est implémentée à l'aide d'un algorithme dynamique qui fonctionne bien que sur des échantillons de taille n modérée ($n \leqslant 5000$). Pour des échantillons de taille supérieure, nous proposons un algorithme combinant l'algorithme CART et une recherche exhaustive partielle. L'idée est, dans un premier temps, de réduire le nombre de configurations possibles en ne gardant que les plus appropriées et, dans un deuxième temps, d'effectuer une recherche exhaustive sur ces configurations pour obtenir une configuration finale proche de l'optimale.
Nous donnons des exemples d'application de cet algorithme sur des données simulées. Une étude de simulation compare les performances des différents algorithmes en termes de risques et de temps de calculs.

Mots clés: DÉTECTION DE RUPTURES – SÉLECTION DE MODÈLE – ALGORITHME CART

A CART based Algorithm for Detection of Multiple Change Points in the Mean for Large Samples

S. Gey [1] and E. Lebarbier

Abstract

The main interest of this paper is to provide an algorithm for the application of a model selection method on the detection of multiple change-points in the mean for very large Gaussian signals. From an algorithmical point of view, visiting all the configurations of change points cannot be performed on large samples. We propose an algorithm combining the CART algorithm and a partial exhaustive search. The idea is first to reduce the number of configurations of change-points by keeping the relevant ones and then to run the exhaustive search in order to obtain a configuration close to the optimal one. Some examples of application to simulated data are given. A simulation study compares the performance of the different algorithms in term of risk and on the other hand in term of computational time.

Keywords : CHANGE-POINT DETECTION – MODEL SELECTION – CART ALGORITHM

5.1 Introduction

The change-points problem has attracted much attention for more than fourty years (see the books of Basseville and Nikiforov [4] and Brodsky and Darkhovsky [14] for a complete bibliography on change detection).
We focus here on the problem of detecting changes in the mean in a sequence of independant normal random variables. The mean value of the observations is assumed to be constant for a while and changes at some change-point instants to another value. The

[1]Université Paris Sud, France

problem consists in detecting and locating the change-points and estimating the magnitude of the jumps. We shall adopt a model selection via penalization approach in which the unknown change-points are estimated by minimizing a suitable penalized criterion. The first authors developing this kind of method are Mallows [47] and Akaike [1], [2]. It is now classical in the literature. In the particular context of detecting change-points in a piecewise constant signal, Yao [66], Miao *et. al* [51] estimate the number of jumps in the mean via the Schwarz' criterion (see Schwarz [59]). More recently Lavielle and Moulines [42] propose to detect and estimate consistenly all those change points, while Lavielle and Lebarbier [41] propose a Bayesian approach of this problem.
In [43] Lebarbier adopts a model selection approach in a non asymptotic context based on Birgé, Massart [7]. The change-points and the means are simultaneously estimated by recovering the underlying piecewise constant function. In this context, the estimation of the function is optimal in term of risk, that means that some change-points could be ignored if the jump in the mean at these instants is too small. The estimator is given by the minimization of a penalized least-squares criterion. In a computational point of view, the procedure, called here exhaustive search, uses a dynamic programming algorithm which requires $\mathcal{O}(n^2)$ operations, if n is the length of the observed signal. While this algorithm works well for moderatly sized signals, $n \leqslant 5000$ in our framework, it cannot be performed on much larger signals.
Our aim in this paper is to propose an extension of this method to process on large samples : we want to reduce drastically the computation time without altering too much the accuracy of the estimation. To proceed, a first idea could be to make a deterministic splitting and then apply the exhaustive search on each piece. In another point of view, Vostrikova [65] and more recently Chong [19] propose some sequential likelihood ratio tests algorithms providing some consistent estimators of the configuration of change-points. From this viewpoint, many authors (as for example Picard [54] or Ghorbanzadeh [30]) propose some methods to test the existence of a change-point. On the other hand, from a model selection point of view, a sequential algorithm will be constructed as the algorithm of Chong [19] despite the fact that it will not test any hypothesis but simply minimize the least-squares criterion at each step.
We are here interested in the CART algorithm (Classification And Regression Tree) developed by Breiman *et al.* [13] whose first stage is a sequential one in the same sense as mentioned above. This gives some potential configurations of change-points instants that are revisited by the second stage of the algorithm that permits to reduce the collection of configurations of change-points by keeping the relevant ones. CART is computationally fast (its computational complexity is often of order $\mathcal{O}(n\log(n))$). Nevertheless, it may add some false alarms. By running an exhaustive search restricted to the configurations associated to the relevant candidates, the false alarms can be removed. So we propose an algorithm combining the CART algorithm and a partial exhaustive search that permits to select in a faster way a configuration of change-points close to the optimal one. A simulation study has been performed to compare the performance of the hybrid algorithm with CART and the exhaustive search in term of risk and computational time.
The paper is organized as follows. The Section 5.2 describes the model and the notations and recall basics about penalized least-squares. The Sections 5.3, 5.4, 5.5 and 5.6 deal with

the CART and exhaustive search methods : Section 5.3 describes how to generate "good" configurations, Section 5.4 gives the motivations for the hybrid algorithm from a simple example for illustration, Section 5.5 is dedicated to the convenient choice of the penalty function while Section 5.6 answers the question : how to choose the final configuration ? Then we propose in Section 5.7 an illustration of the final algorithm combining CART and a partial exhaustive search, and the performance are studied in Section 5.8.

5.2 Preliminaries and Notations

Let us consider the following model

$$y_t = s(t) + \varepsilon_t \quad t = 1, \ldots, n \tag{5.2.1}$$

where the errors (ε_t) are supposed to be zero-mean, identically distributed unobservable Gaussian independant random variables of common variance σ^2. The function s is assumed to be piecewise constant. Thus, there exists some instants $t_0 = 0 < t_1 < \ldots < t_K = n$ such that the function s is constant between two successive instants. In other words, there exists a sequence $(s_1, \ldots, s_K)$ such that

$$s = \sum_{k=1}^{K} s_k \mathbb{1}_{I_k} \quad \text{with } I_k =]t_{k-1}, t_k] \tag{5.2.2}$$

This model means that $K-1$ changes affect the mean of (Y_t) at some unknown instants $(t_k, 1 \leqslant k \leqslant K-1)$. Moreover, the number of change-points is treated here as unknown. Our main interest is to estimate the change-point instants of y. To proceed, we estimate the unknown function s, leading to the unknown number of segments K, the configuration of change-points (t_k) and the means (s_k). Indeed, the relation between the function s and the configurations of change-points is simple : the configuration of change-points (t_k) represents for the function s the partition $\bigcup_{k=1}^{K} I_k$ on which it is constructed. So in the sequel, one assimilates the estimator of the function s and the estimator of the configuration of change-points.

The general method of estimation considered here is based on a penalized least-squares criterion. We shall take the model selection point of view developed in Birgé, Massart [7] to extract some optimal penalty function in the sense of the quadratic risk. This means that we do not want to determine the true function s but the one which fits the best s in the sense of the quadratic risk. For example, if s presents 10 change-points, and if the best minimal risk estimator of s is the one having no change-points, then we want to select this one rather than the estimator of s having 10 change-points.

Hence let us recall model selection principle for which we introduce some notations used in the sequel.
We associate to each configuration of change-points $(t_1, \ldots, t_K)$ the corresponding partition of $\{1, \ldots, n\}$ having $K+1$ pieces.

For any given partition of $\{1, \ldots, n\}$

$$m = \bigcup_{k=1}^{D_m} I_k$$

of dimension D_m, we define $\mathcal{S}_m$ to be the linear set of piecewise constant functions defined on the partition m.

The minimum least-squares estimator $\hat{s}_m$ of s on S_m is defined as follows :

$$\begin{aligned} \hat{s}_m &= \operatorname*{argmin}_{u \in \mathcal{S}_m} \gamma_n(u) \\ &= \sum_{k=1}^{D_m} \overline{y}_k \mathbb{1}_{I_k} \end{aligned}$$

where $\overline{y}_k$ is the empirical mean of y on I_k and γ_n, the empirical contrast, is the least-squares criterion defined for all fonction $u \in \mathbb{L}^2([1, n])$ by

$$\gamma_n(u) = \frac{1}{n} \sum_{t=1}^{n} (y_t - u(t))^2. \tag{5.2.3}$$

Then, to select a good partition, we propose to extract from a chosen family $\mathcal{M}_n$ of partitions of $\{1, \ldots, n\}$ a partition giving an estimator of s whose performance are close to optimal in the sense of the quadratic risk.
Hence for each partition $m \in \mathcal{M}_n$ one can compute the minimum least-squares estimator $\hat{s}_m$ of s. This leads to a collection of estimators $\{\hat{s}_m, m \in \mathcal{M}_n\}$. Then we want to choose an estimator among this collection using a penalized criterion : the idea consists in adding a penalty term to the least-squares criterion in order to avoid over-segmentation. We consider in our framework a classical criterion used in Gaussian model selection with a penalty term depending on the partition m via its dimension D_m :

$$crit_n(m) = \gamma_n(\hat{s}_m) + pen_n(D_m). \tag{5.2.4}$$

Thus the idea is to consider the partitions of same dimension to extract a subcollection $\widetilde{\mathcal{M}}_n$ of $\mathcal{M}_n$ having at most one partition per dimension. Let us denote by $\hat{m}_D$ the best partition for a fixed dimension D, that is

$$\hat{m}_D = \operatorname*{argmin}_{\{m \in \mathcal{M}_n ; D_m = D\}} \{\gamma_n(\hat{s}_m)\} \tag{5.2.5}$$

Then we choose the final estimator as

$$\tilde{s} = \hat{s}_{\hat{m}_{\hat{D}}}$$

where

$$\hat{D} = \operatorname*{argmin}_{D \in \widetilde{\mathcal{M}}_n} [\hat{m}_D + pen_n(D)].$$

In a model selection context, the penalty function should be chosen in an optimal way in term of risk according to the collection $\mathcal{M}_n$.

5.3 How to generate Good Partitions ?

We propose two algorithms : an exhaustive search considering all possible partitions of and CART regression trees considering a relevant subcollection of the maximal one. According to the procedure of each algorithm, $\widetilde{\mathcal{M}}_n$ is then not the same for each case.

5.3.1 Exhaustive Search

The first natural approach is to consider all the possible partitions of the grid $\{1, \ldots, n\}$ where n is the size of the sample. Let us denote by $\mathcal{M}_n^{(es)}$ this family. Remark that $\mathcal{M}_n^{(es)}$ is the maximal family of partitions of $\{1, \ldots, n\}$. For a given dimension D, searching the best D-dimensional partition among $\mathcal{M}_n^{(es)}$ amounts minimizing in $t_1, t_2, \ldots, t_{D-1}$

$$\sum_{k=1}^{D} \sum_{t=t_{k-1}+1}^{t_k} (y_t - \overline{y}_k)^2. \tag{5.3.6}$$

This minimization is given by $\binom{n-1}{D-1}$ operations, that means a computational complexity of order $\mathcal{O}(n^D)$. To reduce the computational load to a more manageable level we employ the technique of dynamic programming. We refer the reader who wants more details about this algorithm to the book of Kay [37]. For a sake a completness, let us give here a short summary of this algorithm.

The dynamic programming algorithm takes advantage of the additivity of the constrast function. It can be seen as the search of the shortest path to travel from a point to another, which can be defined recursively. Indeed, let set

$$\Delta(t_{k-1} + 1 : t_k) = \sum_{t=t_{k-1}+1}^{t_k} (y_t - \overline{y}_k)^2$$

and let define for $D < E \leqslant n$

$$I_D(E) = \min_{t_0=0<t_1<t_2<\ldots<t_{D-1}<t_D=E} \sum_{k=1}^{D} \Delta(t_{k-1} + 1 : t_k)$$

$\Delta(t_{k-1} + 1 : t_k)$ is the least squares error calculated on $[t_{k-1} + 1, t_k]$ and $I_D(E)$ is so the minimal least squares error for $D - 1$ change-points between 1 and E.
Then, one can easily show that $I_D(E)$ can be decomposed as

$$I_D(E) = \min_{t_{D-1}} \{I_{D-1}(t_{D-1}) + \Delta(t_{D-1} + 1 : E)\}.$$

This says that the minimum error for $D - 1$ change-points is the minimum error for the first $D - 2$ change-points that end at t_{D-1} plus the error contributed between the instants

$t_{D-1}+1$ and n.
So dynamic programming does not require to evaluate all the considered partitions and then reduces the computational complexity to $\mathcal{O}(n^2)$ (see subsection 5.8.3 for more details).

We use this procedure to select among $\mathcal{M}_n^{(es)}$ a subcollection $\widetilde{\mathcal{M}}_n^{(es)}$ having one partition per dimension.

5.3.2 CART Regression Trees

The CART algorithm (Breiman *et al.* [13]) is primarily computed in two steps. We focus here on the first step, called the growing procedure, consisting in constructing recursively a large collection of partitions using a data-dependent dyadic splitting. It is computed as follows :

- At the beginning compute the change-point $\hat{t}_c$ by minimizing in $t_c \in \{1, \dots, n\}$ the total squared error

$$\sum_{t=1}^{t_c} (y_t - \bar{y}_L)^2 + \sum_{t=t_c+1}^{n} (y_t - \bar{y}_R)^2$$

  where $\bar{y}_L$ and $\bar{y}_R$ are the empirical means respectively on $[1, t_c]$ and $[t_c+1, n]$. Computed this way, $[1, \hat{t}_c] \bigcup [\hat{t}_c + 1, n]$ corresponds to the best partition (in terms of least-squares criterion) of $\{1, \dots, n\}$.

- Apply the same procedure on I_L and I_R respectively, and so on until the number of points $\{t\}_{1 \leqslant t \leqslant n}$ contained in each resulting segment is smaller than a given minimal number l_{min}.

Recall that the output of such a procedure is usually represented by a binary tree, called the deepest tree (see the bottom of Figure 5.1 in Section 5.4). Then any partition obtained from this tree corresponds to a "pruned subtree", that is any subtree of the deepest one containing its root. This leads to a first large subcollection of partitions $\mathcal{M}_n^{(\text{cart})}$ corresponding to all the subtrees pruned from the deepest one. According to the hierarchical structure of a tree, it is clear that $\mathcal{M}_n^{(\text{cart})} \subset \mathcal{M}_n^{(es)}$ with no equality in general. This phenomenon is supported by the computational complexity of these algorithms. Indeed, the computational complexity of the growing procedure is typically of order $\mathcal{O}(n \log n)$ (see subsection 5.8.3).

Then, among the large collection of trees $\mathcal{M}_n^{(\text{cart})}$, one can compute the best tree for each dimension $D \in \{1, \dots, n\}$, that leads to the expected subcollection $\widetilde{\mathcal{M}}_n^{(\text{cart})}$. But this computation is of exponential order. Hence, some partitions of this subcollection are irrelevant since they will be eliminated when we will latter penalize. A faster procedure (see subsection 5.8.3 for more details about its computational complexity) called pruning does implicitly this two steps and avoids the computation of $\widetilde{\mathcal{M}}_n^{(\text{cart})}$ by selecting directly the relevant partitions. This procedure used in practice is recalled in subsection 5.5.2.

5.4 Motivations for an hybrid algorithm

In this section, we propose a simple example to illustrate the motivation of using an algorithm combining CART and the exhaustive search.

We simulate a signal from model (1.2.1) of size $n = 40$. There exist three change-points at times 10, 15 and 30. This observed signal with the function s to recover are plotted in the top of Figure 5.1.

First, assume that the number of change-points is known to be three. The dimension of the associated partition is then $K = 4$. So we are interested by the partitions of dimension 4 minimizing the empirical quadratic contrast (5.2.3) for the two algorithms proposed in section 5.3. Let us denote these ones by $\hat{m}_4^{(es)}$ and $\hat{m}_4^{(cart)}$ respectively. We obtain $\hat{m}_4^{(es)} = (11, 15, 30)$ and $\hat{m}_4^{(cart)} = (15, 17, 30)$.

In the exhaustive search algorithm we visit all the possible partitions of dimension 4. So, $\hat{m}_4^{(es)}$ is undoubtly the exact global mimimum solution over the 4-dimensional partitions. Since CART is sequential in the first stage, it does not guarantee finding the global minimum. For example, according to Table 5.1, the empirical contrast of $\hat{m}_4^{(es)}$ is smaller than the one of $\hat{m}_4^{(cart)}$. This behaviour is explained by the hierarchical structure of the collection $\mathcal{M}_n^{(\text{cart})}$ represented by a tree plotted on the bottom of Figure 5.1. The $*$ on the tree correspond to the times of the signal. Let us explain shortly the tree representation from this example : the first split is obtained at time 17. Then the left part of the signal is cut at time 15 and the right one at time 30, and so on. With this representation, a pruned subtree of dimension $D - 1$ corresponds to a partition of dimension D and always contains the first split (at time 17 in this example). For example, the possible partitions of dimension 3 are $(15, 17)$ or $(17, 30)$. The main drawback of this procedure is precisely its hierarchical structure. Indeed, this algorithm can add or shift some change-points. For example, if the first split is a false alarm, then it will be kept in all the partitions considered by CART and to obtain the "good" partition one has to consider the one of larger dimension. We observe this phenomenon on the proposed example : the best partitions of dimensions 4 and 5 are respectively $(15, 17, 30)$ and $(11, 15, 17, 30)$. According to the true configuration and the one obtained by the exhaustive search, the change-points at time 17 is not relevant. So we must consider the 5-dimensional partition to obtain the three interesting change-points.

On the other hand, the computational time of the dynamic programming occuring in the exhaustive search is in $\mathcal{O}(n^2)$. Consequently the main drawback of this procedure is that it cannot be performed on large signals. Whereas the CART algorithm allows us to keep the relevant configurations with a computational time in mean of the order of $\mathcal{O}(n \log n)$. For example, to obtain the best partition of dimension 3, the exhaustive search compares $n - 1$ partitions whereas the CART algorithm compares only 2 partitions.

So, we propose an hybrid algorithm combining the CART algorithm and a partial exhaustive search by only keeping the advantages of each of them. The general idea is to perform

the CART algorithm in oder to obtain a family of relevant configurations of change-points (or partitions) and then to run an exhaustive search on this one to free ourselves of its hierarchic structure. For the example, it consists in considering $(11, 15, 17, 30)$ as a new grid and running the exhaustive search on it. Using this scheme, we hope that the exhaustive search removes the change-point at time 17.

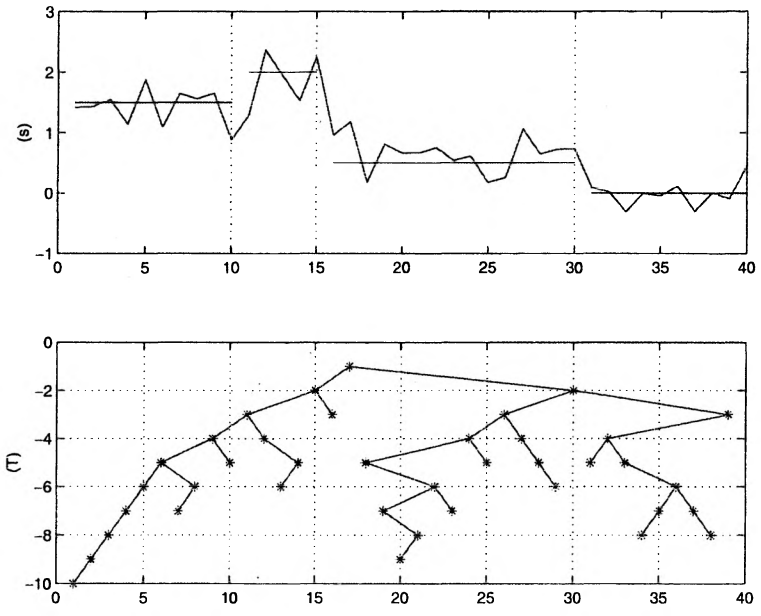


Figure 5.1: An example of a signal (s) and its associated CART tree (T).

Here we consider that the number of change-points, i.e. the size of the partition, is unknown. We then use a penalized criterion which is now classical in the literature. We recall shortly the role of the penalty function : we list in Table 5.1 the best partitions of dimension D, $D = 1, \ldots, 6$ and their associated empirical contrast values given by the two algorithms run on the previous example. Let us focus for example on the exhaustive search algorithm. It is clear that the more we take change-points, the smaller the contrast is. As we can see in Table 5.1, overfitting occurs : the values of γ_n for the partitions of dimension 4 and 5 are close. That means that it does not bring anything new to consider the 5-dimensional partition (for CART the same behaviour occurs for the partitions of dimensions 5 and 6). This is why, when the number of change-points is unknown, a penalty term is added to the contrast in order to control the number of detected change-points. The contrast and the penalty evolve in opposite sense with respect to the dimension D and the best partition is the one making the better compromise between these two terms.

	Exhaustive Search		CART Regression Trees	
	configuration of change-points	γ_n	configuration of change-points	γ_n
$\hat{m}_1$	no change-point	0.493	no change-point	0.493
$\hat{m}_2$	(17)	0.154	(17)	0.154
$\hat{m}_3$	(15,30)	0.099	(17,30)	0.101
$\hat{m}_4$	(11,15,30)	0.072	(15,17,30)	0.09
$\hat{m}_5$	(11,15,17,30)	0.063	(11,15,17,30)	0.063
$\hat{m}_6$	(9,11,15,17,30)	0.056	(9,11,15,17,30)	0.056

Table 5.1: The best partitions and their associated least squares values γ_n obtained for the example by running the exhaustive search and CART regression trees.

5.5 Penalization

We recall in this section the two penalty functions optimal in term of risk for the two different algorithms. Their optimality have been studied respectively in [43] and [28].

5.5.1 Penalty Function for Exhaustive Search

In [43], Lebarbier showed that the penalty function

$$pen_n(m) = \frac{D_m}{n}\sigma^2(2\log\frac{n}{D_m} + 5) \tag{5.5.7}$$

performs well in a theoretical way. However, the main practical issue is to estimate σ^2. An alternative method avoiding the estimation of σ^2 is recalled in section 5.6.

5.5.2 Penalty Function for CART Regression Trees

In addition to the penalty function, we shall describe here the second step of the CART algorithm (pruning procedure) since they are closely related.

The penalty is taken of the form

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

where β is an unknown constant.

The corresponding criterion (5.2.4) also depends on β and will be written as

$$crit_{n,\beta}(m) = \gamma_n(\hat{s}_m) + \beta\frac{D_m}{n}. \tag{5.5.8}$$

The pruning procedure, generally used in practice, gives directly the relevant partitions of $\mathcal{M}_n^{(\text{cart})}$ and avoids the construction of the collection $\widetilde{\mathcal{M}}_n^{(\text{cart})}$. The general strategy is to make β increase so that D_m decreases by minimizing recursively some function of the nodes. This leads to a collection of nested trees $(m_i)_{1\leqslant i\leqslant K_T}$, with $m_{K_T} = [0,1]$, associated with an increasing sequence of temperatures $(\beta_i)_{1\leqslant i\leqslant K_T}$, with $\beta_1 = 0$. Moreover, for all $\beta_i \leqslant \beta < \beta_{i+1}$, the partition minimizing $crit_{n,\beta}$ is exactly m_i. For more details about this algorithm see Breiman *et al.* [13].

It remains to choose a tree among the collection $(m_i)_{1\leqslant i\leqslant K_T}$, i.e. to reach a suitable value of β. Section 5.6 gives a method to do this.

5.6 How to choose the final partition ?

5.6.1 Heuristic method : General Idea

Let us recall that the problem is to calibrate the penalty function having the general form for a given $D \in \widetilde{\mathcal{M}}_n$

$$pen_{\beta,n}(D) = \beta f_n(D)$$

where $f_n(D)$ is a suitable function beforehand defined such that the associated penalty function leads to a good value of the risk.
The aim of this section is to use a heuristic method, based on Birgé and Massart [8], which allows us to estimate the penalty function itself in a close to optimal way, i.e to find a suitable value of β. This method is based on the fact that the contrast of an estimator is of the order of the sum of two terms : a first term which represents some approximation error within the associated model, i.e a bias term, and a second term which represents some estimation error, i.e a variance term which has the form of the penalty function. Then the idea is that when the considered model is high-dimensional, the estimation bias is zero, so the contrast of the estimator of such a model will represent an estimation of the variance term.
The basic principle is to fit a linear regression of $\gamma_n(\hat{s}_D)$ with respect to $f_n(D)$ for large D and use the estimated regression coefficient as an estimator of $-\beta/2$. Then, in order to get a good penalty, it suffices to take $pen_{\hat{\beta},n}$.

However, it occurs that the sequence $\gamma_n(\hat{s}_D)$ is not an affine function of $f_n(D)$ and behaves logarithmic, so this heuristic cannot be performed so easily. The final method proposed by Birgé and Massart answer to this problem. Through theoretical results and some practical observations, they conclude to the following heuristic method : let set $\alpha = \beta/2$. Then for a fixed α, consider the model $\hat{D}_\alpha$ defined by

$$\hat{D}_\alpha = \underset{D\geqslant 1}{\operatorname{argmin}} \left\{\gamma_n(\hat{s}_D) + pen_{\alpha,n}(D)\right\}.$$

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

5.6.2 The Heuristic applied to each Algorithm

CART algorithm

Up to now, the general methods used in the CART algorithm 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* [13]). 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 one proposed above.
According to the property recalled in subsection 5.5.2 and the fact that $f_n(D) = D/n$, it is sufficient here to perform a linear regression of $(n\gamma_n(m_i))_{1\leqslant i\leqslant K_T}$ against $(D_i)_{1\leqslant i\leqslant K_T}$ where D_i is the dimension of the partition m_i.
In practice, for large n, we observe that beyond some dimension, $n\gamma_n(\hat{s}_{m_i})$ becomes an affine function of the number of segments. We apply this procedure on the example plotted in figure 5.5-(a) in section 5.7. The corresponding function $D_i \to n\gamma_n(\hat{s}_{m_i})$ is plotted in figure 5.2. 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 $n\gamma_n(\hat{s}_{m_i})$ becomes linear. That is why this choice is let to the user.

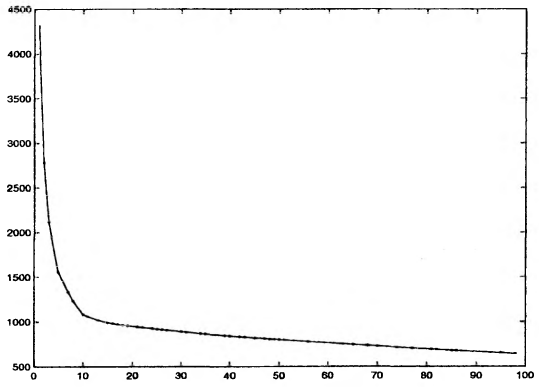


Figure 5.2: Function $D_i \to n\gamma_n(\hat{s}_{m_i})$ for $i = 1, \ldots, K_T$

Remark 7. We could have hoped to be able to fine tune the optimal penalty constant β for this algorithm as done for the constants 2 and 5 for the exhaustive search. But it appears to be impossible since the collection $\mathcal{M}_n^{(\text{cart})}$, and then the number of models of same dimension occuring in the penalty, depends heavily on the observations.

Exhaustive search

In this algorithm, it appears that $\gamma_n(\hat{s}_D)$ is not always an affine function of $f_n(D)$. According to the previous subsection we compute the function $\alpha \to \hat{D}_\alpha$ (plotted in Figure 5.3) 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. However, Lebarbier proposes in [43] a calibration of the value α which allows us to use this method in all of possible situations.

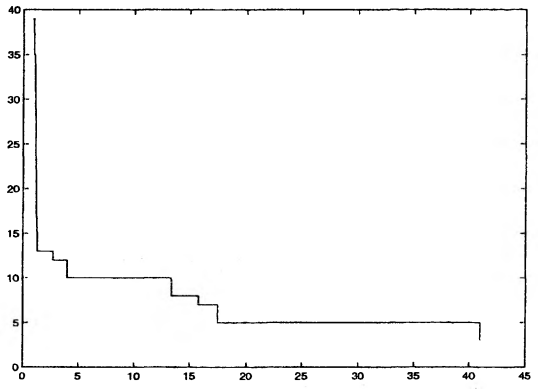


Figure 5.3: Function $\alpha \to \hat{D}_\alpha$ for $\alpha \geqslant 0$

5.7 Illustration of the Hybrid Algorithm

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

1. The CART algorithm is performed and a partition of dimension denoted by $\hat{D}_c$ is obtained by the heuristic given in subsection 5.6.2.

2. We consider the subtree in the sequence $(m_i)_{1 \leqslant i \leqslant K_T}$ which number of segments is larger than $v\hat{D}_c$ 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 instants. Let us denote $K_{\mathcal{L}}$ its size.

3. We have at hand successively the collection of partitions $\mathcal{M}_{n,\mathcal{L}} = \mathcal{P}(\mathcal{L})$ constructed on $\mathcal{L}$, the associated collection of models $\{S_m, m \in \mathcal{M}_{n,\mathcal{L}}\}$ and the corresponding family of estimators $\{\hat{s}_D, D = 1, \ldots, K_{\mathcal{L}}\}$ where for a given dimension D

$$\hat{s}_D = \underset{\{u \in S_m ; m \in \widetilde{\mathcal{M}}_{n,\mathcal{L}}\}}{\operatorname{argmin}} \gamma_n(u).$$

   Then an exhaustive search is performed on the family $\mathcal{M}_{n,\mathcal{L}}$ providing the final estimator $\tilde{s}$.

Remark 8. We take the subtree having $v\hat{D}_c$ leaves in order to catch the relevant change-points instants which could be eventually missed or shifted by the first selection. To be more precise about the choice of v, let us notice that if the ratio $\hat{D}_c/n$ is small and 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 just remark that this value should not be chosen too large to keep the interest of the relevant reduction of collection of partitions done by CART.

We propose an illustration of the different steps of this algorithm and then compare its performance with the two other ones. We simulate from (5.2.1) a sequence of $y = (y_1, \ldots, y_n)$ with $n = 1000$ according to a function s plotted in Figure 5.4 and a variance $\sigma^2 = 1$. The observed serie is plotted in Figure 5.5-(a).

Remark 9. 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 algorithm. However, the interest of the hybrid algorithm is to run it in the cases in which the exhaustive search cannot be performed, i.e in the cases of large samples.

We apply the three proposed algorithms on this realization. The penalized estimators are plotted in Figures 5.5 and 5.6. To show the dynamic of the hybrid algorithm, we give a short description of the different results : first, the penalized estimator obtained by CART is plotted in Figure 5.5-(a). The dimension of its associated partition is $\hat{D}_c = 15$. We take $v = 4$ and the corresponding subtree is displayed Figure 5.5-(c). Then an exhaustive search is performed on the new associated grid and its penalized estimator is plotted in Figure 5.5-(b). The symbols represented on the tree in Figure 5.5-(c) (o, $*$ and $+$) correspond respectively to the change-points removed, kept and added after running the exhaustive search on the tree given by the CART algorithm.

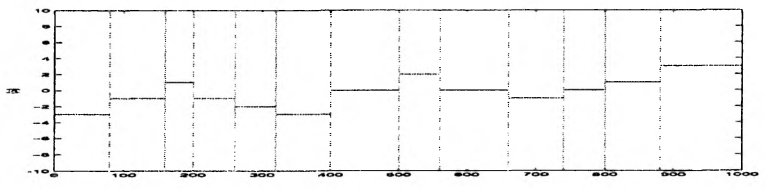


Figure 5.4: The function s.

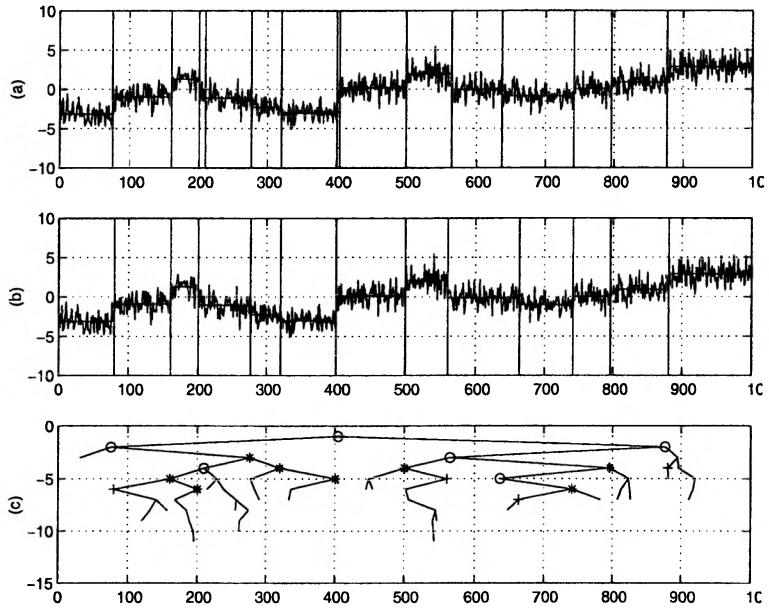


Figure 5.5: Penalized estimators obtained respectively by (a) CART and (b) hybrid algorithms and (c) the tree with the corresponding change-points where o : removed change-points, $*$: kept change-points and $+$: added change-points

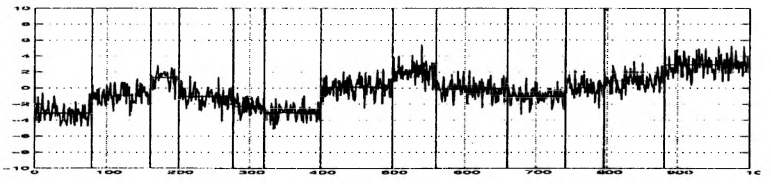


Figure 5.6: Penalized estimator obtained by exhaustive search.

First of all, we can notice in this example that the hybrid algorithm behaves as expected, i.e. that the exhaustive search permits to remove the change-points added by CART and to catch the missed or shifted ones : for example, CART adds 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 4 change-points at 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 algorithm 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 $\|s-\tilde{s}\|^2$ of each penalized estimator $\tilde{s}$ provided by the algorithms, we find 0.11, 0.04 and 0.038 respectively for CART, hybrid algorithm and

exhaustive search. So we can deduce on this example that the hybrid algorithm improves the performance of CART and does not really alter the ones of the exhaustive search. Let us notice that the penalized estimators obtained by the hybrid an the exhaustive search are the penalized estimators of minimal loss among the corresponding and respective collections of partitions.

5.8 Simulation study and Computational Complexities

In this section, we give in a first part some numerical results to compare on one hand the performance of some estimators computed thanks to the different algorithms and, on the other hand, the computational time taken by each of them. Then in a second part, we give some approximations of the computational complexities of the CART and exhaustive search algorithms.

5.8.1 Simulation study

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

- $R_{(.)}$ the risk of the estimator $\tilde{s}_{(.)}$:

$$R_{(.)} = \mathbb{E}(\|s - \tilde{s}_{(.)}\|_n^2)$$

  Since we were not able to reach analytically the exact values of $\mathbb{E}(\|s - \tilde{s}_{(.)}\|_n^2)$, we compute them by a Monte Carlo method, averaging the values of $\|s - \tilde{s}_{(.)}\|_n^2$ over N samples.

- $cput_{(.)}$ the average computation time of the (.) algorithm, 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 (.) algorithm for the jth sample.

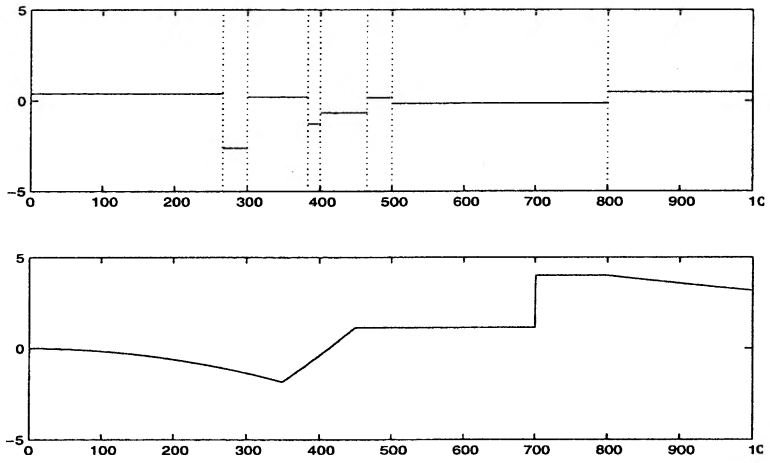


Figure 5.7: The functions s_2 and s_3

This simulation study is performed with Matlab6 scientific software on an Ultra 10-440 MHz SUN workstation. We take the following various parameters : $n = 1000$, $\sigma^2 = 1$ and 3 functions s_1, s_2 and s_3 respectively plotted in Figures 5.4 and 5.7. A more elaborate discussion about the function s_3 and the results obtained by the application of the algorithms on this function is given in the next section.
Moreover, the number of simulated samples to estimate the considered values is $N = 300$ and the regression used in the heuristic method for CART (see section 5.6) is made on the dimensions from 20 to 40. Furthermore, we set $v = 4$ in the hybrid algorithm.
The results are given in Table 2.

	s_1	s_2	s_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 5.2: Results about the estimation of the penalized estimators risk and of the computational time of each agorithm for the three proposed functions.

So we can conclude from these numerical results that :

1. The estimators obtained by the hybrid algorithm are not so far in terms of risk from the estimators obtained by an exhaustive search on the whole sample, but perform better than the ones obtained by CART.
2. The hybrid algorithm takes much less operations than an exhaustive search.

Remark 10. Here we have taken $l_{min} = 1$ to ensure that the close change-points will be detected. However, this leads to a deepest tree having n nodes. So the growing procedure will visit all the instants, but even in that case the computation time is improved. So there is a trade-off to make in the choice of l_{min} between the computation time and the accuracy of the detection.

5.8.2 Additional discussion

In this subsection, we want to answer the question : what happens when the function s does not belong to one of the collection of model S_m, i.e is not a piecewise constant function ? Let us focus on the function s_3 (see Figure 5.7). Indeed, this function present a piecewise constant behaviour only from the change-point around time 450. It is clear that we expect to detect at least the two change-points at 450 and at 700.

We propose an example to show the behaviour of the three algorithms in front of this kind of function. We simulate a sequence $y = (y_1, \ldots, y_n)$ with $n = 1000$ and the variance of the additive noise is $\sigma^2 = 1$. The observed signal is plotted in Figure 5.8.
The penalized estimators obtained by the CART, the hybrid and the exhaustive search algorithms are respectively plotted in Figures 5.8-(a), 5.9 and 5.10. Moreover, the corresponding estimators loss functions are 0.0408, 0.0483 and 0.0477.
First of all, the first impression about the results is that the partitions obtained from these algorithms seem to be natural according to the function s_3.
Then, according to the estimators loss, we can remark that the penalized estimator obtained by CART performs better than the ones obtained by the hybrid and the exhaustive search algorithms. Moreover, let us notice that this estimator has a smaller loss than the estimator having the minimal loss in the exhaustive search. This phenomenon can be explained by the fact that the CART algorithm acts locally while the others search in a more global way. However, the risk of the estimators computed for this function and given in Table 2 are closer.

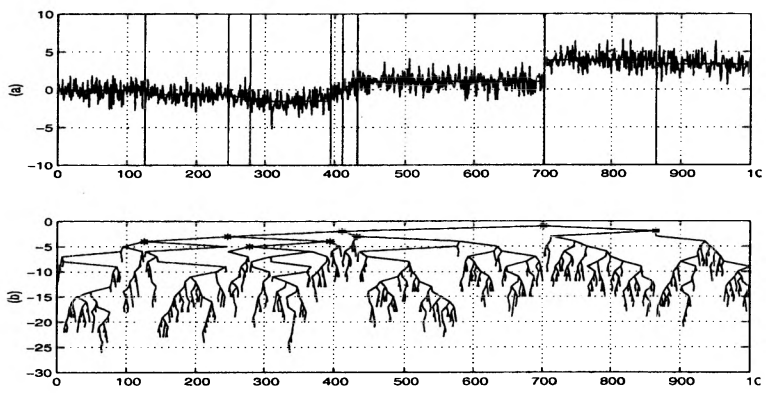


Figure 5.8: Penalized estimator obtained by CART with the deepest tree.

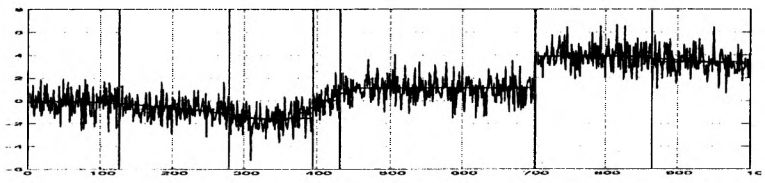


Figure 5.9: Penalized estimator obtained by the hybrid algorithm.

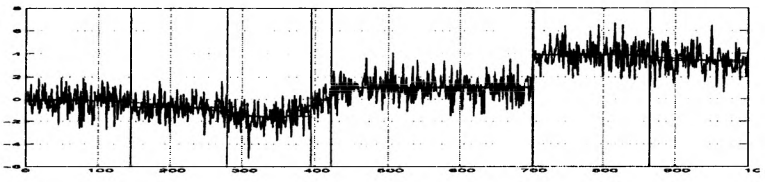


Figure 5.10: Penalized estimator obtained by exhaustive search.

We propose then to simulate a sequence y with a smaller variance $\sigma^2 = 0.5$. The penalized estimators obtained by the three algorithms belong to the model of dimension 8. The ones obtained by CART and hybrid are plotted in Figure 5.11. The only difference with the ones obtained by the exhautive search is that the change-point 410 becomes 412. The losses of the estimators are then very close to each other.

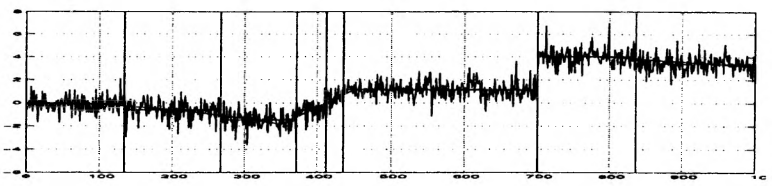


Figure 5.11: Penalized estimator obtained by the CART and hybrid algortihms.

5.8.3 Computational complexities

In this subsection, we give some results on the computational complexities of the two different algorithms previously introduced. These ones are computed for a sample of size n. In fact, we obtain an exact value for the deterministic complexity of the exhaustive search algorithm, and some bounds depending on n and the random number n_t of nodes of the tree considered for the CART algorithm.

Proposition 5.8.1. *Assume that we have at hand a n sample. Then the complexity $C(n)$ of the exhaustive search algorithm is*

$$C(n) = \mathcal{O}(n^2) \tag{5.8.9}$$

Proposition 5.8.2. *Assume that we have at hand a sample of size n. Let denote $n_t \leqslant n$ the number of nodes of the deepest tree constructed during the first step of the CART algorithm. Let $C_1(n, n_t)$ and $C_2(n_t)$ be the respective complexities of the growing procedure and the pruning procedure. Then we have*

$$\mathcal{O}(n \log_2 n_t) \leqslant C_1(n, n_t) \leqslant \mathcal{O}(n n_t) \tag{5.8.10}$$

$$\mathcal{O}(n_t) \leqslant C_2(n_t) \leqslant \mathcal{O}(n_t^2) \tag{5.8.11}$$

The proofs are given in Appendix.

Remark 11. Let us notice that the computational complexity C_2 of the pruning procedure depends only on the number n_t of nodes of the deepest tree, whereas the one of the growing procedure depends on n_t and n. But this is expected since the pruning procedure is performed on a fixed tree. In fact, 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 algorithms used to construct a suitable tree is the same as the one of the exhaustive search algorithm. To reach this computational complexity during the CART algorithm, it is necessary to obtain after the growing procedure a complete thread-like binary tree having exactly n nodes. In this case, it is clear that, by using the hybrid algorithm, 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. Indeed, the results obtained in the table above show that the number of operations of the hybrid algorithm is smaller than the one of the exhaustive search.

5.9 Conclusion

We have proposed an algorithm for the change-points problem with large samples in a model selection context for which an exhaustive search cannot be performed in practice. Numerical experiments have clearly exhibit that the hybrid algorithm leads in a much faster manner to an estimator close in terms of risk to the one given by an exhaustive search.
The main advantage of this procedure is the ability to be applied on large samples. Moreover, we think that this kind of approach can be performed in other frameworks in which the size of the observed sample is too large.
This algorithm has yet been used in the genome framework for the detection of homogeneous area in DNA sequence [43].

Appendix

PROOF OF THE COMPLEXITY OF THE EXHAUSTIVE SEARCH ALGORITHM 5.8.9:

Since this algorithm is composed by three steps, the computation of the complexity of the global algorithm will be the sum of the three complexities :

1. The collection of estimators $\{\hat{s}_{\hat{m}(D)}, D = 1, \ldots, n\}$ is obtained from a dynamic algorithm which has a complexity of $\mathcal{O}(n^2)$.
2. The complexity of the computation of the function $\alpha \to \hat{D}_\alpha$ is $\mathcal{O}(n)$. Morevover the estimation of α needs only one operation, so the complexity is of the order $\mathcal{O}(1)$.
3. Since the best partition is selected among the collection $\{\hat{s}_D, D = 1, \ldots, n\}$, the complexity of this step is $\mathcal{O}(n)$.

Since the complexities of the two last steps are significantly smaller than the first, the complexity of the exhaustive search algorithm is $\mathcal{O}(n^2)$.

PROOF OF THE COMPLEXITY OF THE GROWING PROCEDURE 5.8.10:

Since the complexity of this algorithm depends on the form of the constructed tree, we can not give it by an explicit term. However, we can bound it by considering two cases :

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

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

  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 relation

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

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

PROOF OF THE COMPLEXITY OF THE PRUNING PROCEDURE 5.8.11:

Contrary to the preceding, 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 leave by leave 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 relation

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

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

Troisième partie

Détection de ruptures dans la distribution marginale d'une suite de variables aléatoires discrètes par méthode de sélection de modèle et applications

Chapitre 6

Détection de ruptures dans la distribution marginale d'une suite de variables aléatoires discrètes par méthode de sélection de modèle

Ce chapitre présente un travail en collaboration avec Elodie Nedelec[1].

6.1 Introduction

Nous considérons une suite $(Y_t)_{1\leqslant t\leqslant n}$ de n variables aléatoires discrètes où Y_t est à valeurs dans l'ensemble fini d'entiers $\mathcal{Y} = \{1, \ldots, r\}$ avec $r \geq 2$. Des changements affectent la loi de cette suite : les variables aléatoires suivent la même loi sur chaque segment d'une partition de $\{1, \ldots, n\}$ et ont des lois différentes d'un segment à l'autre. Les changements ou ruptures correspondent aux bornes des segments de la partition, et l'objectif est d'estimer ces instants de ruptures.

Dans ce chapitre, nous nous intéressons exclusivement à une approche du minimum de contraste pénalisé. Parmi les auteurs qui considèrent cette approche, Braun et Müller [11] proposent d'estimer tous les instants de ruptures par minimisation d'un critère de Schwarz modifié. Ils montrent des résultats de consistance des estimateurs des instants de ruptures dans le cas où les variables aléatoires sont indépendantes. Lavielle propose également un critère pénalisé dans [40], il obtient des résultats de consistance dans le cas où ces variables sont faiblement ou fortement dépendantes.

L'approche que nous proposons ici est une approche non-asymptotique. Dans un premier temps, nous considérons le cas où les variables aléatoires sont indépendantes. Nous

1. Université Paris Sud, France

relions la loi de la suite de variables aléatoires à une fonction s. Les coefficients de cette fonction représentent les paramètres des lois sur chacun des segments de la partition. L'objectif est d'estimer la fonction s. Nous adoptons une méthode d'estimation non-paramétrique par sélection de modèle comme dans la partie II. Nous construirons l'estimateur de s par maximisation de vraisemblance pénalisée tel que cet estimateur est proche de s au sens du risque. Nous obtiendrons la forme de la fonction de pénalité en reprenant la démarche suivie par Birgé et Massart [48] dans le cadre de fonctions de contraste borné. Nous verrons que dans cette pénalité apparaît un terme logarithmique qui provient, comme dans le cadre des signaux Gaussiens (cf chapitre 1), de la complexité de la collection de partitions qui est la même ici puisque l'on considérera toutes les partitions de la grille $\{1, \dots, n\}$. Dans un second temps, nous considérons le cas où les variables aléatoires sont générées sur chaque segment de la partition selon une chaîne de Markov d'ordre 1. Nous donnons un résultat similaire au cas indépendant. Cette partie théorique est motivée par une application dans le cadre du génome concernant la recherche de régions homogènes sur des séquences d'ADN, qui fait l'objet du chapitre suivant.

6.2 Cas d'indépendance

6.2.1 Problème statistique

Nous considèrons n variables aléatoires discrètes indépendantes $Y_1, Y_2, ..., Y_n$ où Y_t est à valeurs dans l'ensemble fini d'entiers $\mathcal{Y} = \{1, \dots, r\}$ avec $r \geq 2$ pour $1 \leqslant t \leqslant n$. Soit $y = (y_1, y_2, ..., y_n)$ une réalisation du vecteur aléatoire $Y = (Y_1, Y_2, ..., Y_n)$ à valeurs dans $\mathcal{Y}^n$. Les observations étant indépendantes, la loi P de Y vaut :

$$P(Y_1 = y_1, Y_2 = y_2, ..., Y_n = y_n) = \prod_{t=1}^{n} P(Y_t = y_t).$$

Pour tout $t \in \{1, \dots, n\}$ et pour tout $i \in \mathcal{Y}$, nous posons :

$$P(Y_t = i) = s(t, i), \tag{6.2.1}$$

et nous notons P_s la loi de Y.

Nous supposons que s est constante par morceaux pour la variable t. Ainsi, il existe une partition m_0 de $\{1, ..., n\}$ et une suite $(s_I, I \in m_0)$, où $s_I = (s_I(1), s_I(2), \dots, s_I(r))$, telles que pour chaque segment $I \in m_0$,

$$s(t, .) = s_I(.) \qquad \text{pour tout } t \in I.$$

Ce modèle signifie que pour chaque segment $I \in m_0$, $Y_I = (Y_t)_{t \in I}$ est une suite de variables aléatoires indépendantes et identiquement distribuées de loi commune s_I. Les ruptures correspondent alors aux bornes de chaque segment de la partition m_0.

Nous écrivons s sous la forme suivante :

$$s = \sum_{I \in m_0} \sum_{i=1}^{r} s_I(i)\ \mathbb{1}_{I \times i}. \tag{6.2.2}$$

L'objectif est d'estimer s par une méthode de sélection de modèle.

Pour garder une homogénéité dans les indices utilisés, les segments de la vraie partition m_0 seront indicés par I, ceux de partitions quelconques m et m' de $\{1, ..., n\}$ seront respectivement indicés par J et J', et les valeurs que prend Y_t par i.

6.2.2 Estimateur du maximum de vraisemblance

Cet estimateur est un estimateur par minimum de contraste.

Nous nous fixons une partition m de $\{1, ..., n\}$ de dimension notée $|m|$. Nous définissons **le modèle** associé à la partition m par :

$$S_m = \left\{ \begin{array}{l} u = \sum_{J \in m} \sum_{i=1}^{r} u_J(i)\ \mathbb{1}_{J \times i} \\ \text{avec } u_J(i) \geq 0, \forall J \in m,\ \forall i \in \{1, \ldots, r\} \\ \text{et } \int_{\{1,..,n\} \times \{1,..,r\}} u(x)\ d\mu_n(x) = 1 \end{array} \right\}, \tag{6.2.3}$$

où μ_n désigne le produit de la mesure uniforme sur $\{1, \ldots, n\}$ par la mesure de comptage sur $\{1, \ldots, r\}$:

$$\mu_n = \mathcal{U}\{1, .., n\} \times \mathcal{C}\{1, .., r\}. \tag{6.2.4}$$

Soit $u \in S_m$, nous notons P_u la loi de Y associée à u par la relation donnée en (6.2.1) pour la vraie s. La vraisemblance associée est donnée par :

$$\begin{aligned} V(Y_1, ..., Y_n; u) &= P_u(Y_1, ..., Y_n) \\ &= \prod_{t=1}^{n} u(t, Y_t) \\ &= \prod_{J \in m} \prod_{t \in J} u_J(Y_t) \\ &= \prod_{J \in m} \prod_{t \in J} \prod_{i=1}^{r} u_J(i)^{\mathbb{1}_{Y_t = i}} \\ &= \prod_{J \in m} \prod_{i=1}^{r} u_J(i)^{N_J(i)}. \end{aligned} \tag{6.2.5}$$

où

$$N_J(i) = \sum_{t \in J} \mathbb{1}_{\{Y_t = i\}}, \tag{6.2.6}$$

est le comptage du nombre de Y_t prenant la valeur i dans le segment J, pour $J \in m$ et $i \in \{1, \ldots, r\}$.

Nous considérons le contraste empirique renormalisé associé à la log-vraisemblance. En considérant que Y représente un seul échantillon qui est un n-uplet, la fonction de contraste empirique est l'opposé de la log-vraisemblance et est alors définie pour une loi P_u par :

$$\gamma(P_u) = -\log[V(Y_1, ..., Y_n; u)]. \tag{6.2.7}$$

Il nous faut maintenant définir un contraste sur $\mathcal{S}_m$. Par la relation entre la loi P_u et $u \in \mathcal{S}_m$, nous pouvons définir un contraste empirique pour $u \in \mathcal{S}_m$ par :

$$\gamma'(u) = \gamma(P_u)$$

D'après l'egalité (6.2.5), le contraste empirique est défini pour tout $u \in \mathcal{S}_m$ par :

$$\gamma'(u) = -\sum_{J \in m} \sum_{i=1}^{r} N_J(i) \log(u_J(i)). \tag{6.2.8}$$

L'estimateur de s du minimum de contraste sur le modèle $\mathcal{S}_m$, donné en (6.2.3), est par la relation (6.2.7), l'estimateur du maximum de vraisemblance sur ce modèle. Il est noté $\hat{s}_m$ et est défini par :

$$\begin{aligned} \hat{s}_m &= \underset{u \in \mathcal{S}_m}{\operatorname{argmin}}\ \gamma'(u) \\ &= \sum_{J \in m} \sum_{i=1}^{r} \left(\frac{N_J(i)}{|J|} \right) \mathbb{1}_{J \times i}. \end{aligned} \tag{6.2.9}$$

Pour mesurer la qualité de l'estimateur, nous définissons une fonction de perte notée l et une fonction de risque associées au problème posé. Pour $s \in S_{m_0}$ la vraie et pour tout $u \in S_m$, la fonction de perte est donnée par :

$$l(s, u) = E_s[\, \gamma'(u) - \gamma'(s) \,] \tag{6.2.10}$$

En remarquant que le comptage $N_J(i)$ pour tout $J \in m$ et pour tout $i \in \{1, \ldots, r\}$, défini par (6.2.6), peut s'écrire comme la somme des comptages sur $|I \cap J|$ pour $I \in m_0$:

$$N_J(i) = \sum_{I \in m_0} N_{J \cap I}(i),$$

et d'après l'expression du contraste donnée en (6.2.8), nous avons pour tout $u \in S_m$ que :

$$\gamma'(u) - \gamma'(s) = \sum_{I \in m_0} \sum_{J \in m} \sum_{i=1}^{r} N_{J \cap I}(i) \log\left(\frac{s_I(i)}{u_J(i)} \right).$$

D'après l'équation (6.2.6), l'espérance de $N_{J\cap I}(i)$ s'obtient facilement et vaut pour tout $J \in m$, $I \in m_0$ et $i \in \{1, \ldots, r\}$:

$$E_s\left[N_{J\cap I}(i)\right] = |I \cap J| s_I(i). \tag{6.2.11}$$

La fonction de perte l est alors définie pour $s \in S_{m_0}$ et pour tout $u \in S_m$ par :

$$l(s,u) = \sum_{I\in m_0}\sum_{J\in m}|I\cap J|\sum_{i=1}^{r} s_I(i)\log\left(\frac{s_I(i)}{u_J(i)}\right) \tag{6.2.12}$$

Nous remarquons que si K est l'information de Kullback, alors la perte de l'estimateur $\hat{s}_m$ s'écrit comme une somme de Kullback :

$$l(s,u) = \sum_{I\in m_0}\sum_{J\in m}|I\cap J| K\left(s_I(.), u_J(.)\right),$$

où $s_I(.)$ et $u_J(.)$ sont les densités contre la mesure de comptage sur $\{1, ..., r\}$.

Le risque de l'estimateur $\hat{s}_m$ de s est l'espérance de la fonction de perte l de l'estimateur $\hat{s}_m$:

$$E_s\left[l\left(s, \hat{s}_m\right)\right]. \tag{6.2.13}$$

Supposons que la vraie s appartienne au modèle $\mathcal{S}_m$ $(m = m_0)$. En reprenant la démarche suivie et les résultats obtenus par Castellan [18] dans le cadre de l'estimation de densité par histogrammes, nous obtenons le résultat suivant :

Supposons qu'il existe une constante positive ρ telle que $s_I(i) \geq \rho$ pour tout $I \in m$ et $i \in \{1, \ldots, r\}$, et une constante positive Γ telle que $\inf_{I\in m}|I| \geq \Gamma \log^2(n)$ alors le risque de l'estimateur $\hat{s}_m$ est minoré par :

$$E_s\left[l\left(s, \hat{s}_m\right)\right] \geq C(r-1)|m| + \frac{C(\rho, r, \Gamma)}{n}.$$

La minoration est de l'ordre de $D_m = (r-1)|m|$ qui est la dimension du modèle $\mathcal{S}_m$, ce que nous pouvions espérer.

L'étape suivante consiste à choisir une collection de partitions, construire la collection d'estimateurs $\hat{s}_m$ associé à chaque partition, et choisir le meilleur estimateur de cette collection. Comme nous l'avons vu dans le chapitre 1, puisque le risque de l'estimateur $\hat{s}_m$ dépend de la fonction s qui est inconnue, l'estimateur qui réalise le plus petit risque des estimateurs de la collection dépendra de s et ne pourra être utilisé comme estimateur de s. Nous sélectionnons alors le meilleur estimateur de la collection à l'aide d'un critère construit uniquement à partir des données.

6.2.3 Estimateur du maximum de vraisemblance pénalisé

Soit $\mathcal{F}_n$ une collection finie de partitions de $\{1, .., n\}$. En estimant s sur tous les modèles $\{S_m, m \in \mathcal{F}_n\}$, nous obtenons une collection d'estimateurs $\{\hat{s}_m, m \in \mathcal{F}_n\}$ où $\hat{s}_m$ est défini pour toute partition $m \in \mathcal{F}_n$ par (6.2.9). Choisir ensuite le meilleur estimateur revient à choisir la meilleure partition. Ce choix se fait par la minimisation d'un critère pénalisé :

Etant donnée une fonction $pen : \mathcal{F}_n \to \mathbb{R}^+$, appelée fonction de pénalité, le critère pénalisé est défini par :

$$crit(m) = \gamma'(\hat{s}_m) + pen(m). \tag{6.2.14}$$

La minimisation de ce critère sur la famille de partitions $\mathcal{F}_n$ conduit au choix d'une partition :

$$\hat{m} = \underset{m \in \mathcal{F}_n}{\operatorname{argmin}}\, crit(m). \tag{6.2.15}$$

Et l'estimateur du maximum de vraisemblance pénalisé est alors défini par :

$$\tilde{s} = \hat{s}_{\hat{m}}. \tag{6.2.16}$$

C'est l'estimateur du maximum de vraisemblance calculé dans le modèle $S_{\hat{m}}$.

Il nous faut maintenant établir l'expression de la fonction de pénalité. L'objectif de la sélection de modèle est de trouver cette fonction de pénalité de sorte que le risque de l'estimateur pénalisé $\tilde{s}$, $E_s[l(s, \tilde{s})]$, soit aussi proche que possible du plus petit des risques des estimateurs de la collection $\{\hat{s}_m, m \in \mathcal{F}_n\}$.

6.2.3.1 Résultat principal

Nous énonçons le théorème principal qui donne la forme de la pénalité et le contrôle du risque de l'estimateur pénalisé $\tilde{s}$ correspondant. L'utilisation concrète d'un tel résultat est sensible à la forme de la pénalité utilisée. De ce point de vue, le facteur $\log \frac{n}{|m|}$ apparaîssant dans la pénalité donnée en (6.2.19) peut paraître inhabituel. Il n'intervient pas par exemple dans les critères classiques tels que le BIC (Bayesien Information criterion) ou le AIC (A Information criterion) introduits respectivement par Schwarz [59] et Akaike [1] dans le cadre de l'estimation par maximum de vraisemblance. Ce terme provient comme dans le chapitre 1 de la combinatoire de la collection de modèles de même dimension.

Théorème 6.2.1. *Soient $Y_1, \ldots, Y_n$ n variables aléatoires discrètes indépendantes prenant leurs valeurs dans l'ensemble fini d'entiers $\mathcal{Y} = \{1, \ldots, r\}$ pour $r \geq 2$, dont la loi du n-uplet $Y = (Y_1, \ldots, Y_n)$, notée P_s, est associé à s, défini par (6.2.2), par l'expression (6.2.1).*

Nous prenons $\mathcal{F}_n$ comme étant l'ensemble des partitions de $\{1, .., n\}$ construites sur une partition "dite la plus fine" et notée m_f telle que la taille Γ_{m_f} du plus petit segment de m_f

vérifie :

$$\Gamma_{m_f} \geq \Gamma \log^2(n) \tag{6.2.17}$$

pour Γ une constante strictement positive.

Supposons qu'il existe une constante strictement positive ρ telle que pour tout $I \in m_0$ et tout $i \in \{1, \ldots, r\}$

$$s_I(i) \geq \rho. \tag{6.2.18}$$

Etant donnée $C > 1$, il existe des constantes K_1 et K_2 dépendantes de ρ et C telles que si nous choisissons la pénalité définie pour tout $m \in \mathcal{F}_n$ par :

$$pen(m) = r|m| \left[K_1 \log\left(\frac{n}{|m|}\right) + K_2 \right], \tag{6.2.19}$$

alors nous disposons de la majoration suivante du risque de l'estimateur du maximum de vraisemblance pénalisé $\tilde{s}$, défini par (6.2.16),

$$E_s[\, l(s, \tilde{s})] \leqslant \inf_{m \in \mathcal{F}_n} \left\{ C \inf_{u \in \mathcal{S}_m} l(s, u) + (C+1) pen(m) \right\} + C'(r, \rho, C). \tag{6.2.20}$$

6.2.3.2 Interprétation

D'après la majoration du risque donnée en (6.2.20), nous avons que

$$E_s[l(s, \tilde{s})] \leqslant C \log n \inf_{m \in \mathcal{F}_n} \left\{ \inf_{u \in \mathcal{S}_m} l(s, u) + r|m| \right\},$$

pour une constante C positive.

Pour comprendre si la majoration est fine, intéressons nous au cas du modèle $\mathcal{S}_{m_0}$, modèle auquel appartient s. Puisque $\inf_{u \in \mathcal{S}_{m_0}} l(s, u) = 0$, nous obtenons

$$E_s[l(s, \tilde{s})] \leqslant C \log n r |m_0|.$$

Pour le modèle $\mathcal{S}_{m_0}$, nous avons obtenu dans la sous-section précédente une minoration du risque de l'estimateur $\hat{s}_{m_0}$. Nous avons

$$E_s[l(s, \hat{s}_{m_0})] \geq C' r |m_0|.$$

Donc l'estimateur pénalisé $\tilde{s}$ fait aussi bien à un $\log n$ près que $\tilde{s}_{m_0}$, comme si on connaissait s.

6.2.4 Preuve du théorème 6.2.1

Cette preuve s'appuie sur un théorème de sélection et sa démonstration établi par Massart [48] pour les fonctions de contraste borné. Nous rappelons tout d'abord ce théorème.

Théorème 6.2.2. *Soient $X_1, \ldots, X_n$ n variables aléatoires indépendantes prenant leurs valeurs dans un espace mesurable Ξ et munis d'une distribution commune P dependant d'un paramètre $s \in \mathcal{S}$. Soit $\gamma : \mathcal{S} \times \Xi$ satisfaisant les hypothèses* **H1** *et* **H2**. *Soit $(\mathcal{S}_m)_{m \in \mathcal{M}_n}$ une famille finie de sous-espaces de $\mathcal{S}$. Fixons $(x_m)_{m \in \mathcal{M}_n}$ une famille de poids satisfaisant la condition :*

$$\sum_{m \in \mathcal{M}_n} e^{-x_m} \leqslant \Sigma,$$

où Σ est une constante déterminée.

De plus, considérons les hypothèses suivantes :

H1: *Le contraste est borné par une constante b.*

H2: *Il existe une pseudo-distance d et une constante absolue c satisfaisant pour tout u et u' dans $\mathcal{S}$ telle que*

$$var_s \left[\bar{\gamma}(u) - \bar{\gamma}(u')\right] \leqslant d^2(u, u'),$$

et

$$d^2(s, u) \leqslant cl(s, u).$$

H3: *Pour tout nombre positif σ et tout $u \in \mathcal{S}$, définissons*

$$B_m(u, \sigma) = \left\{u' \in S_m \ ; \ d(u', u) \leqslant \sigma\right\},$$

où d est la pseudo-distance donné par **H2**. *Supposons que pour tout $m \in \mathcal{M}_n$, il existe une fonction continue ϕ_m de $\mathbb{R}_+$ de $\mathbb{R}_+$ telle que $\phi_m(0) = 0$, $\phi_m(x)/x$ est décroissante et*

$$\mathbb{E}\left[\sup_{u' \in B_m(u,\sigma)} |\bar{\gamma}(u') - \bar{\gamma}(u)|\right] \leqslant \phi_m(\sigma),$$

pour tout $\sigma \geqslant \sigma_m$ où σ_m est tel que $\phi_m(\sigma_m) = \sigma_m^2$.

Etant données **H1**, **H2** *et* **H3**, *$C > 1$ et $\xi > 0$, il existe des constantes positives B_1 et B_2 (dépendantes de C et des constantes b et c des hypothèses* **H1** *et* **H2**) *telles que si pour tout $m \in \mathcal{M}_n$,*

$$\mathrm{pen}_n(m) \geqslant B_1 \sigma_m^2 + B_2 x_m,$$

alors sur un ensemble Ω_ξ de probabilité plus grande que $1 - \Sigma e^{-\xi}$, le risque de l'estimateur $\tilde{s}$ est borné par :

$$E_s\left[l(s, \tilde{s})\right] \leqslant C\left[\inf_{m \in \mathcal{M}_n}\left\{\inf_{u \in S_m} l(s, u) + pen(m)\right\} + C'(1 + \Sigma)\right],$$

où $l(s, s_m) = \inf_{u \in S_m} l(s, u)$ et C' est une constante (dépendante de C, b et c).

Pour pouvoir utiliser les techniques employées dans ce théorème et sa démonstration et obtenir la borne de risque de l'estimateur pénalisé, nous définissons les deux notions suivantes :

Pour toute partition $m \in \mathcal{F}_n$, nous nous restreignons au modèle associé à la partition m et défini d'une façon générale pour $\eta \in]0,1[$ par :

$$\mathcal{S}_m(\eta) = \left\{ \begin{array}{l} u = \sum_{J \in m} \sum_{i=1}^{r} u_J(i) \ \mathbb{1}_{J \times i} \\ \text{avec } u_J(i) \geq \eta, \forall J \in m, \ \forall i \in \{1, \ldots, r\} \\ \text{et } \int_{\{1,..,n\} \times \{1,..,r\}} u(x) \ d\mu_n(x) = 1 \end{array} \right\},$$

La raison pour laquelle nous considérons ce modèle est que pour tout $u \in \mathcal{S}_m(\eta)$, les lois marginales de P_u sur la variable t, soit $u(t,.)$ seront minorées par η. Cette hypothèse sera nécessaire pour établir par la suite une inégalité de concentration. Notons que d'après l'hypothèse (6.2.18), la vraie $s \in \mathcal{S}_m(\rho)$. Nous posons $\eta = \frac{\rho}{2}$ et nous considérerons dans toute la suite, les modèles $\mathcal{S}_m\left(\frac{\rho}{2}\right)$, pour $m \in \mathcal{F}_n$. Ainsi, pour une partition m de $\mathcal{F}_n$ fixée et pour tout $u \in \mathcal{S}_m\left(\frac{\rho}{2}\right)$, nous aurons que pour tout $J \in m$ et pour tout $i \in \{1, \ldots, r\}$:

$$u_J(i) \geq \frac{\rho}{2}.$$

Nous considérons alors l'évènement :

$$\Omega_{m_f}(\rho) = \left\{ \forall i \in \{1,..,r\}, \forall J \subset m_f \ ; \ \frac{N_J(i)}{|J|} \geq \frac{\rho}{2} \right\}, \tag{6.2.21}$$

où $J \subset m_f$ signifie que J est une réunion de segments consécutifs de m_f. La restriction à cet évènement $\Omega_{m_f}(\rho)$ permet d'avoir, étant donnée une partition $m \in \mathcal{F}_n$, une forme explicite de l'estimateur du maximum de vraisemblance $\hat{s}_m$ défini par (6.2.9). En effet, toute partition $m \in \mathcal{F}_n$ est construite sur la partition la plus fine m_f. Les segments de la partition m sont donc des segments et des réunions de segments consécutifs de m_f. Se placer sur l'espace $\Omega_{m_f}(\rho)$ nous assure donc que pour tout $J \in m$, $N_J(i)/|J| \geq \frac{\rho}{2}$, $\forall i \in \{1, \ldots, r\}$.

Commençons la preuve du théorème 6.2.1. Nous nous plaçons sur l'évènement $\Omega_{m_f}(\rho)$ et nous cherchons à calculer le risque de l'estimateur $\tilde{s}$.

Pour une partition $m \in \mathcal{F}_n$, nous introduisons la projection de s sur le modèle $\mathcal{S}_m\left(\frac{\rho}{2}\right)$ au sens de la fonction de perte défini pour tout $u \in \mathcal{S}_m\left(\frac{\rho}{2}\right)$ par (6.2.12). Cette projection est

noté $\bar{s}_m$ et est définie par :

$$\begin{aligned} \bar{s}_m &= \underset{u \in \mathcal{S}_m\left(\frac{\rho}{2}\right)}{\operatorname{argmin}} \; l\left(s, u\right) \\ &= \sum_{J \in m} \sum_{i=1}^{r} \left(\frac{\sum_{I \in m_0} |J \cap I| s_I\left(i\right)}{|\, J \,|} \right) \mathbb{1}_{J \times i} \\ &= \sum_{J \in m} \sum_{i=1}^{r} \bar{s}_J(i) \; \mathbb{1}_{J \times i}. \end{aligned} \tag{6.2.22}$$

D'après la définition de la partition $\hat{m}$ donnée en (6.2.15), du critère pénalisé donné en (6.2.14), et l'expression de l'estimateur $\hat{s}_m$ donnée en (6.2.9), nous obtenons la suite d'inégalités suivante :

$$\forall m \in \mathcal{F}_n \quad \gamma'\left(\tilde{s}\right) + pen\left(\hat{m}\right) \leqslant \gamma'\left(\hat{s}_m\right) + pen(m) \leqslant \gamma'\left(\bar{s}_m\right) + pen(m). \tag{6.2.23}$$

De plus, d'après la définition de la fonction de perte donnée en (6.2.10), nous avons que :

$$\begin{aligned} l(s, \tilde{s}) &= E_s\left[\gamma'(\tilde{s}) - \gamma'(s)\right] \\ &= l(s, \bar{s}_m) + E_s\left[\gamma'(\tilde{s}) - \gamma'(\bar{s}_m)\right]. \end{aligned}$$

En notant $\bar{\gamma}'$ le contraste empirique recentré, défini pour tout $u \in \mathcal{S}_m\left(\frac{\rho}{2}\right)$ par :

$$\bar{\gamma}'(u) = \gamma'(u) - E_s\left[\gamma'(u)\right],$$

et en en remaniant les inégalités (6.2.23), nous obtenons la majoration de la perte de l'estimateur pénalisé suivante :

$$l(s, \tilde{s}) \leqslant l(s, \bar{s}_m) + \bar{\gamma}'(s_m) - \bar{\gamma}'(\tilde{s}) + pen(m) - pen(\hat{m}). \tag{6.2.24}$$

Nous aimerions choisir une fonction de pénalité qui compense le terme $\bar{\gamma}'(\bar{s}_m) - \bar{\gamma}'(\tilde{s}) = \bar{\gamma}'(\bar{s}_m) - \bar{\gamma}'(\hat{s}_{\hat{m}})$. Le point délicat tient dans le contrôle du terme $\bar{\gamma}'(\bar{s}_m) - \bar{\gamma}'(\hat{s}_{\hat{m}})$ puisque nous ne savons pas quelle est la partition $\hat{m}$. L'idée alors est d'obtenir un contrôle uniforme du terme $\bar{\gamma}'(\bar{s}_m) - \bar{\gamma}'(\hat{s}_{m'})$ pour toutes les partitions $m' \in \mathcal{F}_n$, pour qu'il puisse être appliqué pour $m' = \hat{m}$. Pour obtenir un tel contrôle, nous reprenons la démarche suivie par Massart pour la démonstration du théorème 6.2.2 donnée dans [48].

6.2.4.1 Pseudo-distance associée à la variance du contraste

La première étape consiste à trouver la renormalisation qui stabilise la quantité $\gamma'\left(u\right) - \gamma'\left(u'\right)$ pour tout $u \in S_m\left(\frac{\rho}{2}\right)$ et tout $u' \in S_{m'}\left(\frac{\rho}{2}\right)$, et pour toute partition $m, m' \in \mathcal{F}_n$. Nous définissons une pseudo-distance notée d définie sur l'espace $S(\frac{\rho}{2}) = \bigcup_{m \in \mathcal{F}_n} \mathcal{S}_m(\frac{\rho}{2})$ et qui est associé à la variance de la quantité considérée, appelée variance du contraste.

Nous donnons tout d'abord une majoration de la variance du contraste.

Proposition 6.2.3. *Soit $u \in S_m\left(\frac{\rho}{2}\right)$ et $u' \in S_{m'}\left(\frac{\rho}{2}\right)$, la variance du contraste est majorée par :*

$$var_s\left[\gamma'(u) - \gamma'(u')\right] \leqslant (2r+1) \sum_{J \in m} \sum_{J' \in m'} |J \cap J'| \sum_{i=1}^{r} \log^2\left(\frac{u_J(i)}{u'_{J'}(i)}\right).$$

Preuve.

Nous donnons tout d'abord le lemme suivant :

Lemme 6.2.4. *Soit $(Z_t)_{1 \leqslant t \leqslant N}$ une suite de variables aléatoires, nous avons l'inégalité suivante :*

$$var\left(\sum_{t=1}^{N} Z_t\right) \leqslant (2N+1) \sum_{t=1}^{N} var\left(Z_t\right).$$

Rappelons que pour $u \in \mathcal{S}_{m'}(\frac{\rho}{2})$ et $u' \in \mathcal{S}_m(\frac{\rho}{2})$,

$$\begin{aligned} \gamma'(u) - \gamma'(u') &= \sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} N_{J \cap J'}(i) \log\left(\frac{u'_{J'}(i)}{u_J(i)}\right) \\ &= \sum_{I \in m_0} \sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} N_{I \cap J \cap J'}(i) \log\left(\frac{u'_{J'}(i)}{u_J(i)}\right). \end{aligned} \tag{6.2.25}$$

En appliquant le lemme 6.2.4, nous obtenons

$$var_s\left[\gamma'(u) - \gamma'(u')\right] \leqslant (2r+1) \sum_{I \in m_0} \sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} var_s\left[N_{I \cap J \cap J'}(i)\right] \log^2\left(\frac{u_J(i)}{u'_{J'}(i)}\right).$$

Nous calculons la variance du comptage $N_{I \cap J \cap J'}(i)$, donné pour un $J \in m$ en (6.2.6) :

$$\begin{aligned} var_s\left[N_{I \cap J \cap J'}(i)\right] &= var_s\left[\sum_{t \in I \cap J \cap J'} \mathbb{1}_{Y_t = i}\right] \\ &= |I \cap J \cap J'| s_I(i)\left[1 - s_I(i)\right]. \end{aligned}$$

Nous concluons en majorant la variance du comptage par :

$$var_s\left[N_{I \cap J \cap J'}(i)\right] \leqslant |I \cap J \cap J'|. \tag{6.2.26}$$

Cette majoration permet de définir une pseudo-distance associée à la variance du contraste.

Definition 6.2.5. *Nous définissons une pseudo-distance d sur* $S(\frac{\rho}{2}) = \bigcup_{m \in \mathcal{F}_n} \mathcal{S}_m(\frac{\rho}{2})$, *qui vaut pour tout* $u \in S_m\left(\frac{\rho}{2}\right)$ *et* $u' \in S_{m'}\left(\frac{\rho}{2}\right)$ *:*

$$d\left(u, u'\right) = \sum_{J \in m} \sum_{J' \in m'} |J \cap J'| \sum_{i=1}^{r} \log^2 \left(\frac{u_J\left(i\right)}{u'_{J'}\left(i\right)} \right). \tag{6.2.27}$$

Elle vérifie l'inégalité suivante :

$$var_s \left[\gamma'\left(u\right) - \gamma'\left(u'\right)\right] \leqslant \left(2r+1\right) d^2\left(u, u'\right).$$

Il s'agit maintenant de relier la pseudo-distance d à la fonction de perte l.

Proposition 6.2.6. *Pour* $s \in \mathcal{S}_{m_0}(\rho)$ *et pour tout* $u \in \mathcal{S}_m(\frac{\rho}{2})$, *nous obtenons la relation suivante entre la pseudo-distance d et la fonction de perte l :*

$$d^2\left(s, u\right) \leqslant \frac{4}{\rho^2} \, l\left(s, u\right).$$

avec

$$d^2\left(s, u\right) = \sum_{I \in m_0} \sum_{J \in m} |I \cap J| \sum_{i=1}^{r} \log^2 \left(\frac{s_I\left(i\right)}{u_J\left(i\right)} \right),$$

et

$$l\left(s, u\right) = \sum_{I \in m_0} \sum_{J \in m} |I \cap J| \sum_{i=1}^{r} s_I\left(i\right) \log \left(\frac{s_I\left(i\right)}{u_J\left(i\right)} \right).$$

Preuve.

Comme nous l'avons vu dans la sous-section 6.2.2, la perte s'écrit comme une somme de Kullback. Nous redonnons son expression :

$$l\left(s, u\right) = \sum_{I \in m_0} \sum_{J \in m} |I \cap J| \; K\left(s_I\left(.\right), t_J\left(.\right)\right).$$

Nous utilisons un lemme sur l'information de Kullback énoncé par Castellan dans [18] que nous rappelons.

Lemme 6.2.7. *Soient p et q deux densités de probabilités sur* χ *contre la mesure* μ, *en notant* $f = \log\left(\frac{q}{p}\right)$, *nous obtenons :*

$$\frac{1}{2} \int_{\chi} f^2 \left(1 \wedge e^f\right) \, p d\mu \leqslant K\left(p, q\right) \leqslant \frac{1}{2} \int_{\chi} f^2 \left(1 \vee e^f\right) p \, d\mu.$$

Nous appliquons ce résultat aux densités $s_I(.)$ et $u_I(.)$ en considérant l'hypothèse sur s donnée en (6.2.18) et pour tout $u \in \mathcal{S}_m(\frac{\rho}{2})$. Nous obtenons la majoration suivante :

$$d^2(s,u) \leqslant \frac{2}{\rho^2} \sum_{I \in m_0} \sum_{J \in m} \sum_{i=1}^{r} |I \cap J| \; K\left(s_I(.) \; , \; u_J(.) \right).$$

Ce qui conclut la démonstrastion de la proposition.

Pour tout $u' \in S_{m'}\left(\frac{\rho}{2}\right)$, nous définissons les deux notions suivantes :

$$w_{m'}(u') = \left[d(s,\bar{s}_m) + d(s,u')\right]^2 + y_{m'}^2, \tag{6.2.28}$$

où $y_{m'}$ sera choisi plus tard, et

$$V_{m'} = \sup_{u' \in S_{m'}\left(\frac{\rho}{2}\right)} \left\{ \frac{|\bar{\gamma}'(u') - \bar{\gamma}'(\bar{s}_m)|}{w_{m'}(u')} \right\}.$$

Reprenons l'inégalité de départ concernant la perte de l'estimateur pénalisé $\tilde{s}$ donnée en (6.2.24). Nous pouvons alors écrire l'inégalité suivante :

$$l(s,\hat{s}_{\hat{m}}) \leqslant l(s,\bar{s}_m) + w_{\hat{m}}(\tilde{s})\, V_{\hat{m}} + pen(m) - pen(\hat{m}). \tag{6.2.29}$$

6.2.4.2 Inégalité de concentration

Dans une seconde étape, nous cherchons à voir si la quantité $V_{m'}$ est proche de son espérance pour toute partition $m' \in \mathcal{F}_n$. Nous appliquons une version de Massart [48] de l'inégalité de Talagrand qui permet d'obtenir une inégalité exponentielle de concentration conduisant à un contrôle de $V_{m'}$ autour de son espérance. Nous donnons tout d'abord le résultat que nous avons obtenu, puis la preuve dans laquelle nous rappelons l'inégalité de Talagrand revisité.

Proposition 6.2.8. *Soit $m' \in \mathcal{F}_n$ et $V_{m'}$ défini par :*

$$V_{m'} = \sup_{u' \in S_{m'}\left(\frac{\rho}{2}\right)} \left\{ \frac{|\bar{\gamma}'(u') - \bar{\gamma}'(\bar{s}_m)|}{w_{m'}(u')} \right\}. \tag{6.2.30}$$

Alors

$$P\left(V_{m'} \geq k_1 \, E_s[V_{m'}] + k_2 \, \left[\sqrt{v\,x}\; y_{m'}^{-1} + b\, x\, y_{m'}^{-2}\right]\right) \leqslant \exp(-x), \tag{6.2.31}$$

pour k_1 et k_2 des constantes positives et pour tout $x > 0$, et avec

$$b = r \log\left(\frac{2}{\rho}\right), \tag{6.2.32}$$

et

$$v = \frac{3r}{4}. \tag{6.2.33}$$

Preuve.

Nous rappelons tout d'abord l'inégalité donnée dans [48] que nous allons utiliser.

Théorème 6.2.9. *Soient $X_1, \ldots, X_n$, n variables aléatoires indépendantes et identiquement distribuées à valeurs dans l'espace Ξ. Soit $\mathcal{F}$ une famille finie de fonctions mesurables sur Ξ telle que $\|f\|_\infty \leqslant b' < \infty$ pour tout $f \in \mathcal{F}$. Soient*

$$Z = \sup_{f \in \mathcal{F}} \left| \sum_{t=1}^{n} f(X_t) \right|,$$

et

$$v' = \sup_{f \in \mathcal{F}} E\left[\sum_{t=1}^{n} f^2(X_t) \right].$$

Alors

$$P\left(Z \geq k_1 E(Z) + k_2\left(\sqrt{v'x} + b'x\right)\right) \leqslant \exp -x \quad \textit{pour tout } x > 0, \tag{6.2.34}$$

où k_1 et k_2 sont des constantes positives.

Pour appliquer ce théorème, nous remanions $V_{m'}$.

Nous définissons pour tout $J \in m$, $J' \in m'$ et $i \in \{1, \ldots, r\}$

$$\bar{N}_{J \cap J'}(i) = \sum_{t \in J \cap J'} \mathbb{1}_{\{Y_t = i\}} - \sum_{I \in m_0} |I \cap J \cap J'| s_I(i) = \sum_{I \in m_0} \sum_{t \in J \cap J' \cap I} \bar{\mathbb{1}}_{\{Y_t = i\}}, \tag{6.2.35}$$

le comptage $N_{J \cap J'}(i)$ recentré. D'après l'expression de $\bar{s}_m$ donnée en (6.2.22) et l'égalité (6.2.25), nous avons pour tout $u' \in S_{m'}\left(\frac{\rho}{2}\right)$ l'égalité suivante :

$$\bar{\gamma}'(u') - \bar{\gamma}'(\bar{s}_m) = \sum_{I \in m_0} \sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \bar{N}_{J' \cap J \cap I}(i) \log\left(\frac{\bar{s}_J(i)}{u'_{J'}(i)}\right). \tag{6.2.36}$$

En définissant pour tout $I \in m_0$, $J \in m$,$J' \in m'$ et $t \in J' \cap J \cap I$,

$$\bar{Z}_{I,J,J',t} = \sum_{i=1}^{r} \bar{\mathbb{1}}_{\{Y_t = i\}} \log\left(\frac{\bar{s}_J(i)}{u'_{J'}(i)}\right),$$

nous pouvons réécrire $V_{m'}$ sous la forme suivante :

$$V_{m'} = \sup_{u' \in S_{m'}\left(\frac{\rho}{2}\right)} \left\{ \frac{\left| \sum_{I \in m_0} \sum_{J \in m} \sum_{J' \in m'} \sum_{t \in J' \cap J \cap I} \bar{Z}_{I,J,J',t} \right|}{w_{m'}(u')} \right\}.$$

Pour obtenir les constantes b' et v' du théorème 6.2.9, nous cherchons à majorer respectivement $\left|\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right|$ et $E_s\left[\sum_{I\in m_0}\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I}\left(\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right)^2\right]$:

- Puisque $u' \in S_{m'}\left(\frac{\rho}{2}\right)$, et d'après l'expression de $w_{m'}(u')$ donnée en (6.2.28), nous obtenons la majoration suivante :

$$\left|\frac{\bar{Z}_{J,K,t}}{w_{m'}(u')}\right| \leqslant \frac{r\log\left(\frac{2}{\rho}\right)}{y_{m'}^2}.$$

- En reprenant la même démarche que dans la démonstration de la majoration de la variance du contraste donnée dans la proposition 6.2.3 et d'après l'expression de d donnée en (6.2.27), nous avons :

$$\begin{aligned}
&E_s\left[\sum_{I\in m_0}\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I}\left(\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right)^2\right] \\
&\qquad = \sum_{I\in m_0}\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I} var_s\left[\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right] \\
&\qquad \leqslant \frac{(2r+1)\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I}\sum_{i=1}^{r}\log^2\left(\frac{\bar{s}_J(i)}{u'_{J'}(i)}\right)}{w_{m'}^2(u')} \\
&\qquad \leqslant \frac{(2r+1)d^2(u',\bar{s}_m)}{w_{m'}^2(u')}.
\end{aligned}$$

D'après l'expression de $w_{m'}(u')$ donnée en (6.2.28), nous obtenons la majoration suivante :

$$E_s\left[\sum_{I\in m_0}\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I}\left(\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right)^2\right] \leqslant \frac{(2r+1)d^2(u',\bar{s}_m)}{\left[d^2(u',\bar{s}_m)+y_{m'}^2\right]^2}.$$

En utilisant l'inégalité suivante :

$$2\alpha\beta \leqslant \alpha^2+\beta^2 \quad \forall\alpha,\beta,$$

nous obtenons

$$E_s\left[\sum_{I\in m_0}\sum_{J\in m}\sum_{J'\in m'}\sum_{t\in J'\cap J\cap I}\left(\frac{\bar{Z}_{I,J,J',t}}{w_{m'}(u')}\right)^2\right] \leqslant \frac{(2r+1)}{4y_{m'}^2}.$$

Nous prenons

$$b' = \frac{b}{y_{m'}^2} \quad , \quad v' = \frac{v}{y_{m'}^2},$$

avec b et v les constantes données respectivement en (6.2.32) et (6.2.33), et nous obtenons par (6.2.34) l'inégalité (6.2.31).

6.2.4.3 Calcul de $E(V_{m'})$

La troisième étape consiste à calculer un majorant de $E_s[V_{m'}]$.

D'après les expressions de $V_{m'}$ et du contraste données respectivement en (6.2.30) et (6.2.36), nous écrivons $V_{m'}$ sous la forme suivante :

$$V_{m'} = \sup_{u' \in \mathcal{S}_{m'}(\frac{\rho}{2})} \left| \sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{\bar{N}_{J \cap J'}(i)}{\sqrt{|J \cap J'|}} \frac{\sqrt{|J \cap J'|}}{w_{m'}(u')} \log\left(\frac{\bar{s}_J(i)}{u'_{J'}(i)}\right) \right|.$$

En utilisant l'inégalité de Cauchy-Schwarz, nous obtenons la majoration de l'espérance de $V_{m'}$ suivante :

$$\begin{aligned} & E_s[V_{m'}] \\ \leqslant\ & E_s\left[\sqrt{\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{\bar{N}_{J \cap J'}(i)^2}{|J \cap J'|}} \sup_{u' \in \mathcal{S}_{m'}(\frac{\rho}{2})} \sqrt{\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{|J \cap J'|}{w_{m'}^2(u')} \log^2\left(\frac{\bar{s}_J(i)}{u'_{J'}(i)}\right)} \right]. \end{aligned}$$

D'après les expressions de d et de $w_{m'}$ données respectivement en (6.2.27) et (6.2.28), et comme précédemment, nous avons :

$$\begin{aligned} & E_s[V_{m'}] \\ & \leqslant E_s\left[\sqrt{\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{\bar{N}_{J \cap J'}(i)^2}{|J \cap J'|}} \right] \sup_{u' \in \mathcal{S}_{m'}(\frac{\rho}{2})} \sqrt{\frac{d^2(u', s_m)}{\left[d^2(u', s_m) + y_{m'}^2\right]^2}} \\ & \leqslant \frac{1}{2y_{m'}} E_s\left[\sqrt{\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{\bar{N}_{J \cap J'}(i)^2}{|J \cap J'|}} \right]. \end{aligned}$$

Par l'inégalité de Jensen, nous obtenons

$$\begin{aligned} & E_s[V_{m'}] \\ & \leqslant \frac{1}{2y_{m'}} \sqrt{E_s\left[\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{\bar{N}_{J \cap J'}(i)^2}{|J \cap J'|} \right]} \\ & \leqslant \frac{1}{2y_{m'}} \sqrt{\sum_{J \in m} \sum_{J' \in m'} \sum_{i=1}^{r} \frac{var_s[N_{J \cap J'}(i)]}{|J \cap J'|}}. \end{aligned}$$

Or

$$var_s[N_{J \cap J'}(i)] = \sum_{I \in m_0} var_s[N_{I \cap J \cap J'}(i)],$$

et nous obtenons la majoration de $E_s[V_{m'}]$ suivante :

$$E_s[V_{m'}] \leqslant \frac{\sqrt{r}}{2y_{m'}} (\sqrt{|m|} + \sqrt{|m'|}). \tag{6.2.37}$$

6.2.4.4 Majoration du risque

Pour la dernière étape, nous cherchons à contrôler le risque de l'estimateur pénalisé.

Etant donné ξ, nous posons $x = x_{m'} + \xi$ où d'après le théorème 6.2.2, $(x_m)_{m \in \mathcal{M}_n}$ est une famille de poids telle que :

$$\sum_{m \in \mathcal{F}_n} e^{-x_m} \leqslant \Sigma < +\infty.$$

Nous définissons l'évènement suivant :

$$\Omega_\xi = \left\{ \forall m' \in \mathcal{F}_n, V_{m'} \leqslant k_1 \frac{\sqrt{r}}{2} y_{m'}^{-1} \left(\sqrt{|m|} + \sqrt{|m'|} \right) + k_2 \left(\sqrt{v(x_{m'} + \xi)} y_{m'}^{-1} + b(x_{m'} + \xi) y_{m'}^{-2} \right) \right\},$$

où b et v sont donnés respectivement en (6.2.32) et (6.2.33). D'après les inégalités (6.2.31) et (6.2.37), cet évènement est de probabilité supérieure à $1 - \Sigma e^{-\xi}$:

$$P(\Omega_\xi) = 1 - P\left(\Omega_\xi^c\right) \geq 1 - \sum_{m' \in \mathcal{F}_n} e^{-x_{m'} - \xi} \geq 1 - \Sigma e^{-\xi}.$$

Sur Ω_ξ, pour tout $m' \in \mathcal{F}_n$, $V_{m'}$ est majoré par :

$$V_{m'} \leqslant k_1 \frac{\sqrt{r}}{2} y_{m'}^{-1} \left(\sqrt{|m|} + \sqrt{|m'|} \right) + k_2 \left(\sqrt{v(x_{m'} + \xi)} y_{m'}^{-1} + b(x_{m'} + \xi) y_{m'}^{-2} \right).$$

Posons

$$y_{m'} = 2K \left[k_1 \frac{\sqrt{r}}{2} \left(\sqrt{|m|} + \sqrt{|m'|} \right) + k_2 \sqrt{v(x_{m'} + \xi)} + \sqrt{k_2 b(x_{m'} + \xi)} \right],$$

alors sur Ω_ξ,

$$V_{m'} \leqslant K^{-1} \quad \forall m' \in \mathcal{F}_n,$$

et l'inégalité (6.2.29) devient

$$l(s, \tilde{s}) \leqslant l(s, \bar{s}_m) + K^{-1} w_{\hat{m}}(\tilde{s}) - pen(\hat{m}) + pen(m).$$

En remplaçant $w_{\hat{m}}(\tilde{s})$ par son expression donnée en (6.2.28), nous obtenons

$$l(s, \tilde{s}) \leqslant l(s, \bar{s}_m) + K^{-1} \left\{ [d(s, \bar{s}_m) + d(s, \tilde{s})]^2 + y_{\hat{m}}^2 \right\} - pen(\hat{m}) + pen(m).$$

En utilisant l'inégalité suivante

$$\forall \alpha, \beta > 0 \quad (\alpha + \beta)^2 \leqslant 2(\alpha^2 + \beta^2), \tag{6.2.38}$$

et la relation entre d et la perte l donnée dans la proposition 6.2.6, nous obtenons

$$[d(s, \bar{s}_m) + d(s, \tilde{s})]^2 \leqslant 2 \left[d^2(s, \bar{s}_m) + d^2(s, \tilde{s}) \right] \leqslant 2c(\rho) \left[l(s, \bar{s}_m) + l(s, \tilde{s}) \right],$$

où $c(\rho) = \frac{4}{\rho^2}$,

D'autre part, en utilisant de nouveau deux fois l'inégalité (6.2.38), nous obtenons la majoration de $y_{\hat{m}}^2$ suivante :

$$y_{\hat{m}}^2 \leqslant 8K^2 \left[k_1^2 \frac{r}{2} |m| + k_1^2 \frac{r}{2} |\hat{m}| + k_2 (x_{\hat{m}} + \xi)(\sqrt{k_2 v} + \sqrt{b})^2 \right].$$

Donc sur Ω_ξ, la perte de l'estimateur pénalisé est majorée par :

$$\begin{aligned} (6.2.39) \quad l(s, \tilde{s}) &\leqslant l(s, \bar{s}_m) \left(1 + \frac{2c(\rho)}{K}\right) \\ &+ \frac{2c(\rho)}{K} l(s, \tilde{s}) + 8K \left[k_1^2 \frac{r}{2} |\hat{m}| + k_2 (x_{\hat{m}} + \xi)(\sqrt{k_2 v} + \sqrt{b})^2 \right] \\ &- pen(\hat{m}) + pen(m) + 4K k_1^2 r |m|. \end{aligned}$$

Nous cherchons à définir les poids x_m. Nous considèrons que toutes les partitions possibles au pire sont visitées. Comme dans le cadre de la détection de ruptures dans la moyenne étudié au chapitre 1 (section 2.4), en prenant x_m comme une fonction de la dimension, nous avons

$$\begin{aligned} \sum_{m \in \mathcal{F}_n} e^{-x_m} &= \sum_{D=1}^{n} e^{-x_D} \#\{m \in \mathcal{F}_n, |m| = D\} \\ &\leqslant \sum_{D=1}^{n} e^{-x_D} C_n^D \\ &\leqslant \sum_{D=1}^{n} e^{-\left(x_D - D - D \log\left(\frac{n}{D}\right)\right)}. \end{aligned}$$

Il suffit donc de prendre

$$x_m = |m| \left(\log \frac{n}{|m|} + 2 \right) \qquad \forall m \in \mathcal{F}_n,$$

et nous avons $\Sigma < 1$.

En subtituant alors les poids $x_{\hat{m}}$ dans l'inégalité (6.2.39) et d'après les constantes b et v obtenues et données respectivement en (6.2.32) et (6.2.33), nous obtenons

$$\begin{aligned} \left(1 - \frac{2c(\rho)}{K}\right) l(s, \tilde{s}) \leqslant & l(s, \bar{s}_m) \left(1 + \frac{2c(\rho)}{K}\right) \\ & + r|\hat{m}| \left\{ 8Kk_2 \left[\sqrt{k_2 \frac{3}{4}} + \sqrt{\log\left(\frac{2}{\rho}\right)} \right]^2 \log \frac{n}{|\hat{m}|} \right. \\ & \left. + 16Kk_2 \left[\sqrt{k_2 \frac{3}{4}} + \sqrt{\log\left(\frac{2}{\rho}\right)} \right]^2 + 4Kk_1^2 \right\} \\ & - pen(\hat{m}) + pen(m) + 4Kk_1^2 r|m| + 8Kk_2 r\xi \left[\sqrt{k_2 \frac{3}{4}} + \sqrt{\log\left(\frac{2}{\rho}\right)} \right]^2. \end{aligned}$$

D'après la forme de la fonction de pénalité donnée en (6.2.19), si nous choisissons

$$K = \frac{2c(\rho)\,(1+C)}{C-1},$$

$$K_1 = 8Kk_2\left[\sqrt{k_2\frac{3}{4}} + \sqrt{\log\left(\frac{2}{\rho}\right)}\right]^2,$$

et

$$K_2 = 2K_1 + 4Kk_1^2,$$

alors nous avons

$$\left(\frac{2}{1+C}\right) l\,(s,\tilde{s}) \;\leqslant\; l\,(s,\bar{s}_m)\left(\frac{2C}{1+C}\right) + pen(m) + 4Kk_1^2 r|m| + rK_1\xi.$$

En remarquant que $4Kk_1^2 r|m| \leqslant pen(m)$, nous obtenons la majoration suivante :

$$\left(\frac{2}{1+C}\right) l\,(s,\tilde{s}) \;\leqslant\; l\,(s,\bar{s}_m)\left(\frac{2C}{1+C}\right) + 2pen(m) + rK_1\xi.$$

D'où

$$l\,(s,\tilde{s}) \;\leqslant\; Cl\,(s,\bar{s}_m) + (1+C)pen(m) + \frac{(1+C)}{2}rK_1\xi.$$

En intégrant, nous obtenons

$$E_s\left[l\,(s,\tilde{s})\right] \;\leqslant\; Cl\,(s,\bar{s}_m) + (1+C)pen(m) + \frac{(1+C)}{2}rK_1\Sigma.$$

Ce qui conclue la démonstration du théorème 6.2.1.

Cette majoration est obtenu sur l'évènement $\Omega_{m_f}(\rho)$ donnée en (6.2.21). Nous cherchons maintenant à évaluer la probabilité de cet évènement. Nous allons nous intéresser plus particulièrement à la probabilité de son evènement complémentaire.

6.2.4.5 Contrôle de $P\left(\Omega_{m_f}(\rho)^c\right)$

En remaniant l'évènement $\Omega_{m_f}(\rho)^c$, nous cherchons à contrôler pour $J \subset m_f$ et $i = 1,\ldots,r$, la quantité suivante :

$$N_J(i) - \sum_{I\in m_0} |I\cap J| s_I(i).$$

C'est une somme de variables aléatoires indépendantes et l'inégalité de Bernstein permet de contrôler ce type de somme.

Lemme 6.2.10. *Soit l'évènement $\Omega_{m_f}(\rho)$ défini pour ρ vérifiant (6.2.18) par:*

$$\Omega_{m_f}(\rho) = \left\{ \forall i \in \{1,..,r\}, \forall\, J \subset m_f \;;\; \frac{N_J(i)}{\mid J \mid} \geq \frac{\rho}{2} \right\}.$$

Supposons qu'il existe une constante strictement positive Γ telle que la taille du plus petit segment de la partition m_f vérifie

$$\Gamma_{m_f} = \inf_{J \in m_f} |J| \geq \Gamma \, \log^2(n),$$

alors pour tout $a > 0$, il existe une constante $C(\Gamma, \rho, a, r) > 0$ telle que

$$P\left(\Omega_{m_f}(\rho)^c\right) \leqslant \frac{C(\Gamma, \rho, a, r)}{n^a}.$$

Preuve.

Tout d'abord, en écrivant pour tout $J \subset m_f$, $|J| = \sum_{I \in m_0} |I \cap J|$ et par l'hypothèse sur s donnée en (6.2.18), nous obtenons la suite d'inégalités suivante:

$$\begin{aligned}
P\left(\Omega_{m_f}(\rho)^c\right) &\leqslant \sum_{i=1}^{r} \sum_{J \subset m_f} P\left(N_J(i) \leqslant \rho\left(1 - \frac{1}{2}\right) \sum_{I \in m_0} \mid I \cap J \mid \right) \\
&\leqslant \sum_{i=1}^{r} \sum_{J \subset m_f} P\left(N_J(i) - \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \leqslant -\frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \right) \\
&\leqslant \sum_{i=1}^{r} \sum_{J \subset m_f} P\left(N_J(i) - \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \geq \frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \right).
\end{aligned}$$

Nous rappelons l'inégalité de concentration de Bernstein que nous allons utiliser.

Corollaire 6.2.11. *Soit $n \in \mathbb{N}$. Soient $X_1, \ldots, X_n$ n variables aléatoires indépendantes. Supposons qu'il existe des réels positifs e et c tels que pour tout $k \geq 2$*

$$\sum_{t=1}^{n} E|X_t|^k \leqslant \frac{k}{2} e c^{k-2}. \tag{6.2.40}$$

Soit $S_n = \sum_{t=1}^{n} [X_t - E(X_t)]$, alors pour tout $x > 0$,

$$P(S_n \geq x) \leqslant \exp\left(-\frac{x^2}{2(e + cx)} \right).$$

Remarque 6.12. *Si les variable X_t sont bornées, $|X_t| \leqslant b'$, alors l'hypothèse* (6.2.40) est vérifiée avec

$$e = \sum_{t=1}^{n} E(X_t^2) \quad \text{et} \quad c = b'/3.$$

Nous fixons $J \subset m_f$ et $i \in \{1, .., r\}$. Pour évaluer

$$P\left(N_J(i) - \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \geq \frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i)\right),$$

nous appliquons le corollaire précédent avec $b' = 1$, $e = \sum_{I \in m_0} |I \cap J| \; s_I(i)$ et en posant $x = \frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i)$. Nous obtenons la majoration suivante :

$$\begin{aligned} P\left(N_J(i) - \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \geq \frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i)\right) & \\ \leqslant \exp\left(-\frac{1/4 \left(\sum_{I \in m_0} \mid I \cap J \mid s_I(i)\right)^2}{2 \left(\sum_{I \in m_0} \mid I \cap J \mid s_I(i) + \frac{1}{6} \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \right)}\right) & \\ \leqslant \exp\left(-\frac{1/4 \sum_{I \in m_0} \mid I \cap J \mid s_I(i)}{2 \left(1 + \frac{1}{6}\right)}\right) & \\ \leqslant \exp\left(-\frac{3}{28} \sum_{I \in m_0} \mid I \cap J \mid s_I(i)\right). & \end{aligned}$$

Soit Γ_{m_f} la taille du plus petit segment de la partition m_f et rappelons que ρ vérifie (6.2.18), alors nous obtenons la majoration suivante :

$$\begin{aligned} \sum_{i=1}^{r} \sum_{J \subset m_f} P\left(N_J(i) - \sum_{I \in m_0} \mid I \cap J \mid s_I(i) \geq \frac{1}{2} \sum_{I \in m_0} \mid I \cap J \mid s_I(i)\right) & \\ \leqslant r \; |m_f|^2 \; \exp\left(-\frac{3}{28} \Gamma_{m_f} \rho\right) & \\ \leqslant \frac{r \; n^2}{\Gamma_{m_f}^2} \; \exp\left(-C \; \Gamma_{m_f} \rho\right), & \end{aligned}$$

avec $C = \frac{3}{28}$.

Prenons $\Gamma_m \geq \Gamma \log^2(n)$, nous en déduisons que sous réserve que $C\Gamma\rho \geq 1$, il existe une constante $C(\Gamma, \rho, a, r)$ telle que

$$P\left(\Omega_{m_f}(\rho)^c\right) \leqslant \frac{C(\Gamma, \rho, a, r)}{n^a},$$

pour $a > 0$.

La contrainte sur la taille du plus petit segment de la partition la plus fine m_f donnée en (6.2.17) dans le théorème (6.2.1) permet d'obtenir un contrôle de la probabilité $P\left(\Omega_{m_f}(\rho)^c\right)$. Cette probabilité tend vers 0 quand n tend vers l'infini. Le risque entre s et son estimateur $\tilde{s}$ donné en (6.2.20) est donc calculé sur l'évènement $\Omega_{m_f}(\rho)$ dont la probabilité est proche de 1 lorsque n est grand.

Sur le complémentaire, nous pouvons montrer que $E_s\left[\, l(s,\tilde{s})\ \mathbb{1}_{\Omega_{m_f}(\rho)^c}\right]$ est majoré par une quantité qui est asymptotiquement négligeable devant la borne majorante du risque de l'estimateur obtenu et donnée en (6.2.20). En effet, par l'inégalité de Cauchy-Schwarz, nous avons

$$E_s\left[\, l(s,\tilde{s})\ \mathbb{1}_{\Omega_{m_f}(\rho)^c}\right] \leqslant E_s\left[\, l(s,\tilde{s})^2\right]^{1/2} P\left(\Omega_{m_f}(\rho)^c\right)^{1/2}.$$

Or en remarquant que $|l(s,u)| \leqslant rn\log(2/\rho)$ pour tout $u \in S_m\left(\frac{\rho}{2}\right)$ et d'après le lemme 6.2.10, nous avons la majoration suivante :

$$E_s\left[\, l(s,\tilde{s})\ \mathbb{1}_{\Omega_{m_f}(\rho)^c}\right] \leqslant \frac{r\log(2/\rho)C(\rho,\Gamma,a,r)^{1/2}n}{n^{a/2}}.$$

En prenant, $a > 2$, ce terme est donc négligeable comparé à la borne supérieure proposée dans (6.2.20).

6.3 Cas de dépendance markovienne d'ordre 1

Nous supposons maintenant que sur chaque segment de la partition m_0, les Y_t sont générés selon une chaîne de Markov homogène d'ordre 1. Dans la mesure où ce modèle sera utilisé lors des applications effectuées dans le chapitre suivant, nous donnons les notions théoriques nécessaires à l'application, comme le contraste, l'estimateur du maximum de vraisemblance et le théorème qui donne la forme de la pénalité. Nous présentons les points de difficultés rencontrés par rapport au modèle indépendant et donnons les solutions envisagées.

Modèle, contraste, estimateur du minimum de contraste et perte

Nous considérons $Y_1, Y_2, ..., Y_n$, n variables aléatoires à valeurs dans l'ensemble $\mathcal{Y} = \{1, \ldots, r\}$ où $r \geq 2$. Nous supposons que ces variables sont générées selon une chaîne de Markov homogène par morceaux : il existe une partition m_0 de $\{1, ..., n\}$ telle que la transition et la loi initiale de la chaîne sont constantes sur chacun des segments de cette partition. Pour tout segment $I \in m_0$, nous supposons que

- $(Y_I)_{I \in m_0}$ est une suite de variables indépendantes.
- $Y_I = (Y_t)_{t \in I}$ est une chaîne de markov de matrice **irréductible et apériodique** de distribution initiale π_I et de transition s_I.

- la chaîne Y_I est stationnaire.

Nous écrivons s sous la forme :

$$s = \sum_{I \in m_0} \sum_{i,j=1}^{r} s_I(i,j)\ \mathbb{1}_{I \times (i,j)},$$

et l'objectif est d'estimer s par une méthode de sélection de modèle.

Pour tout partition m de $\{1, \ldots, n\}$, nous donnons successivement le modèle associé à m, la vraisemblance, le contraste, l'estimateur du minimum de contraste et enfin la perte :

- Nous définissons le **modèle** associé à la partition m par :

$$S_m = \left\{ \begin{array}{l} u = \sum_{J \in m} \sum_{i,j=1}^{r} u_J(i,j)\ \mathbb{1}_{J \times (i,j)} \\ \text{avec } u_J(i,j) \geq 0, \forall J \in m,\ \forall (i,j) \in \{1, \ldots, r\}^2 \\ \text{et } \int_{\{1,..,n\} \times \{1,..,r\}^2} u(x)\ d\mu_n(x) = 1 \end{array} \right\},$$

où $\mu_n = \mathcal{U}\{1,..,n\} \times \mathcal{U}\{1,..,r\} \times \mathcal{C}\{1,..,r\}$.

- Soit $u \in S_m$, nous notons P_u la loi de $Y = (Y_1, ..., Y_n)$ associée à u. Nous considérons que les lois stationnaires sont des paramètres nuisibles et nous travaillons alors avec la **vraisemblance** partielle suivante :

$$V'(Y_1, ..., Y_n; u) = \prod_{J \in m} \prod_{i,j=1}^{r} s_J(i,j)^{N_J(i,j)} \tag{6.3.41}$$

$$\text{où } N_J(i,j) = \sum_{t-1 \in J} \sum_{t \in J} 1_{\{Y_{t-1}=i,\ Y_t=j\}}.$$

Pour $J \in m$ et $(i,j) \in \{1, \ldots, r\}^2$, $N_J(i,j)$ représente le comptage du nombre de i suivi de j dans le segment J.

- Comme dans la section précédent, nous considérons un **contraste** sur $\mathcal{S}_m$ défini par :

$$\gamma'(u) = \gamma(P_u) = -\log\left[V'(Y_1, ..., Y_n; u)\right]$$

$$= -\sum_{J \in m} \sum_{i,j=1}^{r} N_J(i,j) \log\left[s_J(i,j)\right]. \tag{6.3.42}$$

En considérant que s et u sont construit sur la même partition plus fine que m_0 et que m

$$m' = (I \cap J)_{I \in m_0\ ;\ J \in m},$$

nous avons que

$$\gamma'(u) - \gamma'(s) = \sum_{I \in m_0} \sum_{J \in m} \sum_{i,j=1}^{r} N_{I \cap J}(i,j) \log\left(\frac{s_I(i,j)}{u_J(i,j)}\right).$$

- Notons $N_J(i,+) = \sum_{j=1}^{r} N_J(i,j)$. **L'estimateur de s du minimum de contraste** sur $\mathcal{S}_m$ est l'estimateur du maximum de vraisemblance défini par:

$$\hat{s}_m = \sum_{J \in m} \sum_{i,j=1}^{r} \left(\frac{N_J(i,j)}{N_J(i,+)}\right) 1_{J \times (i,j)}. \tag{6.3.43}$$

- Nous définissons la **perte** l pour tout $u \in \mathcal{S}_m$ par:

$$l(s,u) = \sum_{I \in m_0} \sum_{J \in m} \sum_{i,j=1}^{r} |I \cap J| s_I(i,j) \log\left(\frac{s_I(i,j)}{u_J(i,j)}\right).$$

Résultat principal

Nous avons obtenu le théorème suivant de majoration du risque de l'estimateur penalisé.

Theorem 6.3.1. *Soit $\mathcal{F}_n$ l' ensemble des partitions de $\{1,..,n\}$ construite sur une partition dite la plus fine m_f telle que la taille Γ_{m_f} du plus petit segment de m_f vérifie:*

$$\Gamma_{m_f} = \Gamma \log^2(n)$$

pour Γ constante absolue.

Supposons qu'il existe une constante strictement positive ρ telle que pour tout $I \in m_0$ et tout $(i,j) \in \{1,\ldots,r\}^2$

$$s_I(i,j) \geq \rho.$$

Il existe des constantes K_1 et K_2 dépendantes de ρ et de r, telles si nous choisissons pour tout $m \in \mathcal{F}_n$

$$pen(m) = |m| \left(K_1 \log\left(\frac{n}{|m|}\right) + K_2\right),$$

alors en choisissant l'estimateur pénalisé suivant

$$\hat{m} = \operatorname*{argmin}_{m \in \mathcal{F}_n} \left\{\gamma'(\hat{s}_m) + pen(m)\right\},$$
$$et\ \tilde{s} = \hat{s}_{\hat{m}},$$

nous disposons de la majoration suivante du risque de l'estimateur pénalisé $\tilde{s}$

$$E_s[\, l(s,\tilde{s})] \leqslant C_1(\rho) \inf_{m \in \mathcal{F}_n} \left\{ \inf_{u \in \mathcal{S}_m} l(s,u) + pen(m) \right\} + C_2(r,\rho).$$

Idées de preuve

Le problème principal qui se pose est que nous ne disposons pas d'inégalité de concentration pour le contrôle de $V_{m'}$ défini par :

$$V_{m'} = \sup_{u' \in S_{m'}\left(\frac{\rho}{2}\right)} \left\{ \frac{|\bar{\gamma}'(u') - \bar{\gamma}'(s_m)|}{w_{m'}(u')} \right\},$$

autour de son espérance quand Y est une chaîne de Markov homogène d'ordre 1. Pour palier à ce manque, l'idée est de procéder en deux étapes : la première consiste à chercher un contrôle de la probabilité $P(\bar{\gamma}'(u) - \bar{\gamma}'(u') \geq x)$ par une inégalité exponentielle. Et la seconde est d'obtenir le passage au supremum. Nous donnons les solutions envisagées pour ces deux calculs ainsi que pour le calcul de deux autres quantités.

• **Majoration de la variance de $\gamma'(u) - \gamma'(u')$ pour $u' \in \mathcal{S}_{m'}(\frac{\rho}{2})$ et $u \in \mathcal{S}_m(\frac{\rho}{2})$:** le calcul de cette quantité se réduit au calcul du terme suivant pour $I \in m_0$, $K \in m$ et $(i,j) \in \{1,.,r\}^2$

$$var_s \left[\sum_{J \in m} N_{I \cap J}(i,j) \log \left(\frac{s_I(i,j)}{t_J(i,j)} \right) \right].$$

Il fait intervenir le calcul de la variance du comptage $N_{I \cap J}(i,j)$ et celui de la covariance des deux comptages $N_{I \cap J}(i,j)$ et $N_{I \cap K}(i,j)$ pour tout segment $K > J$. La notation $K > J$ indique que le segment K a pour support des indices supérieurs aux indices du support de J sur la séquence. Pour obtenir un calcul des deux termes, nous pouvons utiliser l'ergodicité uniforme de la chaîne de Markov : les transitions p_I convergent sur chaque segment I vers sa loi invariante π_I avec une vitesse qui est le rayon spectrale de la matrice de transition symétrisée $p_I^* p_I$ (résultat de Fill énoncé dans le livre de Duflo [24]). La majoration de la variance du contraste permettra de définir, comme dans le cas indépendant, une pseudo-distance d^2.

• **Contrôle de $P(\bar{\gamma}'(u) - \bar{\gamma}'(u') \geq x)$ pour $u' \in \mathcal{S}_{m'}(\frac{\rho}{2})$ et $u \in \mathcal{S}_m(\frac{\rho}{2})$:** pour obtenir ce contrôle, nous pouvons reprendre la démarche suivie par Lezaud [46] qui propose une inéglité de concentration dans le cadre des chaînes de Markov stationnaire irréversible.

• **Inégalité de concentration pour $\sup_{u' \in S_{m'}(\rho_\varepsilon)} \left\{ \frac{|\bar{\gamma}'(u') - \bar{\gamma}'(u)|}{w_{m'}(u')} \right\}$:** le passage au supremum s'obtient à l'aide d'une technique de chaînage.

• **Calcul de $P\left(\Omega_{m_f}\left(\frac{\rho}{2}\right)^C\right)$:** la quantité $N_J(i,j)$ n'est pas une somme de variables aléatoires indépendantes et donc l'inégalité de Bernstein (comme vu dans la section précédente) ne peut être appliquée. Pour obtenir l'indépendence à J fixé, l'idée est de se placer sur les segments $I \cap J \neq \emptyset$, $I \in m_0$ et de conditionner par le nombre comptage de i sur chacun des segments. En effet, $N_J(i,j) = \sum_{I \in m_0} N_{J \cap I}(i,j)$ conditionnellemt aux $N_{J \cap I}(i,+)$ pour tout $I \in m_0$, est une somme de variables aléatoires indépendantes.

Chapitre 7

Applications au génome

7.1 Introduction

7.1.1 Présentation du problème

Ce dernier chapitre est consacré à la recherche de régions homogènes dans des séquences d'ADN.

Une séquence d'ADN est constituée de bases ou nucléotides codées par les 4 lettres : A pour adénine, G pour guanine, T pour thymine et C pour cytosine. Cet enchaînement constitue le message génétique permettant à la cellule de fonctionner selon le programme de l'espèce. Les séquences d'ADN sont formées de régions aux fonctions biologiques déterminées comme par exemple un gène, une zone codante ou une zone non-codante. L'identification de ces régions permet aux biologistes de comprendre la structure de la séquence étudiée. A cause de la taille importante des séquences qui peut atteindre plusieurs milliards de paires de bases (bp), les biologistes sont à la recherche de méthodes statistiques permettant cette identification. Selon l'analyse de séquences simples, il existe à l'intérieur d'une région biologiquement déterminée, une stabilité en fréquence des différentes bases. Du point de vue du statisticien, il s'agit de choisir un modèle qui prenne en compte cette réalité biologique, et de l'utiliser pour délimiter les régions, dites homogènes, de la séquence étudiée.

Une séquence d'ADN de longueur n est représentée par une suite $Y_1, \ldots, Y_n$ de n variables aléatoires où Y_t est à valeurs dans l'alphabet de l'ADN $\{A, C, G, T\}$. Le modèle le plus simple consiste à supposer que :

- les variables Y_t sont indépendantes,
- la séquence est composée de segments tels que la distribution des variables Y_t sur chaque segment est la même et diffère d'un segment à l'autre.

Nous l'appelons **le modèle indépendant.**

Le problème de segmentation des séquences d'ADN a déjà une longue histoire. Braun et Muller [12] dresse un état de l'art des différents modèles issus du modèle simple et des méthodes, appelées méthodes de segmentation. La plupart de ces modèles sont fondées sur une hypothèse supplémentaire par rapport au modèle simple, une hypothèse d'hétérogénéité notée :

(H) : les différents segments de la séquence peuvent être classés dans un ensemble fini d'états, et la distribution des variables Y_t dépend de l'état dans lequel elles se situent.

Ces états, dits états cachés, sont décrits par une suite sous-jacente aux observations. Les auteurs qui ont étudié ce modèle sont nombreux. Parmi ceux qui supposent qu'il existe plusieurs segments, Fu *et. al* [26] estime la position des segments par maximum de vraisemblance. Ils font les hypothèse suivantes : il n'existe que deux états cachés possibles et les distributions dans chaque état ainsi que le nombre segments sont supposés connus.
Si le nombre, la taille des segment et les distributions dans chaque état sont inconnus, le problème devient plus complexe. Churchill [20], Boys *et al* [56] et Muri [52] considèrent un modèle de chaînes de Markov cachées : la suite sous-jacente aux observations est une chaînes de Markov d'ordre 1 à espace d'états fini et fixé mais qui peut être supérieur à deux. Ils estiment la distribution de la suite sous-jacente conditionnellement aux observations et les distributions des variables Y_t dans chaque état par un algorithme EM. Ce qui permet l'identification des régions homogènes et de les classer dans un nombre fini d'états. Muri [52] étend les résultats au cas où la suite des variables Y_t est une chaîne de Markov de différents ordres sur chaque segment et propose de plus une estimation par méthodes de Monte Carlo par chaînes de Markov (MCMC).
Cette hypothèse d'hétérogénéité (**H**) contraint la segmentation de la séquence. Pour ces approches, la compléxité tient dans le modèle lui-même.

Si l'hypothèse d'hétérogénéité (**H**) n'est pas prise en compte, le modèle est un modèle plus simple (par exemple, le modèle indépendant), et le problème de segmentation des séquences d'ADN peut s'inscrire dans un problème de détection de ruptures multiples : les ruptures correspondent aux bornes des segments, i.e. aux changements dans la distribution des variables Y_t. Le problème consiste alors à déterminer le nombre de ruptures qui existent dans la séquence et à les localiser. Pour résoudre ce problème, plusieurs approches apparaîssent dans la littérature : un algorithme séquentiel de test basé sur un calcul d'entropie (Olivier et al [36]) ou basé sur le rapport de vraisemblance (algorithme proposé dans Braun et Müller [12]). Dans une autre approche, Braun et al [11] proposent d'estimer tous les instants de ruptures par un critère de quasi-déviance pénalisé, critère de Schwarz modifié. Ils obtiennent des résultats de consistance pour l'estimateur des ruptures et du nombre de ruptures. En pratique, la pénalité est calibrée par simulations de telle sorte que le nombre de ruptures estimé soit proche du vrai nombre de ruptures.

Dans la section 6.2 du chapitre précédent, nous proposons une méthode d'estimation d'une fonction s définie par l'équation (6.2.2) dont les coefficients sont :

$$s_I(i) = P(Y_t = i) \quad \forall t \in I,\ I \in m_0 \text{ et } i \in \mathcal{Y} = \{1, \ldots, r\}.$$

Dans le cadre des séquences d'ADN, puisque Y_t est à valeurs dans l'alphabet $\{A, C, G, T\}$, nous prenons $\mathcal{Y} = \{1, 2, 3, 4\}$. Donc $r = 4$ dans tous les résultats théoriques obtenus. Le problème est posé en terme de partitions : les segments de la séquence sont les segments d'une partition m_0. Sur chaque segment $I \in m_0$, la distribution de probabilité des variables Y_t est $s_I(.)$. L'estimation de s donne alors l'estimation de la meilleure partition et des distributions des variables Y_t sur chaque segment de cette partition.
Comme Braun *et. al* [11], nous estimons s par un critère pénalisé qui est l'opposé de la log-vraisemblance pénalisée. La complexité de ces deux approches se situe non pas dans le modèle mais dans la collection de partitions que nous considérons. Notre approche diffère de celle de Braun *et. al* en deux points : c'est une approche non asymptotique dont le but n'est pas d'estimer tous les instants de ruptures (la vraie partition) mais de sélectionner l'estimateur de s qui a le plus petit risque, défini pour un estimateur noté $\hat{s}_m$ par l'équation (6.2.13). Le théorème 6.2.1 donne la forme de la pénalité pour laquelle nous obtenons un contrôle du risque de l'estimateur final défini en (6.2.20). Et c'est dans cette fonction de pénalité que se trouve la complexité de la collection de partitions. Pour obtenir ce résultat théorique, nous avons supposé que la distribution sur chaque segment est "proche" de la distribution uniforme sur $\{1, 2, 3, 4\}$. Ceci nous a permis de supposer l'existence de $\rho > 0$ tel que

$$P(Y_t = i) \geq \rho \quad \forall t \in I,\ I \in m_0 \text{ et } i \in \mathcal{Y} = \{1, 2, 3, 4\},$$

C'est l'hypothèse donnée par l'équation (6.2.18).

Dans la section 6.3 du chapitre précédent, nous avons étendu nos résultats au cas plus complexe suivant : sur chaque segment $I \in m_0$, nous supposons que la suite de variables Y_t, $Y_I = (Y_t)_{t \in I}$ est une chaîne de Markov d'ordre 1 stationnaire, et que les suites $(Y_I)_{I \in m_0}$ sont des suites indépendantes. Nous l'appelons **le modèle markovien**. Dans ce modèle, nous ne prenons pas en compte la même information que dans le modèle indépendant, ainsi la détection de ruptures pourra être différente et révéler d'autres fonctions biologiques.

7.1.2 Algorithme

Comme pour la détection de ruptures multiples dans la moyenne par sélection de modèle, la méthode requiert la visite de toutes les partitions possible de la grille $\{1, \ldots, n\}$. Si l'échantillon observé est très grand, comme c'est le cas des séquences d'ADN, l'implémentation de cette méthode est impossible.

Nous proposons d'adapter l'**algorithme hybride** mis en oeuvre pour la détection de ruptures multiples de moyenne sur des signaux Gaussiens de très grandes tailles et présenté dans le chapitre 5. Il consiste à réduire de façon significative la collection de partitions avant d'appliquer la méthode. Pour appliquer l'algorithme hybride dans le cadre des séquences d'ADN, il suffit de changer le contraste empirique et la fonction de pénalité par ceux obtenus dans ce cadre. Le contraste empirique, noté γ', est définie dans le cas indépendant (resp. le cas markovien) par l'équation (6.2.8) (resp. l'équation (6.3.42)). La forme de la fonction de pénalité est la même dans les deux cas. Elle est donnée en (6.2.19). Nous

rappelons succintement les trois étapes de l'algorithme hybride et donnons les paramètres que nous avons choisi pour l'application :

1. Dans la première étape, nous appliquons tout d'abord l'algorithme CART qui construit une sous-collection de partitions $\widetilde{\mathcal{M}}_n^{(\text{cart})}$. La partition sélectionnée est ensuite celle qui minimise sur cette sous-collection le critère défini par :

$$\gamma'(\hat{s}_m) + \hat{\beta}|m|,$$

où $\hat{s}_m$ est l'estimateur du minimum de contraste pour la partition m défini par l'équation (6.2.9) (resp. (6.3.43)) et $\hat{\beta}$ est obtenu par la méthode heuristique (cf la sous-section 5.6.2 du chapitre 5).
Pour l'application de l'algorithme CART sur les deux séquences étudiées, nous fixons la taille minimale de découpe à $l_{min} = 1000$, ce qui est largement raisonnable par rapport à la taille moyenne des gènes présents dans les séquences. Pour des applications supplémentaires sur des extraits de séquences, nous réduisons ce paramètre à $l_{min} = 500$.

2. Si $\hat{D}_c$ est la dimension de la partition sélectionnée par la première étape, nous extrayons de l'arbre maximal construit par CART, l'arbre de dimension $4 \times \hat{D}_c$. La partition associée représente une nouvelle grille.

3. Dans la dernière étape, nous appliquons la méthode sur la nouvelle grille par un algorithme appelée algorithme de recherche exhaustive. Nous choisissons la meilleure partition $\hat{m}$ à l'aide d'un critère pénalisé définie par :

$$\gamma'(\hat{s}_m) + pen(m),$$

où la fonction de pénalité est définie sous la forme générale par :

$$pen(m) = |m| \left(K_1 \log\left(\frac{n}{|m|}\right) + K_2 \right).$$

Les valeurs optimales des constantes K_1 et K_2 ne sont pas accessible théoriquement. Nous pouvons voir s défini par (6.2.2) comme un histogramme contre la mesure μ_n donnée en (6.2.4). Nous supposons alors que la fonction de pénalité est proportionnelle à celle calibrée par Birgé et Rozenholc [9] dans le cadre de l'estimation de densité par histogrammes par une méthode de sélection de modèle. Pour toute partition m de $\mathcal{M}_n$, la fonction de pénalité est définie par :

$$\begin{aligned} pen(m) &= \alpha|m| \left(\log\left(\frac{n}{|m|}\right) + 2.5 \right) \\ &= 2\alpha \frac{|m|}{2} \left(\log\left(\frac{n}{|m|}\right) + 2.5 \right) \end{aligned} \tag{7.1.1}$$

où n est la taille de l'échantillon observé et α une constante strictement positive. L'estimation de α est ensuite obtenue par la méthode heuristique.

Remarque. Il serait intéressant de calibrer les constantes K_1 et K_2 de façon optimale en suivant la même procédure que celle proposée dans le chapitre 3. En pratique, si les séquences observées sont très grandes, le $\log n$ présent dans la fonction de pénalité est prédominant et une pénalité uniquement proportionnelle à la dimension de la partition pourrait être envisagée.

La mise en oeuvre de cet algorithme a été réalisé par le logiciel Matlab.

Dans la section 7.2, nous présentons les résultats de l'application de l'algorithme hybride sur la bactérie B.subtilis et dans la section 7.3 sur le bactériophage Lambda. Pour ces deux applications, nous nous intéressons particulièrement au cas où les variables Y_t sont dépendantes : nous considérons le modèle markovien. Nous le comparons ensuite au modèle indépendant.

7.2 Recherche des régions homogènes de la bactérie B.subtilis

La séquence complète de la bactérie Bacillus Subtilis comporte 4214630 *bp*. L'institut Pasteur a publié sur le réseau informatique, les données de cette bactérie (http://genolist.pasteur.fr), ainsi qu'une description très précise de la localisation et de la fonction des gènes présents.

Dans cette section, nous appliquons l'algorithme hybride revisité sur un extrait de la séquence de longueur 200000 *bp* : plus précisément, nous l'étudions des bases allant de 1 à 200000.

7.2.1 Résultats pour le modèle markovien

Nous décrivons les résultats obtenus au cours des trois étapes de l'algorithme :

1. La partition de dimension 13 est sélectionnée. Les ruptures associées sont : 9722, 14800, 30294, 35533, 90504, 101200, 117104, 144656, 158486, 160859, 176325 et 194718.

2. L'arbre de dimension $v \times 13 = 56$ est extrait de l'arbre maximal construit par CART et forme une nouvelle grille de ruptures potentielles.

3. L'algorithme de recherche exhaustive est appliqué sur la nouvelle grille. Le graphe de $(f_n(D), \gamma_n(\hat{s}_D))$ pour $D = 1, \ldots, 56$ et la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ sont respectivement représentés à gauche et à droite de la Figure 7.1. La partition finale est de dimension 12. Elle est représentée sur la Figure 7.2 avec la séquence observée, ses gènes et leurs fonctions (cf légende). Pour une meilleure lecture graphique, la région k, associée au segment k de la partition finale, est indiquée par le numéro k encadré (cf 7.2). Les deux ruptures, localisées en 144656 et 158486 et sélectionnées par la

première étape ne sont pas retenues par cette dernière étape, et une nouvelle rupture est sélectionnée (150287).

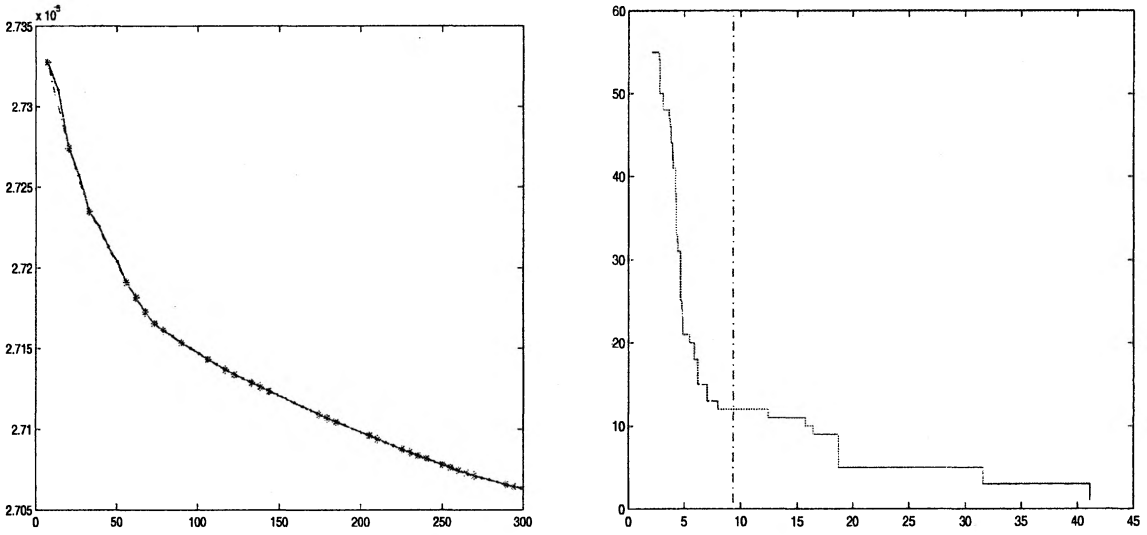
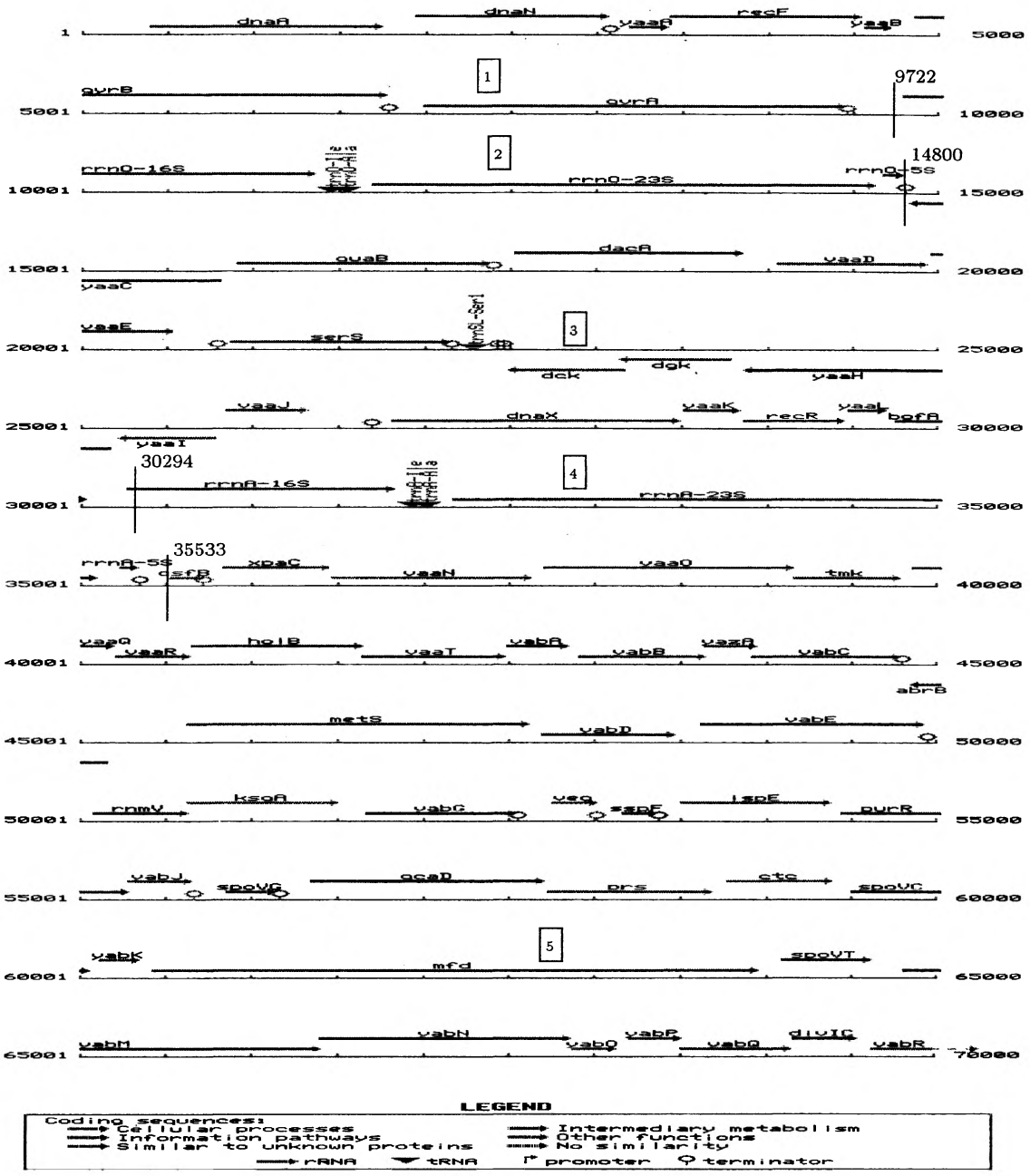


FIG. 7.1: $(f_n(D), \gamma_n(\hat{s}_D))$ *pour* $D = 1, \ldots, 56$ *(à gauche) et* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ *(à droite) où le symbole -. correspond à* $2\hat{\alpha}$.

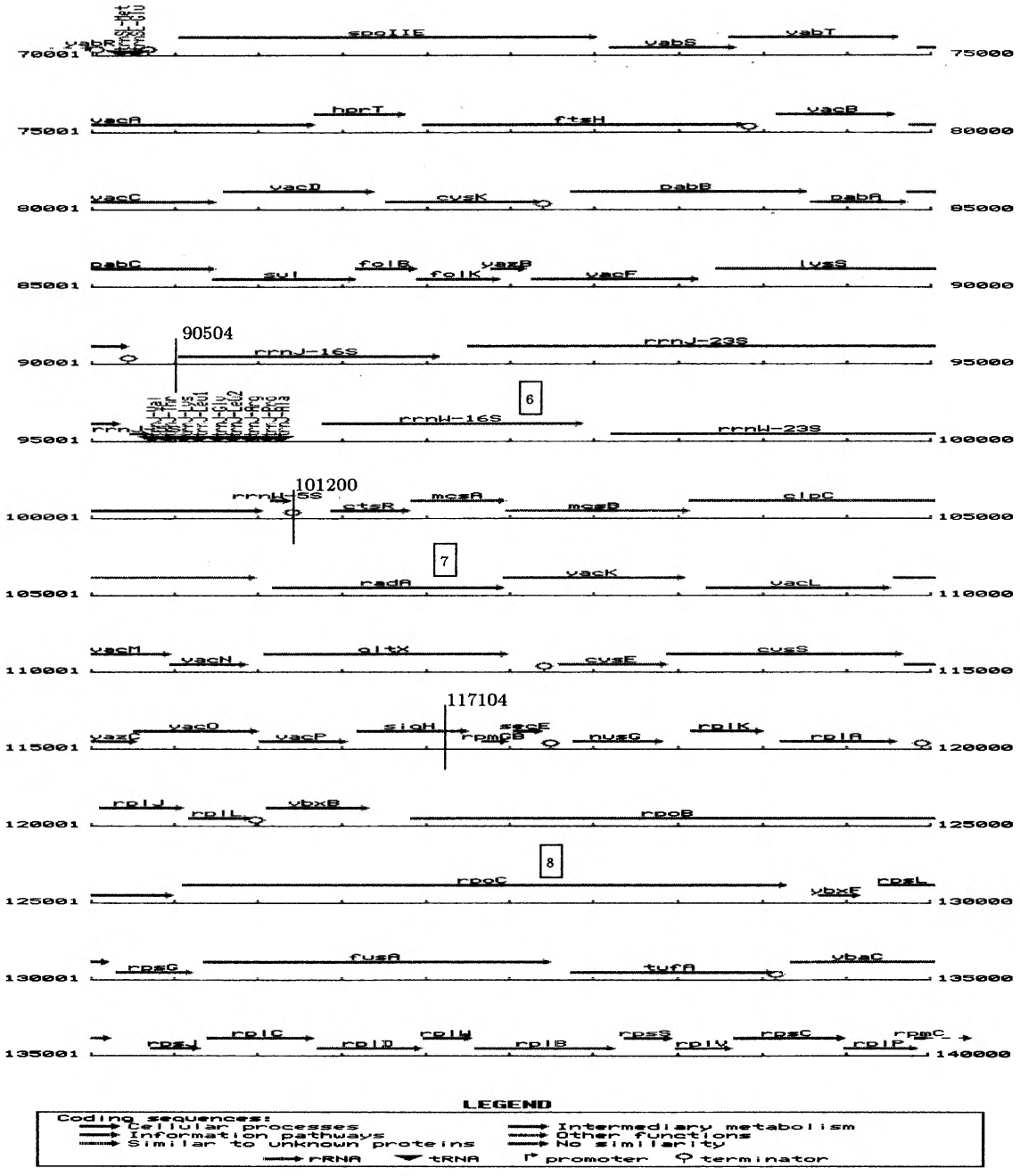
La première remarque face à ce résultat est que la méthode sélectionne une partition qui ne coupe presque jamais de gènes. En effet, nous remarquons que seul le gène *sigH* localisé entre les instants de ruptures 116597 et 117250 est coupé au 2/3 de sa longueur. Ce qui est tout à fait encourageant. Les ruptures qui n'ont pas été retenues dans la sélection finale (144656 et 158486) coupent respectivement les gènes *secY* et *gerD*, alors que la nouvelle rupture (150287) ne coupe pas de gènes (cf Figure 7.2).



LEGEND

Coding sequences:
- Cellular processes
- Information pathways
- Similar to unknown proteins
- Intermediary metabolism
- Other functions
- No similarity

rRNA tRNA promoter terminator


90504
101200
117104
6
7
8
LEGEND
Coding sequences:
Cellular processes
Information pathways
Similar to unknown proteins
Intermediary metabolism
Other functions
No similarity
rRNA
tRNA
promoter
terminator

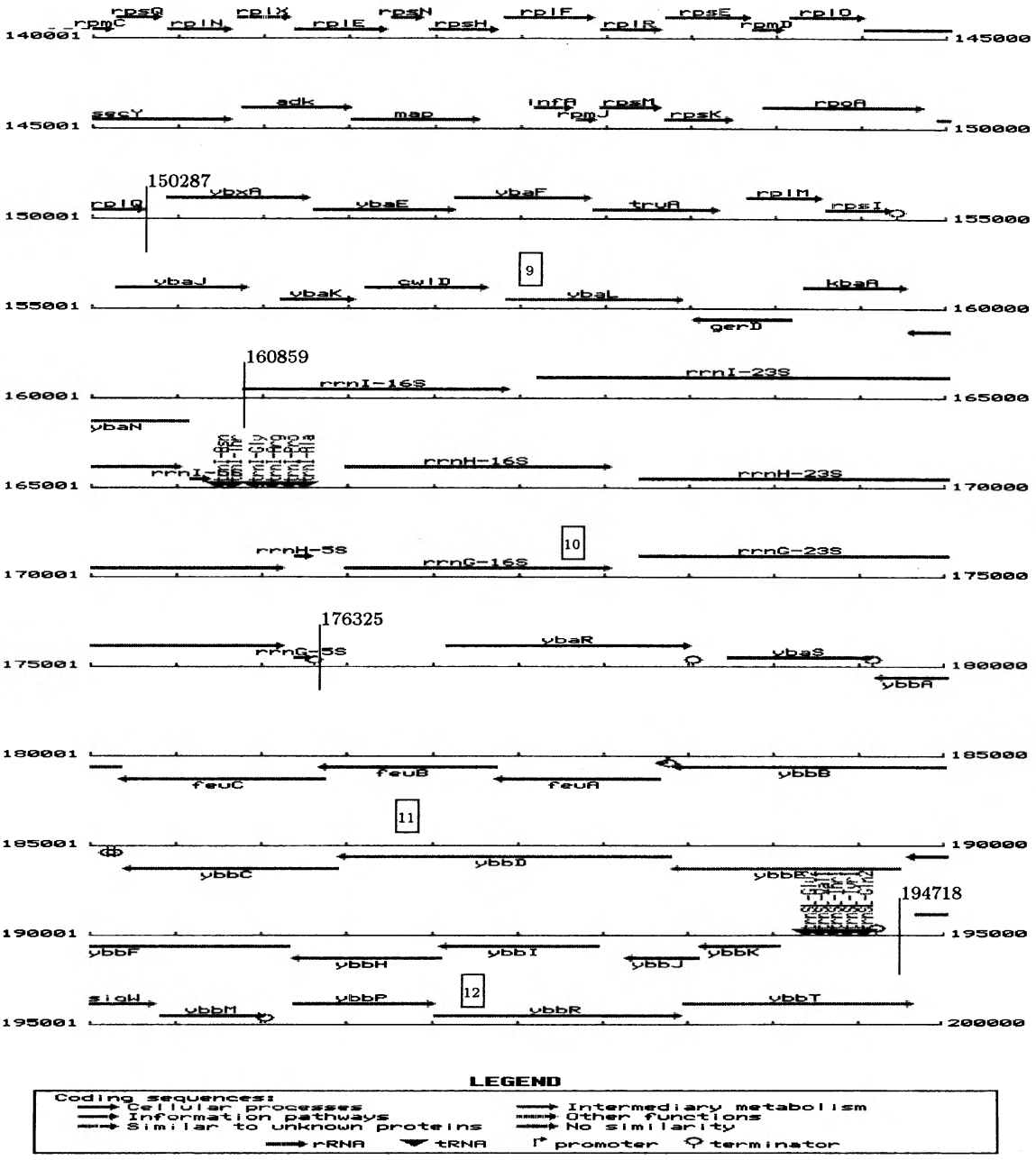


FIG. 7.2: *Régions homogènes détectées sur la bactérie B. subtilis. Les flèches indiquent le sens de transcription des gènes.*

nom des gènes	coordonnées	description
gyrA	6993..9455	DNA gyrase (sous-unité A)
rrnO – *16S*	9809..11361	ribosomal RNA-16S
rrnO – *5S*	14690..14801	ribosomal RNA-5S
yaac	14848..15792	inconnu; similaire aux protéines inconnues
guaB	15913..17376	inosine-monophosphaste dehydrogenase
rrnA – *16S*	30277..31829	ribosomal RNA-16S
rrnA – *5S*	35235..35346	ribosomal RNA-5S
csfB	35529..35720	gène transcrit sigma-F
lysS	88724..90220	lysyl-tRNA synthetase
rrnJ – *16S*	90533..92085	ribosomal RNA-16S
rrnJ – *23S*	92251..95178	ribosomal RNA-23S
trnJ – *Ala*	96143..96218	transfert RNA-Ala
rrnW – *16S*	96389..97941	ribosomal RNA-16S
rrnW – *23S*	98107..101034	ribosomal RNA-23S
rrnW – *5S*	101090..101205	ribosomal RNA-5S
sigH	116597..117250	facteur stationnaire-phase sigma
rplQ	149951..150310	protéine ribosomale L17 (BL15); inconnu, similaire to ABC
kbaA	159181..159774	signalant le passage à la sporulation
rrnI – *16S*	160892..162443	ribosomal RNA-16S
rrnI – *23S*	162609..165534	ribosomal RNA-23S
rrnI – *5S*	165590..165705	ribosomal RNA-5S
trnI – *Ala*	166252..166327	transfert RNA-Ala
rrnH – *16S*	166498..168051	ribosomal RNA-16S
rrnH – *23S*	168217..171140	ribosomal RNA-23S
rrnH – *5S*	171196..171307	ribosomal RNA-5S
rrnG – *16S*	171497..173046	ribosomal RNA-16S
rrnG – *23S*	173213..176140	ribosomal RNA-23S
rrnG – *5S*	176196..176307	ribosomal RNA-5S
ybaR	177082..178515	inconnu
trnSL – *Tyr*1	194447..194531	transfert RNA-Tyr
sigW	194838..195398	RNA polymerase;facteur sigma de type ECF

FIG. 7.3: *Description de certains gènes de la bactérie B. subtilis.*

Dans les tableaux 7.4 et 7.1, nous donnons respectivement les estimations des lois stationnaires et des probabilités de transitions sur chaque segment. Les probabilités permettant de dégager les caractéristiques particulières des régions sont représentées en gras dans les deux tableaux. De plus, la localisation précise des gènes et leurs fonctions sont données dans le tableau 7.3.

π_{I_1}	A	C	G	T
	0.3220	**0.1997**	0.2214	0.2570

π_{I_2}	A	C	G	T
	0.2580	0.2284	**0.3078**	**0.2058**

π_{I_3}	A	C	G	T
	0.3029	**0.1871**	0.2310	0.2790

π_{I_4}	A	C	G	T
	0.2628	0.2270	**0.3039**	**0.2063**

π_{I_5}	A	C	G	T
	0.3134	**0.1866**	0.2398	0.2601

π_{I_6}	A	C	G	T
	0.2602	0.2256	**0.3031**	**0.2111**

π_{I_7}	A	C	G	T
	0.3162	**0.1813**	0.2455	0.2570

π_{I_8}	A	C	G	T
	0.3083	**0.1966**	0.2273	0.2678

π_{I_9}	A	C	G	T
	0.2926	**0.1868**	0.2307	**0.2899**

$\pi_{I_{10}}$	A	C	G	T
	0.2637	0.2242	**0.3020**	**0.2101**

$\pi_{I_{11}}$	A	C	G	T
	0.2478	**0.2082**	0.2345	**0.3094**

$\pi_{I_{12}}$	A	C	G	T
	0.3181	**0.1971**	0.2446	0.2402

FIG. 7.4: *Estimation des lois stationnaires sur chacune des régions détectées de la bactérie B. subtilis entre les instants* 1 *et* 200000.

p_{I_1}	A	C	G	T
A	**0.3691**	**0.1793**	0.2001	0.2515
C	**0.3364**	**0.1819**	0.2411	0.2406
G	**0.3620**	0.2147	0.2077	0.2156
T	0.2169	0.2261	0.2445	**0.3125**

p_{I_2}	A	C	G	T
A	0.2832	0.2115	**0.3366**	**0.1687**
C	0.2252	0.2735	0.2728	0.2235
G	0.2745	0.2079	**0.3007**	0.2169
T	0.2383	0.2306	**0.3158**	0.2153

p_{I_3}	A	C	G	T
A	**0.3878**	**0.1379**	0.2082	0.2661
C	**0.3056**	**0.1707**	0.2318	0.2918
G	**0.3093**	0.2548	0.2104	0.2255
T	0.2036	0.1953	0.2723	**0.3288**

p_{I_4}	A	C	G	T
A	0.2929	0.2078	**0.3314**	**0.1679**
C	0.2246	0.2767	0.2708	0.2279
G	0.2758	0.2067	**0.3034**	0.2142
T	0.2479	0.2257	**0.3062**	0.2202

p_{I_5}	A	C	G	T
A	**0.3708**	**0.1520**	0.2203	0.2570
C	**0.3301**	**0.1656**	0.2481	0.2561
G	**0.3489**	0.2359	0.2202	**0.1950**
T	0.1997	0.1981	0.2754	**0.3268**

p_{I_6}	A	C	G	T
A	0.2893	0.2066	**0.3291**	**0.1750**
C	0.2293	0.2716	0.2745	0.2247
G	0.2739	0.2042	**0.3020**	0.2199
T	0.2378	0.2303	**0.3034**	0.2285

p_{I_7}	A	C	G	T
A	**0.3584**	**0.1484**	0.2409	0.2524
C	**0.3458**	**0.1575**	0.2452	0.2515
G	**0.3542**	0.2228	0.2323	**0.1908**
T	0.2072	0.1987	0.2643	**0.3298**

p_{I_8}	A	C	G	T
A	**0.3947**	**0.1911**	**0.1922**	0.2220
C	0.2858	**0.1547**	0.2593	**0.3002**
G	**0.3094**	0.2162	0.2020	0.2723
T	0.2243	0.2169	0.2656	0.2931

p_{I_9}	A	C	G	T
A	**0.3495**	**0.1497**	0.2218	0.2790
C	**0.3180**	**0.1742**	0.2167	0.2911
G	**0.3104**	0.2337	0.2386	0.2173
T	0.2046	0.1948	0.2425	**0.3580**

$p_{I_{10}}$	A	C	G	T
A	0.2943	0.2062	**0.3261**	**0.1734**
C	0.2267	0.2717	0.2766	0.2250
G	0.2786	0.2034	**0.2996**	0.2184
T	0.2434	0.2255	**0.3025**	0.2286

$p_{I_{11}}$	A	C	G	T
A	**0.3205**	**0.1406**	0.2179	**0.3210**
C	0.2624	0.2217	0.2339	0.2820
G	0.2740	0.2740	0.2114	0.2406
T	**0.1599**	0.2033	0.2659	**0.3708**

$p_{I_{12}}$	A	C	G	T
A	**0.3774**	**0.1690**	0.2107	0.2429
C	**0.3256**	**0.1748**	0.2699	0.2296
G	**0.3455**	0.2471	0.2200	**0.1875**
T	0.2057	0.2009	0.2939	**0.2994**

TAB. 7.1: *Estimation des probabilités de transition sur chacune des régions détectées de la bactérie B. subtilis entre les instants* 1 *et* 200000.

Nous remarquons une similitude entre certaines des régions détectées. Deux groupes caractéristiques se dégagent nettement :

- Les régions 2, 4, 6 et 10 présentent une même composition des bases. Tout d'abord, elles sont très riches en G et pauvres en T. L'estimation des transitions des bases permet d'apporter une caractérisation supplémentaire : nous pouvons observer une richesse en GG, TG et AG et une pauvreté en AT. Le tableau 7.3 montre que les gènes appartenant à ces régions sont des gènes spécifiant des *ARN ribosomiques* et des *ARN de transfert.*

- Les régions 1, 3, 5, 7, 8, 9 et 12 sont très riches en A et pauvres en C. Plus précisément, elles sont caractérisées par une grande richesse en AA, GA, CA et TT et une pauvreté en CC et AC. Ceci est particulièrement net pour les régions 1, 5 et 7.

Seule la région 11 se détache de ces deux compositions en bases. Cette région est composée de gènes de fonctions différentes mais dont le sens de transcription va de droite à gauche (sauf 2, cf Figure 7.2), sens opposé au reste des gènes présents dans la séquence. Ce qui peut expliquer sa composition différente.

En conclusion, la méthode détecte bien les régions composées de gènes codants pour l'*ARN ribosomiques* (ARNr, présent dans les ribosomes) entrecoupés de gènes codants pour

l'*ARN transfert* (ARNt). Avec l'ARN messager (ARNm), ils sont les trois type fondamentaux d'ARN. Ils interviennent dans la phase de traduction de la synthèse des protéines, avec des fonctions différentes. Il serait bien sûr intéressant de confronter ces résultats aux avis des biologistes.

Dans le but d'affiner la détection, nous réitérons la procédure sur chacune des régions détectées. Seule une rupture se dégage de cette étude : la région 9 se sépare en deux au niveau de l'instant 158504 (proche de la rupture retenue par l'algorithme CART et non retenue par la recherche exhaustive). Néanmoins, deux résultats ont retenus notre attention. Ils sont exposés dans la section suivante.

7.2.2 Application sur deux régions détectées

Nous avons décidé de "zoomer" sur deux des régions détectées : la région entre les instants 90505 et 101200, et celle entre les instants 160860 et 176325. Pour ces deux régions, la partition sélectionnée est de dimension 1, c'est-à-dire sans ruptures. Cependant, nous observons un même phénomène : dans la suite des dimensions $(D_i)_{i=1,\dots,K}$, un saut important de dimensions est repéré juste après la dimension 1. Par exemple, pour la première région, la suite des dimensions passe de la dimension 8 à la dimension 1. En considérant la partition la plus proche au sens de α (*i.e.* la partition associé au α_i le plus proche de α), nous sélectionnons aussi la partition de dimension 8. Pour la seconde région considérée, nous sélectionnons de la même façon, la partition de dimension 11. Les partitions sont représentées respectivement sur les Figures 7.5 et 7.6.

Ces Figures montrent que les partitions sélectionnées mènent à une délimitation très nette des gènes codants pour des ARN ribosomiques.

A titre d'indication, nous donnons l'estimation des lois stationnaires et des probabilités de transition sur chacune des régions détectées entre les instants 160860 et 176325 respectivement dans les tableaux 7.7 et 7.2.

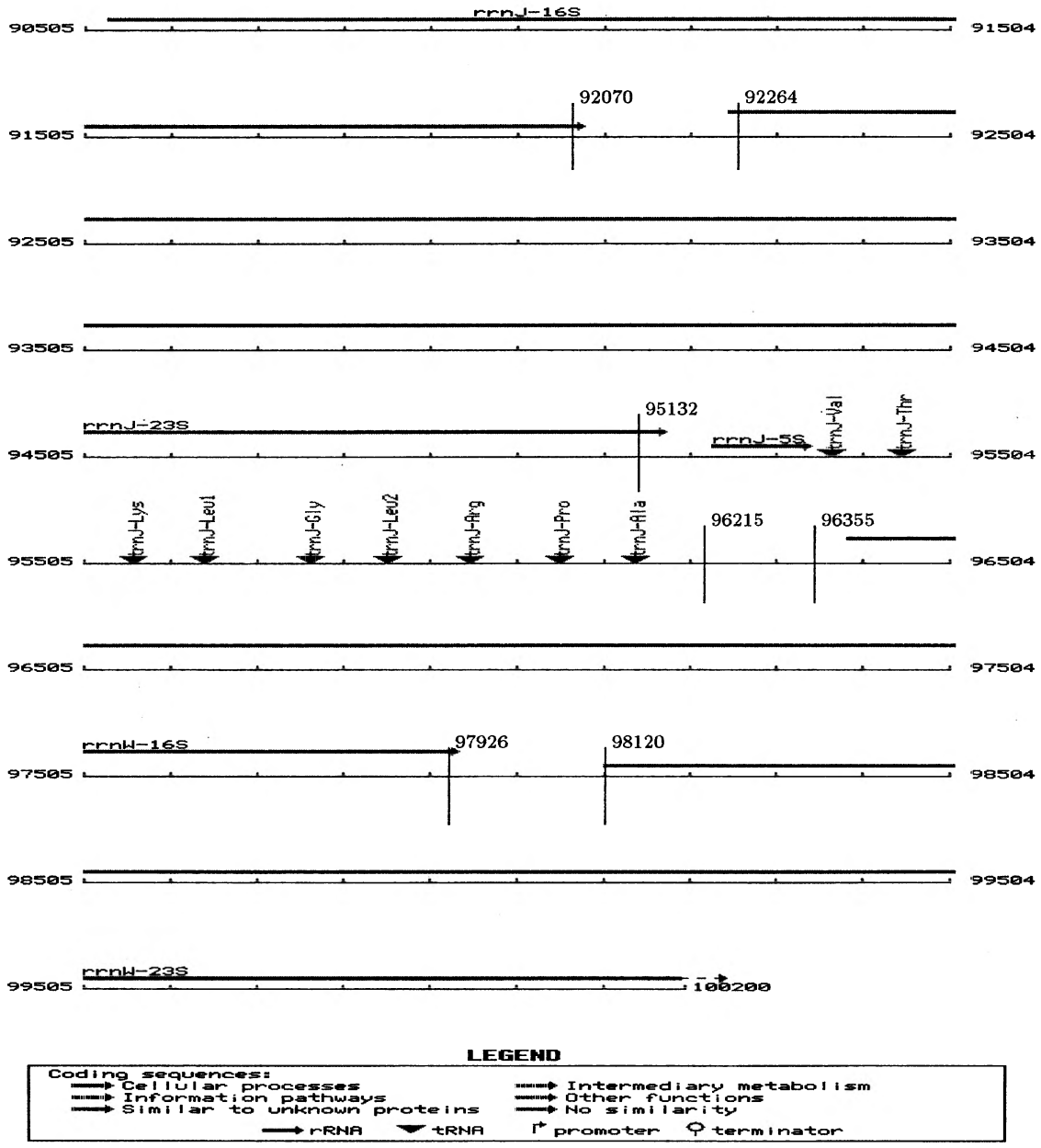


FIG. 7.5: *Régions homogènes détectées sur la bactérie B. subtilis entre les instants* 90505 *et* 101200. *Les flèches indiquent le sens de transcription des gènes.*

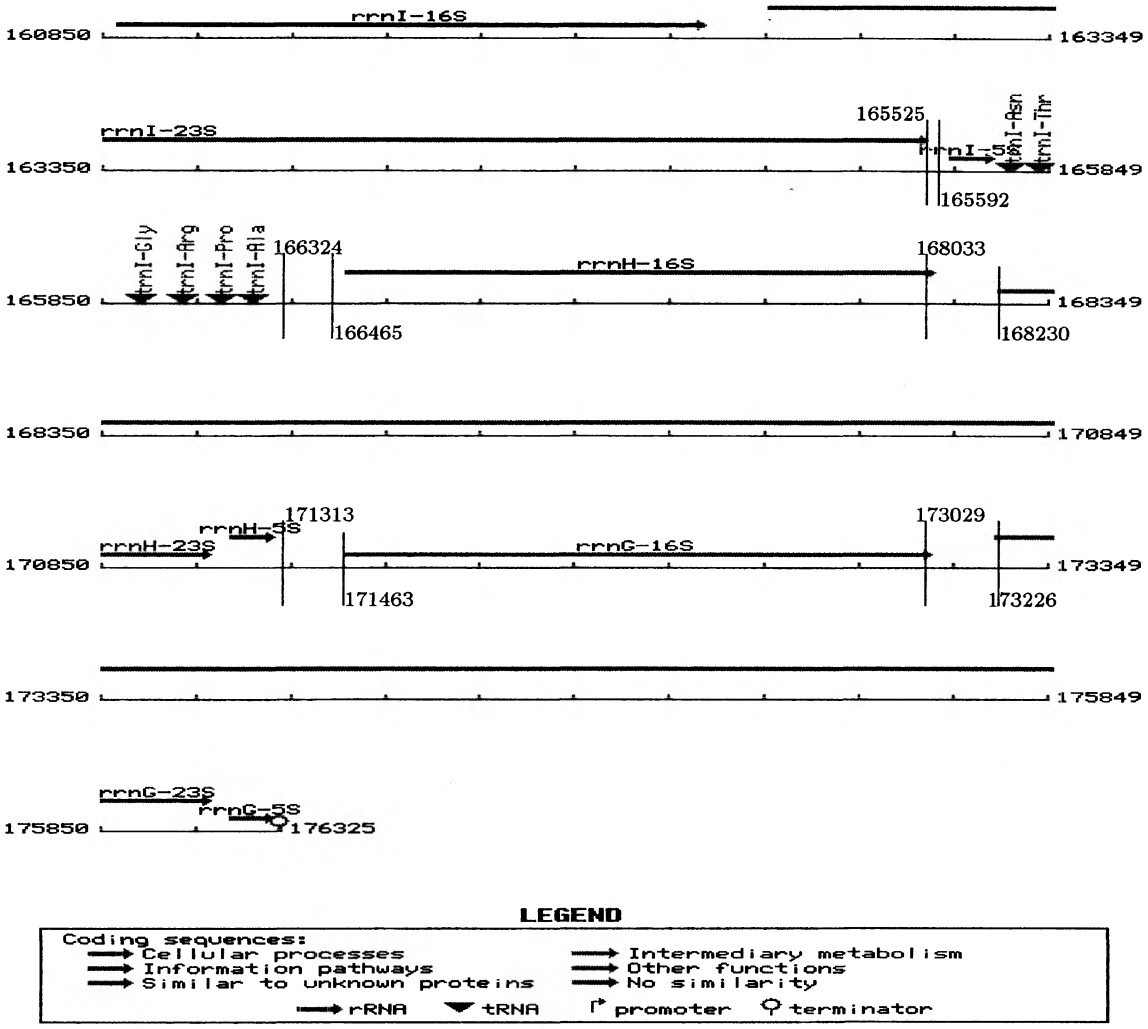


FIG. 7.6: *Régions homogènes détectées sur la bactérie B. subtilis entre les instants* 160860 *et* 176325. *Les flèches indiquent le sens de transcription des gènes.*

π_{I_1}	A	C	G	T
	0.2624	0.2255	0.3091	0.2030

π_{I_2}	A	C	G	T
	0.2687	0.1642	0.1343	0.4328

π_{I_3}	A	C	G	T
	0.2281	0.2281	0.2773	0.2664

π_{I_4}	A	C	G	T
	0.4397	0.1277	0.1418	0.2908

π_{I_5}	A	C	G	T
	0.2487	0.2347	0.3157	0.2009

π_{I_6}	A	C	G	T
	0.3249	0.1726	0.1675	0.3350

π_{I_7}	A	C	G	T
	0.2644	0.2267	0.3104	0.1985

π_{I_8}	A	C	G	T
	0.4200	0.1200	0.1400	0.3200

π_{I_9}	A	C	G	T
	0.2503	0.2324	0.3155	0.2018

$\pi_{I_{10}}$	A	C	G	T
	0.3249	0.1726	0.1675	0.3350

$\pi_{I_{11}}$	A	C	G	T
	0.2643	0.2262	0.3111	0.1985

FIG. 7.7: *Estimation des lois stationnaires sur chacune des régions détectées de la bactérie B. subtilis entre les instants* 160860 *et* 176325.

p_{I_1}	A	C	G	T
A	0.2911	0.2069	0.3328	0.1693
C	0.2224	0.2728	0.2804	0.2443
G	0.2788	0.2039	0.3037	0.2136
T	0.2450	0.2302	0.3178	0.2070

p_{I_2}	A	C	G	T
A	0.2222	0.2222	0.1667	0.3889
C	0.4545	0.0909	0	0.4545
G	0.5556	0	0	0.4444
T	0.1429	0.1786	0.2143	0.4643

p_{I_3}	A	C	G	T
A	0.2169	0.1386	0.3735	0.2711
C	0.2335	0.2635	0.2695	0.2335
G	0.2217	0.2512	0.3054	0.2217
T	0.2410	0.2513	0.1692	0.3385

p_{I_4}	A	C	G	T
A	0.5738	0.1803	0.1311	0.1148
C	0.4444	0.1111	0.1111	0.3333
G	0.4000	0.1000	0.1000	0.4000
T	0.2683	0.0488	0.1951	0.4878

p_{I_5}	A	C	G	T
A	0.2795	0.2615	0.3077	0.1513
C	0.2418	0.2364	0.3071	0.2147
G	0.2470	0.2206	0.3057	0.2267
T	0.2222	0.2222	0.3524	2032

p_{I_6}	A	C	G	T
A	0.3175	0.2063	0.2063	0.2698
C	0.3235	0.1765	0.1471	0.3529
G	0.4242	0	0.2121	0.2267
T	0.2879	0.2121	0.1212	0.3788

p_{I_7}	A	C	G	T
A	0.2924	0.1892	0.3489	0.1695
C	0.2074	0.3047	0.2704	0.2175
G	0.2926	0.1964	0.3030	0.2079
T	0.2484	0.2353	0.3154	0.2010

p_{I_8}	A	C	G	T
A	0.5323	0.1935	0.1290	0.1452
C	0.5000	0.0556	0.1111	0.3333
G	0.3810	1429	0.0476	0.4286
T	0.2708	0.0417	0.2083	0.4792

p_{I_9}	A	C	G	T
A	0.2781	0.2602	0.3087	0.1531
C	0.2473	0.2280	0.3077	0.2170
G	0.2475	0.2211	0.3043	0.2272
T	0.2247	0.2215	0.3513	0.2025

$p_{I_{10}}$	A	C	G	T
A	0.3175	0.2063	0.2063	0.2698
C	0.3235	0.1765	0.1471	0.3529
G	0.4242	0	0.2121	0.3636
T	0.2879	0.2121	0.1212	0.3788

$p_{I_{11}}$	A	C	G	T
A	0.2930	0.1856	0.3529	0.1685
C	0.2068	0.3024	0.2710	0.2197
G	0.2918	0.1994	0.3022	0.2066
T	0.2488	0.2358	0.3138	0.2016

TAB. 7.2: *Estimation des probabilités de transition sur chacune des régions détectées de la bactérie B. subtilis entre les instants* 160860 *et* 176325.

7.2.3 Comparaison avec le modèle indépendant

Nous avons appliqué l'algorithme hybride en supposant l'indépendance des bases. Dans ce cas, le contraste a une expression plus simple et l'algorithme est bien plus simple à mettre en oeuvre. La partition sélectionnée est de dimension 11. Les ruptures associées sont : 9713, 14834, 30295, 35366, 90538, 101234, 158486, 160860, 176133 et 194695. La localisation des ruptures est proche de celle obtenue dans le modèle markovien. Seule la rupture 158486 non retenue dans le modèle markovien est ici conservé. Par contre, la rupture 117104 détectée dans le modèle markovien n'apparaît pas ici.

En conclusion, le modèle markovien n'apporte pas d'informations supplémentaires sur la séquence étudiée par rapport au modèle indépendant. Deux résultats intéressants se dégagent de cette étude :

1. le modèle markovien permet de détecter des ruptures dans la loi des $(Y_t)_{t=1,\dots,n}$,
2. sur cet exemple, il n'existe pas de ruptures dans les probabilités de transition.

Il serait intéressant par la suite d'augmenter l'ordre de la chaîne de Markov, et plus particulièrement de considérer le modèle où la suite des variables Y_t sur chaque segment est une chaîne de Markov d'ordre 2. Les biologistes sont en effet intéressés par ce type de

modèle qui prend en compte la dépendance de 3 bases successives : il pourraît permettre de révéler des parties codantes, par exemple les codons qui sont des trinucléotides traduits en acides aminés pour constituer les protéines.

7.3 Recherche des régions homogènes du bactériophage Lambda

Dans cette section, nous appliquons l'algorithme hybride sur le bactériophage lambda complet qui comporte 48502 *bp*. Les données sont disponibles sur le site http://www.infobiogen.fr. Nous nous plaçons tout d'abord dans le modèle markovien.

7.3.1 Résultats pour le modèle markovien

Nous effectuons la même étude que précédemment :

1. La partition de dimension 8 est sélectionnée. Les ruptures associées sont respectivement : 20018, 20652, 21633, 22605, 27829, 37955 et 46526.
2. L'arbre de dimension $v \times 8 = 32$ est extrait de l'arbre maximal.
3. L'algorithme de recherche exhaustive est appliqué sur la nouvelle grille. Le graphe de $(f_n(D), \gamma_n(\hat{s}_D))$ pour $D = 1, \ldots, 32$ et la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ sont respectivement représentés à gauche et à droite sur la Figure 7.8. La partition finale est de dimension 7. La rupture 21633 a été retirée de la partition précédente. La partition finale est représentée en Figure 7.9 et une partie de la carte génétique du phage lambda y est ajoutée.

La première caractéristique biologique qui se dégage des régions détectées est que ces régions contiennent uniquement des gènes ayant un même sens de transcription. Dans les tableaux 7.10 et 7.11, nous donnons respectivement les estimations des lois stationnaires et des probabilités de transition sur chacune des régions détectées. Les probabilités permettant de dégager les caractéristiques particulières des régions obtenues sont indiquées en gras sur ces tableaux. Les régions présentent des compositions en bases différentes. Par exemple, la région 1 est riche en CG et TG et pauvre en TA, CT et GT, la région 5 est riche en TT et AT et pauvre en TA et AC. Néanmoins, certains profils particuliers, ainsi que certaines similitudes se dégagent :

- Deux régions présentent une composition en bases extrêmement riche : la région 2 et 4. Toutes les probabilités de transition sont à prendre en considération.
  La région 4 présente une grande richesse en iA et iT et une pauvreté en iC et iG quelque soit $i \in \{A, C, G, T\}$. La région 7 présente en grande majorité ce même profil.
  La région 2 est riche en iA, iG, jC et jG et pauvre en iC, iT, jA et jT pour $i \in \{A, C\}$ et $j \in \{G, T\}$.

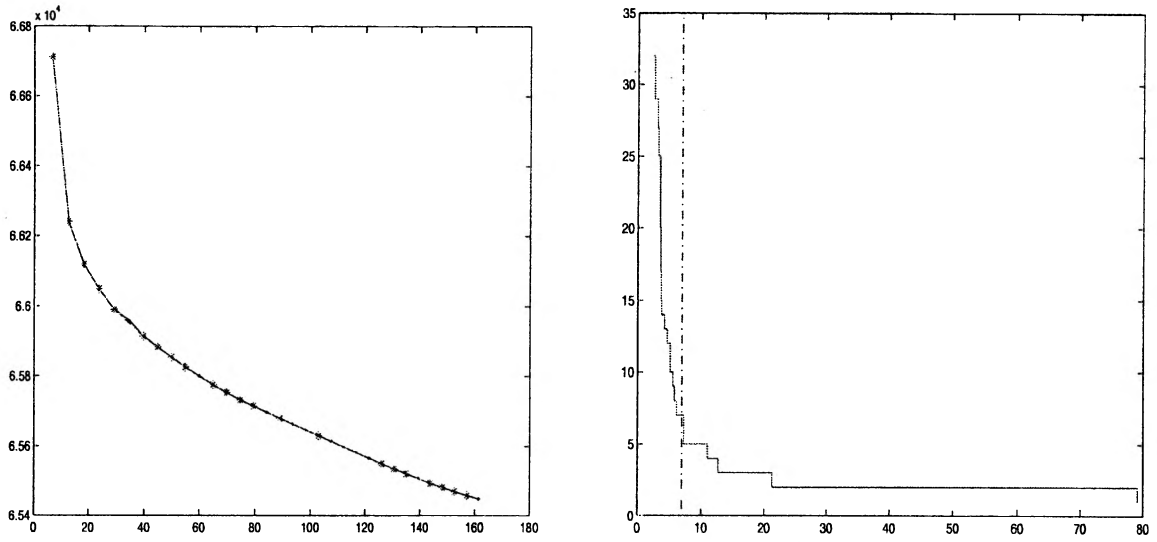


FIG. 7.8: $(f_n(D), \gamma_n(\hat{s}_D))$ *pour* $D = 1, \ldots, 32$ *(à gauche) et* $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$ *(à droite) où le symbole -. correspond à* $2\hat{\alpha}$.

- Les régions 3 et 6 sont caractérisées par une richesse en AA, CA et TG mais ne présentent pas la même pauvreté de bases.

D'après le graphe de la fonction $\sum_{i=1}^{K} D_i \mathbb{1}_{[\alpha_i, \alpha_{i+1}[}$, et la valeur de $2\hat{\alpha}$, il peut être intéressant de considérer les deux partitions les plus proches au sens de α : celles de dimension 5 et 8. Pour la partition de dimension 8, l'instant de rupture 174 est ajoutée.

Des similitudes existent avec les résultats obtenus par Muri [52] sur cette séquence dans le modèle $M_1 - M_1$ (les bases sont générées selon une chaîne de Markov d'ordre 1) où l'espace d'états de la chaîne de Markov cachée est de dimension 3. Par exemple, la région 1 et la région 6 se dégagent des deux études.

7.3.2 Comparaison avec le modèle indépendant

Nous avons appliqué l'algorithme hybride en supposant l'indépendance des bases. La partition sélectionnée est de dimension 5. Les ruptures associées sont : 22547, 27830, 38005 et 46529. La partition la plus proche au sens de α est la partition de dimensions 7. Les deux ruptures supplémentaires sont 20011 et 20920. Nous retrouvons des ruptures proches de celles sélectionnées dans le modèle markovien. Nous concluons au mêmes résultats que pour la bactérie B.subtilis.

Nous observons une concordance avec les résultats obtenus par Braun *et. al* [11] sous l'hypothèse d'indépendance. Dans le papier de Braun et Muller [12], les différentes méthodes de segmentation présentées sont appliquées sur le bactériophage Lambda.

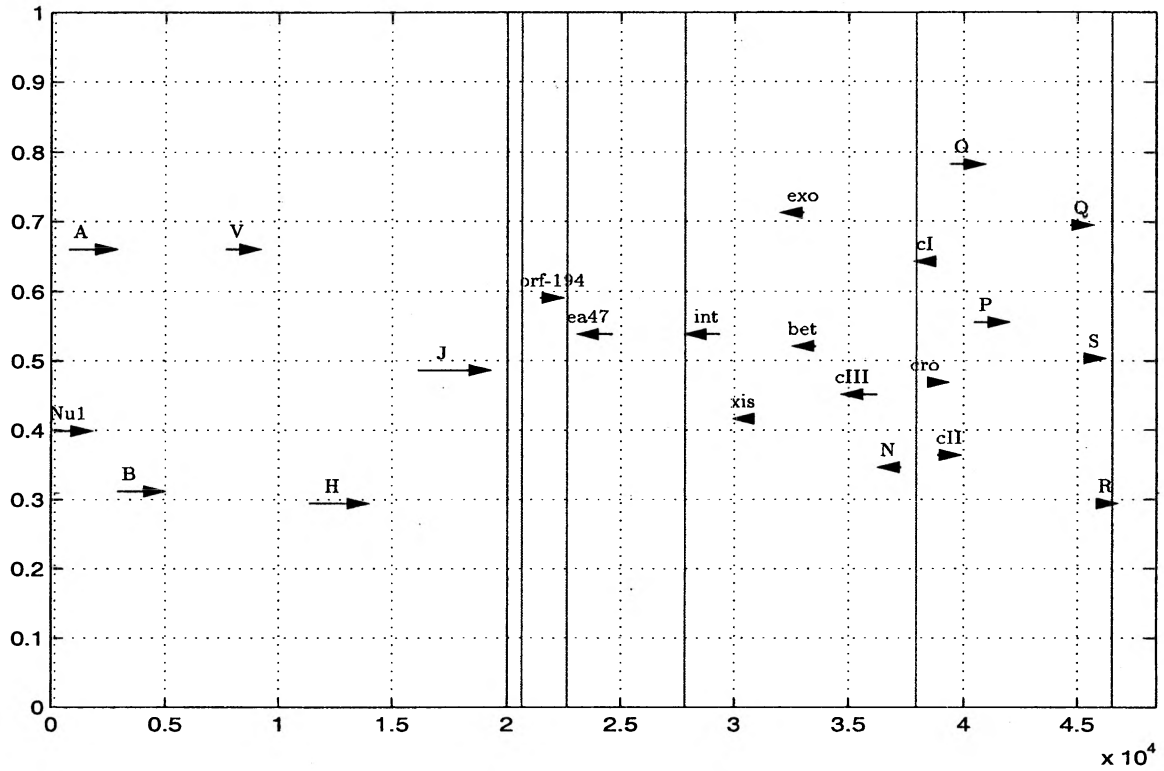


FIG. 7.9: *Régions homogènes détectées sur le phage lambda et comparaison avec la carte génétique du phage lambda. Les flèches indiquent le sens de transcription des gènes.*

π_{I_1}	A	C	G	T
	0.2268	0.2520	**0.3169**	**0.2043**

π_{I_2}	A	C	G	T
	0.2697	**0.3060**	**0.3123**	**0.1120**

π_{I_3}	A	C	G	T
	0.2698	0.2386	0.2709	0.2207

π_{I_4}	A	C	G	T
	0.2919	**0.1817**	**0.1761**	**0.3503**

π_{I_5}	A	C	G	T
	0.2474	0.2369	**0.2144**	**0.3013**

π_{I_6}	A	C	G	T
	0.2956	0.2273	0.2598	**0.2172**

π_{I_7}	A	C	G	T
	0.2697	**0.1817**	0.2176	**0.3310**

FIG. 7.10: *Estimation des lois stationnaires sur chacune des régions détectées du phage lambda complet.*

p_{I_1}	A	C	G	T
A	0.2603	0.2484	0.2594	0.2319
C	0.2462	0.2381	**0.3330**	**0.1828**
G	0.2403	0.2888	0.2806	**0.1903**
T	**0.1451**	0.2162	**0.4168**	0.2219

p_{I_2}	A	C	G	T
A	0.3706	0.1882	0.3294	0.1118
C	0.3505	0.1753	0.3557	0.1186
G	0.1919	0.4545	0.2273	0.1263
T	0.0141	0.5352	0.3944	0.0563

p_{I_3}	A	C	G	T
A	**0.3207**	0.2524	**0.1935**	0.2334
C	**0.3283**	0.2103	0.2768	**0.1845**
G	0.2405	0.2784	0.2784	0.2027
T	**0.1810**	0.2042	**0.3503**	0.2645

p_{I_4}	A	C	G	T
A	0.3370	0.1495	0.1692	0.3443
C	0.3298	0.1728	0.1465	0.3509
G	0.2837	0.2272	0.1772	0.3120
T	0.2389	0.1903	0.1963	0.3745

p_{I_5}	A	C	G	T
A	0.2883	**0.1773**	0.1969	**0.3375**
C	0.2980	0.2301	0.1972	0.2747
G	0.2464	0.2893	0.2004	0.2639
T	**0.1747**	0.2540	0.2520	**0.3192**

p_{I_6}	A	C	G	T
A	**0.3453**	0.2044	0.2174	0.2328
C	**0.3154**	**0.1931**	0.2784	0.2131
G	0.2910	0.2797	0.2375	**0.1917**
T	0.2132	0.2309	**0.3249**	0.2309

p_{I_7}	A	C	G	T
A	**0.3152**	**0.1689**	0.1764	**0.3396**
C	0.3036	0.1978	0.2256	0.2730
G	0.2867	0.1958	0.1841	**0.3333**
T	0.2034	**0.1728**	0.2691	**0.3547**

FIG. 7.11: *Estimation des probabilités de transition sur chacune des régions détectées du phage lambda complet.*

Bibliographie

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

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

[3] AVERY, P., AND HENDERSON9, D. Detecting a changed segment in DNA sequences. *J. Roy. Statist. Soc. Ser. C 48*, 4 (1999), 489–503.

[4] BASSEVILLE, M., AND NIKIFOROV, N. *The Detection of abrupt changes - Theory and applications.* Prentice-Hall: Information and System sciences series, 1993.

[5] BELLALAH, M., AND LAVIELLE, M. A simple decomposition of empirical distributions and its applications in asset pricing. *(submitted)* (1997).

[6] BESAG, J., GREEN, P., HIDGON, D., AND MENGERSEN, K. Baesyian computation and stochastic systems. *Statistical Science 10* (1995), 3–66.

[7] BIRGÉ, L., AND MASSART, P. Gaussian model selection. *J. Eur. Math. Soc. 3* (2001), 203–268.

[8] BIRGÉ, L., AND MASSART, P. A generalized C_p criterion for Gaussian model selection. Tech. rep., Publication Université Paris-VI, 2001.

[9] BIRGÉ, L., AND ROZENHOLC, Y. How many bins should be put in a regular histogram. Tech. rep., Publication Université Paris-VI, 1999.

[10] BISCAY, R., LAVIELLE, M., GONZÁLEZ, A., CLARK, I., AND VALDÉS, P. Maximum a posteriori estimation of change points in the EEG. *Int. J. of Bio-Medical Computing 38* (1995), 189–196.

[11] BRAUN, J. V., BRAUN, R. K., AND MÜLLER, H.-G. Multiple changepoint fitting via quasilikelihood, with application to DNA sequence segmentation. *Statistical Science 87*, 2 (2000), 142–162.

[12] BRAUN, J. V., AND MÜLLER, H.-G. Statistical methods for DNA sequence segmentation. *Biometrika 13*, 2 (1998), 301–314.

[13] BREIMAN, L., FRIEDMAN, J. H., OLSHEN, R. A., AND STONE, C. J. *Classification And Regression Trees.* Chapman & Hall, 1984.

[14] BRODSKY, B., AND DARKHOVSKY, B. *Nonparametric methods in change-point problems.* Kluwer Academic Publishers, the Netherlands, 1993.

[15] CAPPÉ, O., DOUCET, A., LAVIELLE, M., AND MOULINES, E. Methods for blind maximum-likelihood linear system identification. *Signal Processing 73* (1999), 3–25.

[16] CARTER, R. L., AND BLIGHT, B. J. N. A Bayesian change-point problem with an application to the prediction and detection of ovulation in women. *Biometrics 37*, 4 (1981), 743–751.

[17] CASELLA, G., AND ROBERT, C. Rao-blackwellisation of sampling schemes. *Biometrika 83* (1996), 81–94.

[18] CASTELLAN, G. Modified akaike's criterion for histogram density estimation. Tech. Rep. 61, Université Paris XI, 1999.

[19] CHONG, T. T.-L. Estimating the locations and number of change points by the sample-splitting method. *Statist. Papers 42*, 1 (2001), 53–79.

[20] CHURCHILL, G. Stochastic Models for Heterogeneous DNA Sequences. *Bulletin of Mathematical Biology 51*, 1 (1989), 79–94.

[21] CSORGO, M., AND HORVÀTH, L. Limit theorems in change-point analysis. Tech. rep., U.K.: Wiley, 1997.

[22] DELYON, B., LAVIELLE, M., AND MOULINES, E. Convergence of a stochastic approximation version of the EM algorithm. *The Annals of Stat. 27*, 1 (1999), 94–128.

[23] DEMPSTER, A., LAIRD, N., AND RUBIN, D. Maximum-likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B 39* (1977), 1–38.

[24] DUFLO, M. *Algorithmes Stochastiques.* SMAI, Springer, 1996.

[25] ENGL, H., AND GREVER, W. Using the L-curve for determining optimal regularization parameters. *Numer. Math. 69*, 1 (1994), 25–31.

[26] FU, Y.-X., AND CURNOW, R. N. Maximum likelihood estimation of multiple change points. *Biometrika 77*, 3 (1990), 563–573.

[27] GEMAN, S., AND GEMAN, D. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Trans. on Pattern Anal. Machine Intell. 6* (1984), 721–741.

[28] GEY, S., AND NEDELEC, E. Model selection for CART regression trees. Tech. Rep. 56, Université Paris XI, 2001.

[29] GHORBANZADEH, D. Un test de détection de rupture de la moyenne dans un modèle gaussien. *Rev. Statist. Appl. 43*, 2 (1995), 67–76.

[30] GHORBANZADEH, D. Un test de détection de rupture de la moyenne dans un modèle gaussien. *Rev. Statist. Appl. 43*, 2 (1995), 67–76.

[31] GREEN, P. Reversible jump mcmc computation and bayesian model determination. *Biometrika 82* (1995), 711–732.

[32] HALL, P., KAY, J., AND TITTERINGTON, D. Asymptotically optimal difference-based estimation of variance in nonparametric regression. *Biometrika 77* (1990), 521–8.

[33] HANKE, M. Limitations of the L-curve method in ill-posed problems. *BIT 36*, 2 (1996), 287–301.

[34] HANSEN, P. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Rev. 34*, 4 (1992), 561–580.

[35] HANSEN, P., AND O'LEARY, D. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput. 14*, 6 (1993), 1487–1503.

[36] J.L. OLIVIER, R. ROMÁN-ROLDÁN, J. P., AND BERNAOLA-GALVÁN, P. Segment: identifying compositional domains in DNA sequences. *Bioinformatics 15*, 12 (1999), 974–979.

[37] KAY, S. M. *Fundamentals of statistical signal processing - Detection theory*, vol. II. Prentice Hall signal processing series, 1998.

[38] LAVIELLE, M. A stochastic procedure for parametric and non-parametric estimation in the case of incomplete data. *Signal Processing 42* (1995), 3–17.

[39] LAVIELLE, M. Optimal segmentation of random processes. *IEEE Trans. on Signal Processing 46*, 5 (1998), 1365–1373.

[40] LAVIELLE, M. Detection of multiple changes in a sequence of dependent variables. *Stoch. Proc. and Appl. 83* (1999), 79–102.

[41] LAVIELLE, M., AND LEBARBIER, E. An application of MCMC methods for the multiple change-points problem. *Signal processing 81* (2001), 39–53.

[42] LAVIELLE, M., AND MOULINES, E. Least Squares estimation of an unknown number of shifts in a time series. *Jour. of Time Series Anal. 21* (2000), 33–59.

[43] LEBARBIER, E. *Quelques approches pour la détection de ruptures à horizon fini.* PhD thesis, Université Paris XI, 2002.

[44] LEE, C.-B. Estimating the number of change points in exponential families distributions. *Scand. J. Statist. 24*, 2 (1997), 201–210.

[45] LETUÉ, F. *Modèle de Cox: estimation par sélection de modèle et modèle de chocs bivarié.* PhD thesis, Université de Paris-Sud, 2000.

[46] LEZAUD, P. Chernoff-type bound for finite Markov chains. *Ann. Appl. Probab. 8*, 3 (1998), 849–867.

[47] MALLOWS, C. Some comments on Cp. *Technometrics 15* (1974), 661–675.

[48] MASSART, P. Some Applications of Concentration Inequalities to Statistics. *Annales de la Faculté des Sciences de Toulouse IX*, 2 (2000), 245–303.

[49] MÉTIVIER, M., AND PRIOURET, P. Théorèmes de convergence presque sure pour une classe d'algorithmes stochastiques à pas décroissant. *Probab. Theory Related Fields 74*, 3 (1987), 403–428.

[50] MEYN, S. P., AND TWEEDIE, R. L. *Markov chains and stochastic stability.* Springer-Verlag London Ltd., London, 1993.

[51] MIAO, B. Q., AND ZHAO, L. C. On detection of change points when the number is unknown. *Chinese J. Appl. Probab. Statist. 9*, 2 (1993), 138–145.

[52] MURI, F. *Comparaison d'algorithmes d'identification de chaînes de Markov cachées et application à la détection de régions homogènes dans les séquences d'ADN.* PhD thesis, Université René Descartes, Paris V, 1997.

[53] NORMAND, S.-L. T., AND DOKSUM, K. Empirical Bayes procedures for a change point problem with application to HIV/AIDS data. In *Empirical Bayes and likelihood inference (Montreal, QC, 1997).* Springer, New York, 2001, pp. 67–79.

[54] PICARD, D. Testing and estimating change points in time series. *J. Applied Prob. 17* (1985), 841–867.

[55] RAO, C. R. *Linear statistical inference and its applications*, second ed. John Wiley & Sons, New York-London-Sydney, 1973. Wiley Series in Probability and Mathematical Statistics.

[56] R.J. BOYS, D. H., AND WILKINSON, D. Detecting homogeneous segments in DNA sequences by using hidden Markov models. *J. Roy. Statist. Soc. Ser. C 48*, 4 (1999), 489–503.

[57] ROBERT, C. *Méthodes de Monte Carlo par Chaînes de Markov.* Statistique mathématique et Probabilité. Economica, 1996.

[58] ROBERT, C. P., Ed. *Discretization and MCMC convergence assessment.* Springer-Verlag, New York, 1998.

[59] SCHWARZ, G. Estimating the dimension of a model. *Ann. Stat. 6* (1978), 461–464.

[60] SMITH, A. F. M. Change-point problems: approaches and applications. In *Bayesian statistics (Valencia, 1979).* Univ. Press, Valencia, 1980, pp. 83–98.

[61] TIBSHIRANI, R., WALTHER, G., AND HASTIE, T. Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc. Ser. B Stat. Methodol. 63*, 2 (2001), 411–423.

[62] TOURNERET, J., AND CHABERT, M. Off-line detection and estimation of abrupt changes corrupted by multiplicative colored gaussian noise. Tech. rep., Proc. of ICASSP'97, Munich, April, 1997.

[63] TOURNERET, J., COULON, M., AND DOISY, M. Least Squares estimation of multiple abrupt changes contamined by multiplicative noise using mcmc. Tech. rep., Proc. of HOS'99, Caesarea, June, 1999.

[64] VOGEL, C. R. Non-convergence of the L-curve regularization parameter selection method. *Inverse Problems 12*, 4 (1996), 535–547.

[65] VOSTRIKOVA, L. Detecting "disorder" in multidimensional random processes. *Soviet. Math. Dokl. 24* (1981), 55–59.

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

[67] YASHCHIN, E. Change-point models in industrial applications. In *Proceedings of the Second World Congress of Nonlinear Analysts, Part 7 (Athens, 1996)* (1997), vol. 30, pp. 3997–4006.