Exploring the segmentation space for the assessment of multiple change-point models

Yann Guédon

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Exploring the segmentation space for the assessment of multiple change-point models

Yann Guédon

N° 6619

Août 2008

Thème BIO

ISSN 0249-6399 ISRN INRIA/RR--6619--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Exploring the segmentation space for the assessment of multiple change-point models

Yann Guédon* †

Thème BIO — Systèmes biologiques
Équipe-Projet Virtual Plants

Rapport de recherche n° 6619 — Août 2008 — 37 pages

Abstract: This paper addresses the retrospective or off-line multiple change-point detection problem. Methods for exploring the space of possible segmentations of a sequence for a fixed number of change points may be divided into two categories: (i) enumeration of segmentations, (ii) summary of the possible segmentations in change-point or segment profiles. Concerning the first category, a forward dynamic programming algorithm for computing the top L most probable segmentations and a forward-backward algorithm for sampling segmentations are derived. Concerning the second category, a forward-backward dynamic programming algorithm and a smoothing-type forward-backward algorithm for computing two types of change-point and segment profiles are derived. The proposed methods are mainly useful for exploring the space of possible segmentations for successive numbers of change points and provide a set of assessment tools for multiple change-point models. We show using examples that the proposed methods may help to compare alternative multiple change-point models (e.g. Gaussian model with piecewise constant variances or global variance), predict supplementary change points, highlight overestimation of the number of change points and summarize the uncertainty concerning the location of change points.

Key-words: dynamic programming algorithm, multiple change-point detection, plant structure analysis, smoothing algorithm.

* CIRAD, UMR Développement et Amélioration des Plantes
† INRIA, Équipe-projet Virtual Plants

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Exploration de l'espace des segmentations possibles pour l'évaluation de modèles de détection de ruptures multiples

Résumé : Le problème de détection *a posteriori* de ruptures multiples est étudié. Les méthodes permettant d'explorer l'espace des segmentations possibles d'une séquence pour un nombre de segments fixé peuvent être regroupées en deux catégories: (i) énumération des segmentations, (ii) résumé des segmentations possibles dans des profils de ruptures ou de segments. En ce qui concerne la première catégorie, un algorithme "avant" de programmation dynamique permettant de calculer les L segmentations les plus probables et un algorithme "avant-arrière" pour simuler des segmentations sont présentés. En ce qui concerne la seconde catégorie, un algorithme "avant-arrière" de programmation dynamique et un algorithme "avant-arrière" de type lissage permettant de calculer deux types de profils de ruptures et de segments sont présentés. Ces algorithmes sont surtout utiles pour explorer l'espace des segmentations pour des nombres de segments successifs et constituent un ensemble d'outils d'évaluation. Les méthodes proposées permettent de comparer des modèles alternatifs de détection de ruptures multiples (par exemple modèle gaussien de changement sur la moyenne ou de changement sur la moyenne et la variance) de prédire des ruptures supplémentaires, de mettre en évidence une sur-estimation du nombre de ruptures et de synthétiser l'incertitude sur la localisation des ruptures.

Mots-clés : algorithme de lissage, algorithme de programmation dynamique, analyse de la structure des plantes, détection de ruptures multiples.

1 Introduction

Multiple change-point models have been applied to diverse types of data including DNA sequences (Braun and Müller, 1998; Liu and Lawrence, 1999), protein sequences (Auger and Lawrence, 1989), comparative genomic hybridization (CGH) profiles (Picard *et al.*, 2006) and astronomical time series (Dobigeon *et al.*, 2007). With regard to the retrospective multiple change-point detection problem, much effort has been devoted in recent years to the inference of multiple change-point models and in particular to the selection of the optimal number of change points; see Lavielle (2005), Lebarbier (2005), Zhang and Siegmund (2007) and references therein. Here, we explore another research direction which focuses on exploring the space of possible segmentations for successive numbers of change points in an aim of model assessment. The knowledge of solely the most probable segmentation of a sequence (for a fixed number of change points) tells us nothing about the remainder of the segmentation space. Questions of interest are:

- Is the most probable segmentation most probable by a long way or are there other segmentations with near-optimal probability?
- Are these near-optimal segmentations very similar to the most probable segmentation or do they differ greatly?

Methods for exploring the space of possible segmentations may be divided into two categories:

- enumeration of possible segmentations,
- change-point or segment profiles i.e. possible segmentations summarized in a $J \times T$ array where J is the number of segments (hence, the number of change points is $J-1$) and T the length of the sequence.

The first category includes two methods, a deterministic and a randomized method. The top L most probable segmentations can be computed by a direct generalization of the classic forward dynamic programming algorithm (Auger and Lawrence, 1989) for computing the most probable segmentation. It is also possible to simulate segmentations using a dedicated forward-backward algorithm. One of the main outcome of this paper is to propose a forward-backward dynamic programming algorithm for computing a first type of change-point and segment profiles and a smoothing-type (with reference to state-space models) forward-backward algorithm for computing a second type of change-point and segment profiles. Both the smoothing-type forward-backward algorithm and its variant for sampling segmentations apply to a more restricted class of models than the dynamic programming algorithms. Models of this class are characterized by a separability property, i.e. there is no global parameter that depends on within-segment parameters.

For stochastic models involving a latent structure such as multiple change-point models or hidden Markov models, conditional independence properties can be used to design both dedicated dynamic programming algorithms and filtering/smoothing algorithms. In the case of multiple change-point models, dynamic programming algorithms were initially proposed in a non-Bayesian context (see Auger and Lawrence (1989), Hawkins (2001) and references therein)

while algorithms based on filtering recursions were always proposed up to now in a Bayesian context where the segment probabilities were obtained by integrating over the within-segment parameter space (see Liu and Lawrence (1999), Fearnhead (2006) and references therein). One outcome of this paper is to propose a smoothing-type algorithm and its variant for sampling segmentation in a non-Bayesian context where the segment probabilities are obtained by maximizing (not integrating) over the within-segment parameter space.

The proposed algorithms share a common structure with similar algorithms recently proposed for exploring the space of possible state sequences for hidden semi-Markov chains (Guédon, 2007). Conditional independence at change-point instants for multiple change-point models corresponds to conditional independence at state change instants for hidden semi-Markov chains. To illustrate this point, it is possible to draw an analogy between a multiple change-point model and a "left-right" hidden semi-Markov chain (i.e. composed of a succession of transient states and a final absorbing state) such that the succession of states is deterministic. Exploring the segmentation space means exploring a family of multiple change-point models for a fixed number of segments since each possible segmentation corresponds to a multiple change-point model. In this respect, hidden semi-Markov chains and multiple change-point models are very different in nature since in the case of hidden semi-Markov chains, the possible state sequences are computed on the basis of a previously estimated hidden semi-Markov chain. In the case of a family of multiple change-point models, the segment length is not explicitly modeled as in the case of a hidden semi-Markov chain with state occupancy distributions, but implicitly, as a consequence of the location of the two change points that delimit this segment. In the case of a hidden semi-Markov chain, observation distributions are fixed parameters (previously estimated) for state sequence or state profile computation while, in the case of a multiple change-point model, parameters attached to the segments are "contextual" parameters that depend on the change points.

Change-point profiles can be used to explore the parameter space for a fixed number of segments since the change points are parameters of the model for which posterior probabilities can be computed. Due to the deterministic succession of segments, most of the proposed algorithms have transdimensional properties (Hawkins, 2001), that is, the output of an algorithm for K segments, with $K = 2, \ldots, J-1$, can be computed as an almost free byproduct of the application of this algorithm for J segments.

The remainder of this paper is organized as follows. Some multiple change-point models and associated contrast functions are introduced in Section 2. The forward-backward dynamic programming algorithm is presented in Section 3 while the forward dynamic programming algorithm for computing the top L most probable segmentations is presented in Section 4. The smoothing-type forward-backward algorithm is presented in Section 5 and its variant for sampling segmentations is presented in Section 6. These algorithms for exploring the segmentation space are illustrated in Section 7 by examples corresponding to different multiple change-point models. We will show using these examples that the proposed methods may help to compare alternative multiple change-point models (e.g. Gaussian model with piecewise constant variances or global variance), predict supplementary change points, highlight overestimation of the number of change points and a specific type of change-point profile summarizes

the uncertainty concerning the location of change points. Section 8 consists of concluding remarks.

2 Multiple change-point model definition and recursive computation of contrast functions

Consider a sequence of observations $x_0, \ldots, x_{T-1}$ assumed to be partitioned into J segments where $\tau_1, \ldots, \tau_{J-1}$ denote the $J-1$ integer-valued change points with $0 < \tau_1 < \tau_2 < \cdots < \tau_{J-1} < T$. Let $\Delta\left(x_{\tau_j}, \ldots, x_{\tau_{j+1}-1};\ \widehat{\theta}_j\right)$ be a contrast (or fitness) function useful for estimating parameters θ_j attached to subsequence or segment $x_{\tau_j}, \ldots, x_{\tau_{j+1}-1}$. By convention, a change point is associated with an entry into a segment (rather than with an exit from a segment) and, $\tau_0 = 0$ and $\tau_J = T$. For the dynamic programming algorithms presented in Sections 3 and 4, this contrast function needs not be a log-likelihood function while for the smoothing-type algorithms presented in Sections 5 and 6, this contrast function is necessarily a log-likelihood function.

All the algorithms presented in the following sections compute functions of the change points $\tau_1, \ldots, \tau_{J-1}$. In fact, the problem solved by these algorithms decompose in two stages, the "outer" specific problem and the "inner" problem which consists in estimating the within-segment parameters $\theta_0, \ldots, \theta_{J-1}$ given the change points $\tau_1, \ldots, \tau_{J-1}$. In this Section, we focus on this inner problem. In contrast with the usual presentation of the dynamic programming algorithm for computing the most probable segmentation (see Auger and Lawrence (1989) and Braun and Müller (1998)), we do not present the algorithms with a preprocessing phase for computing the contrast functions for all the possible segments since this entails storing $T(T+1)/2$ contrast function values. We rather chose to embed the computation of the contrast functions within the recursions of the algorithms; see Appendix A for an illustration. Our objective is thus to not exceed the complexity of the core of these algorithms i.e. $O(JT^2)$-time and $O(JT)$-space (this linear space complexity is the stronger constraint for long sequences).

We first introduce two models (Gaussian model with piecewise constant means and variances and multinomial model with piecewise constant probabilities) that share a separability property in the sense that each parameter is attached to a single segment (i.e. there are no global parameters). Contrast functions which are true log-likelihood functions can then be defined.

Gaussian change in the mean and the variance model

It is assumed that the sequence consists of independent Gaussian distributed observations with piecewise constant means and variances $\left(\mu_j, \sigma_j^2;\ j = 0, \ldots, J-1\right)$. Given the change points $\tau_1, \ldots, \tau_{J-1}$, the maximum log-likelihood of a J-segment model for the sequence $x_0, \ldots, x_{T-1}$ is given by

$$
\begin{aligned}
&\log f\left(x_0,\ldots,x_{T-1};\widehat{\mu}_0,\widehat{\sigma}_0^2,\ldots,\widehat{\mu}_{J-1},\widehat{\sigma}_{J-1}^2\right)\\
&=-\sum_j \frac{\tau_{j+1}-\tau_j}{2}\left\{\log\frac{\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\widehat{\mu}_j\right)^2}{\tau_{j+1}-\tau_j}+\log 2\pi+1\right\} \qquad (1)\\
&=\sum_j \Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right),
\end{aligned}
$$

with $\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\widehat{\mu}_j\right)^2>0$ for each j and $\widehat{\mu}_j=\sum_{t=\tau_j}^{\tau_{j+1}-1}x_t/\left(\tau_{j+1}-\tau_j\right)$. The contrast function $\Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\ \widehat{\theta}_j\right)$ is the maximized log-likelihood for the segment $x_{\tau_j},\ldots,x_{\tau_{j+1}-1}$ obtained by substituting the estimate $\widehat{\theta}_j=\left(\widehat{\mu}_j,\widehat{\sigma}_j^2\right)$ for θ_j in the log-likelihood. We applied the recursive computation of the running sum of squared deviations from the running mean proposed by Hawkins and Zamba (2005)

$$
\begin{aligned}
\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\frac{\sum_{k=\tau_j}^{\tau_{j+1}-1}x_k}{\tau_{j+1}-\tau_j}\right)^2&=\sum_{t=\tau_j+1}^{\tau_{j+1}-1}\left(x_t-\frac{\sum_{k=\tau_j+1}^{\tau_{j+1}-1}x_k}{\tau_{j+1}-\tau_j-1}\right)^2\\
&\quad+\frac{\tau_{j+1}-\tau_j-1}{\tau_{j+1}-\tau_j}\left(x_{\tau_j}-\frac{\sum_{k=\tau_j+1}^{\tau_{j+1}-1}x_k}{\tau_{j+1}-\tau_j-1}\right)^2. \qquad (2)
\end{aligned}
$$

It is thus necessary to store the running sum $\sum_{t=\tau_j}^{\tau_{j+1}-1}x_t$ and the running sum of squared deviations from the running mean in order to recursively compute $\Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\ \widehat{\theta}_j\right)$ for all the possible segments. This solution is illustrated in the pseudo code of Appendix A.

Multinomial model

It is assumed that the sequence consists of independent categorical observations with piecewise constant probabilities $(p_{j,0},\ldots,p_{j,Y-1};\ j=0,\ldots,J-1)$. Given the change points $\tau_1,\ldots,\tau_{J-1}$, the maximum log-likelihood of a J-segment model for the sequence $x_0,\ldots,x_{T-1}$ is given by

$$
\begin{aligned}
&\log f\left(x_0,\ldots,x_{T-1};\widehat{p}_{0,0},\ldots,\widehat{p}_{0,Y-1},\ldots,\widehat{p}_{J-1,0},\ldots,\widehat{p}_{J-1,Y-1}\right)\\
&=\sum_j\sum_y n_y\left(\tau_j:\tau_{j+1}-1\right)\log\frac{n_y\left(\tau_j:\tau_{j+1}-1\right)}{\tau_{j+1}-\tau_j} \qquad (3)\\
&=\sum_j \Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right),
\end{aligned}
$$

with $n_y\left(\tau_j:\tau_{j+1}-1\right)=\sum_{t=\tau_j}^{\tau_{j+1}-1}I\left(x_t=y\right)$ and $I()$ denotes the indicator function. The contrast function $\Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\ \widehat{\theta}_j\right)$ is the log-likelihood of parameter $\widehat{\theta}_j=\left(\widehat{p}_{j,0},\ \ldots,\widehat{p}_{j,Y-1}\right)$ for the segment $x_{\tau_j},\ldots,x_{\tau_{j+1}-1}$.

Assuming that $x_{\tau_j}=a$, the following recursive computation of the contrast function (useful if the number of categories ≥ 4) can be applied

$$
\begin{aligned}
&\sum_y n_y\left(\tau_j : \tau_{j+1}-1\right)\log\frac{n_y\left(\tau_j : \tau_{j+1}-1\right)}{\tau_{j+1}-\tau_j}\\
=&\sum_y n_y\left(\tau_j+1 : \tau_{j+1}-1\right)\log\frac{n_y\left(\tau_j+1 : \tau_{j+1}-1\right)}{\tau_{j+1}-\tau_j-1}\\
&+\left(\tau_{j+1}-\tau_j-1\right)\log\frac{\tau_{j+1}-\tau_j-1}{\tau_{j+1}-\tau_j}+\log\frac{n_a\left(\tau_j+1 : \tau_{j+1}-1\right)+1}{\tau_{j+1}-\tau_j}\\
&-n_a\left(\tau_j+1 : \tau_{j+1}-1\right)\log\frac{n_a\left(\tau_j+1 : \tau_{j+1}-1\right)}{n_a\left(\tau_j+1 : \tau_{j+1}-1\right)+1}.
\end{aligned}
$$

These models can be combined in (possibly heterogeneous) multivariate change-point models. We here introduce another classic change-point model for which the contrast function is not a log-likelihood function.

Gaussian change in the mean model

It is assumed that the sequence consists of independent Gaussian distributed observations with piecewise constant means $(\mu_j;\ j=0,\ldots,J-1)$ and constant variance σ^2. Given the change points $\tau_1,\ldots,\tau_{J-1}$, the maximum log-likelihood of a J-segment model for the sequence $x_0,\ldots,x_{T-1}$ is given by

$$
\begin{aligned}
&\log f\left(x_0,\ldots,x_{T-1};\widehat{\mu}_0,\ldots,\widehat{\mu}_{J-1},\widehat{\sigma}^2\right)\\
=&-\frac{T}{2}\left\{\log\frac{\sum_j\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\widehat{\mu}_j\right)^2}{T}+\log 2\pi+1\right\},
\end{aligned}
\tag{4}
$$

with $\sum_j\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\widehat{\mu}_j\right)^2>0$.

The contrast function is $-1\times$(sum of squared deviations from the mean) for the segment $x_{\tau_j},\ldots,x_{\tau_{j+1}-1}$,

$$
\Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right)=-\sum_{t=\tau_j}^{\tau_{j+1}-1}\left(x_t-\widehat{\mu}_j\right)^2,
$$

since the dynamic programming algorithms are stated as maximization algorithms (instead of minimization algorithms). This contrast function can be recursively computed using (2). With this model, the dynamic programming algorithms (Sections 3 and 4) can still be applied while the smoothing-type algorithms (Sections 5 and 6) cannot be applied.

3 Forward-backward dynamic programming algorithm

The distinctive property of the change-point and segment profiles computed by the forward-backward dynamic programming algorithm is to select parts of segmentations and in particular the entire most probable segmentation. In this way, structural differences between sub-optimal segmentations and the most probable segmentation are highlighted. The forward-backward dynamic programming algorithm is based on the following decomposition

$$
\begin{aligned}
&\max_{0<\tau_1<\cdots<\tau_{j-1}<\tau_j=t<\tau_{j+1}<\cdots<\tau_{J-1}<T} \sum_{i=0}^{J-1} \Delta\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
=&\max_{0<\tau_1<\cdots<\tau_{j-1}<\tau_j=t} \sum_{i=0}^{j-1} \Delta\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
&+\max_{\tau_j=t<\tau_{j+1}<\cdots<\tau_{J-1}<T} \sum_{i=j}^{J-1} \Delta\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
=&\alpha_{j-1}(t-1)+\beta_j(t),
\end{aligned}
$$

where $\alpha_{j-1}(t-1)$ is computed in a forward recursion (i.e. from 0 to $T-1$) and $\beta_j(t)$ is computed in a backward recursion (i.e. from $T-1$ to 0). In the standard statement of the algorithm, $\alpha_j(t)$ is the maximum log-likelihood for the first segments, segment j ending at time t, $\beta_j(t)$ is the maximum log-likelihood for the last segments, segment j beginning at time t and $\alpha_{j-1}(t-1)+\beta_j(t)$ is the maximum log-likelihood for the J segments, segment j beginning at time t.

In the case of change-point profiles, the output of the forward-backward dynamic programming algorithm is directly the quantities $\alpha_{j-1}(t-1)+\beta_j(t)$ (scaled appropriately; see (7) and (9)) for each time t and each segment j. The aim of the forward-backward dynamic programming algorithm used to compute segment profiles is to compute for each time t and each segment j

$$
\gamma_j(t)=\max_{0<\tau_1<\cdots<\tau_j\leq t<\tau_{j+1}<\cdots<\tau_{J-1}<T} \sum_{i=0}^{J-1} \Delta\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right).
$$

In the standard statement of the algorithm, $\gamma_j(t)$ is the maximum log-likelihood for the J segments, time t falling in segment j. In the case of the Gaussian change in the mean model, the contrast function $\Delta\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right)$ is $-1\times$(sum of squared deviations from the mean) for segment j and the quantities $\alpha_j(t),\beta_j(t)$ and $\gamma_j(t)$ are defined accordingly.
The forward recursion is given by,
$j=0:$

$$
\alpha_0(t)=\Delta\left(x_0,\ldots,x_t;\widehat{\theta}_0\right),
$$

$j=1,\ldots,J-1:$

$$
\begin{aligned}
\alpha_j(t)=&\max_{0<\tau_1<\cdots<\tau_j<\tau_{j+1}=t+1} \sum_{i=0}^{j} \Delta\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
=&\max_{j\leq\tau_j\leq t}\left\{\Delta\left(x_{\tau_j},\ldots,x_t;\widehat{\theta}_j\right)+\alpha_{j-1}(\tau_j-1)\right\}. \qquad (5)
\end{aligned}
$$

This forward recursion complemented by a backtracking procedure to retrieve the most probable segmentation constitutes the classic forward dynamic programming algorithm; see Auger and Lawrence (1989) and Hawkins (2001).

The backward recursion is given by,
$j = 0, \ldots, J-2$:

$$\begin{aligned}\beta_j(t) &= \max_{\tau_j = t < \tau_{j+1} < \cdots < \tau_{J-1} < T} \sum_{i=j}^{J-1} \Delta\left(x_{\tau_i}, \ldots, x_{\tau_{i+1}-1}; \widehat{\theta}_i\right) \\ &= \max_{t < \tau_{j+1} \leq T-(J-j)+1} \left\{\Delta\left(x_t, \ldots, x_{\tau_{j+1}-1}; \widehat{\theta}_j\right) + \beta_{j+1}(\tau_{j+1})\right\}, \qquad (6)\end{aligned}$$

$j = J-1$:

$$\beta_{J-1}(t) = \Delta\left(x_t, \ldots, x_{T-1}; \widehat{\theta}_{J-1}\right).$$

For segment profile computation, the backward recursion includes the following additional maximization step. For each time $u > t$ (taken in decreasing order), the quantity

$$\alpha_{j-1}(t-1) + \max_{u < \tau_{j+1} \leq T-(J-j)+1} \left\{\Delta\left(x_t, \ldots, x_{\tau_{j+1}-1}; \widehat{\theta}_j\right) + \beta_{j+1}(\tau_{j+1})\right\}$$

should be compared with the current evaluation of

$$\gamma_j(u) = \max_{0 < \tau_1 < \cdots < \tau_j \leq u < \tau_{j+1} < \cdots < \tau_{J-1} < T} \sum_{i=0}^{J-1} \Delta\left(x_{\tau_i}, \ldots, x_{\tau_{i+1}-1}; \widehat{\theta}_i\right).$$

In this manner, the mandatory maximization required for computing $\beta_j(t)$ (see (6)) is re-used in the computation of $(\gamma_j(u)\ ; u = t+1, \ldots, T-(J-j))$ with only a single supplementary maximization for each u. It should be noted that the quantity $\gamma_j(t)$ is initialized at time t with $\alpha_{j-1}(t-1) + \beta_j(t)$.
In the computation of

$$\gamma_j(t) = \max_{0 < \tau_1 < \cdots < \tau_j \leq t < \tau_{j+1} < \cdots < \tau_{J-1} < T} \sum_{i=0}^{J-1} \Delta\left(x_{\tau_i}, \ldots, x_{\tau_{i+1}-1}; \widehat{\theta}_i\right),$$

the maximizations on τ_{j+1} for each τ_j are byproducts of the computation of $\{\beta_j(\tau_j)\ ;\ \tau_j = j, \ldots, t\}$ while the maximization on τ_j requires a supplementary maximization for each τ_j. This means that the computation of $\gamma_j(t)$ instead of solely $\alpha_{j-1}(t-1) + \beta_j(t)$ does not entail a change in the order of magnitude of algorithm complexity. The complexity of the forward-backward dynamic programming algorithm is thus similar to the complexity of the classic forward dynamic programming algorithm (Auger and Lawrence, 1989) i.e. $O(JT^2)$-time and $O(JT)$-space. It should be noted that $\beta_0(0) = \alpha_{J-1}(T-1) = \gamma_0(0) = \gamma_{J-1}(T-1)$. An implementation of this algorithm is proposed in Appendix A in pseudo-code form.

The most probable segmentation $(s_t; t = 0, \ldots, T-1)$ can be directly deduced as

$$s_t = \arg\max_j \{\gamma_j(t)\}.$$

In practice, if the contrast function $\Delta(\,)$ is a log-likelihood denoted by $\log f(\,)$, the posterior probabilities

$$\begin{aligned}
&\exp\left\{\max_{0<\tau_1<\cdots<\tau_{j-1}<\tau_j=t<\tau_{j+1}<\cdots<\tau_{J-1}<T}\ \sum_{i=0}^{J-1}\log f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right)\right.\\
&\qquad\left.-\log f\left(x_0,\ldots,x_{T-1};J\right)\right\}\\
&=\exp\left\{\alpha_{j-1}\left(t-1\right)+\beta_j\left(t\right)-\log f\left(x_0,\ldots,x_{T-1};J\right)\right\},
\end{aligned}\tag{7}$$

for change-point profiles, or

$$\begin{aligned}
&\exp\left\{\max_{0<\tau_1<\cdots<\tau_j\leq t<\tau_{j+1}<\cdots<\tau_{J-1}<T}\ \sum_{i=0}^{J-1}\log f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right)\right.\\
&\qquad\left.-\log f\left(x_0,\ldots,x_{T-1};J\right)\right\}\\
&=\exp\left\{\gamma_j\left(t\right)-\log f\left(x_0,\ldots,x_{T-1};J\right)\right\},
\end{aligned}\tag{8}$$

for segment profiles should preferably be used. The normalizing constant $\log f\left(x_0,\ldots,x_{T-1};J\right)$, which is the log-likelihood of all the possible J-segment models for the sequence $x_0,\ldots,x_{T-1}$, is computed by the forward recursion (12) of the smoothing-type forward-backward algorithm (see Section 5) with

$$f\left(x_0,\ldots,x_{T-1};J\right)=\sum_{\tau_1,\ldots,\tau_{J-1}}\ \prod_{j=0}^{J-1}f\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right).$$

In the case of the Gaussian change in the mean model, the normalized quantities (between 0 and 1)

$$\begin{aligned}
&\exp\left[-\frac{T}{2}\left\{\log-\frac{\alpha_{j-1}\left(t-1\right)+\beta_j\left(t\right)}{T}+\log 2\pi+1\right\}\right.\\
&\qquad\left.+\frac{T}{2}\left\{\log-\frac{\alpha_{J-1}\left(T-1\right)}{T}+\log 2\pi+1\right\}\right]\\
&=\left\{\frac{\alpha_{j-1}\left(t-1\right)+\beta_j\left(t\right)}{\alpha_{J-1}\left(T-1\right)}\right\}^{-T/2},
\end{aligned}\tag{9}$$

for change-point profiles, or

$$\begin{aligned}
&\exp\left[-\frac{T}{2}\left\{\log-\frac{\gamma_j\left(t\right)}{T}+\log 2\pi+1\right\}+\frac{T}{2}\left\{\log-\frac{\max_i\gamma_i\left(t\right)}{T}+\log 2\pi+1\right\}\right]\\
&=\left\{\frac{\gamma_j\left(t\right)}{\alpha_{J-1}\left(T-1\right)}\right\}^{-T/2},
\end{aligned}$$

for segment profiles should preferably be used.

In the same manner as for the classic forward dynamic programming algorithm for computing the most probable segmentation (Hawkins, 2001), the

change-point profiles for $K = 2, \ldots, J-1$ segments, can be obtained as an almost free byproduct of the forward-backward dynamic programming algorithm for J segments. It is also easy to generalize the above-described algorithm for simultaneous computation of the segment profiles for $K = 2, \ldots, J-1$. In this case, the resulting algorithm is not an almost free byproduct of the basic algorithm for J segments and its complexity is $O(J^2T^2)$-time.

4 Forward dynamic programming algorithm for computing the top L most probable segmentations in J segments

The aim here is to compute the top L most probable segmentations. The optimality of this algorithm relies on the fact that the top L most probable segmentations of $x_0, \ldots, x_t$ leaving segment j at time t require at most computation of the top L most probable segmentations of $x_0, \ldots, x_{\tau_j - 1}$ leaving segment $j-1$ at times $\tau_j - 1$ for $\tau_j = j, \ldots, t$, the extreme situation being the case where the top L most probable segmentations leaving segment j at time t are built from the top L most probable segmentations leaving segment $j-1$ at the same time.

The number of possible segmentations is $\binom{T-1}{J-1}$. Let $L_j(t)$ denote the number of segmentations of $x_0, \ldots, x_t$ leaving segment j at time t. Hence, $L_j(t) = \binom{t}{j}$. For a given time t, the number of segmentations can be recursively computed ($L_0(t) = 1$ and $L_j(t) = L_{j-1}(t)(t-j+1)/j$) and then thresholded at L.
The forward recursion is given by,
$j = 0$:

$$\alpha_0^1(t) = \Delta\left(x_0, \ldots, x_t; \widehat{\theta}_0\right),$$

$j = 1, \ldots, J-1$:

The ranks of the segmentations of $x_0, \ldots, x_{\tau_j - 1}$ leaving segment $j-1$ at times $\tau_j - 1, (r(\tau_j);\ \tau_j = j, \ldots, t)$ should be initialized at 1.
$n = 1, \ldots, L_j(t)$:

$$\alpha_j^n(t) = \max_{j \le \tau_j \le t} \left\{ \Delta\left(x_{\tau_j}, \ldots, x_t; \widehat{\theta}_j\right) + \alpha_{j-1}^{r(\tau_j)}(\tau_j - 1) \right\}.$$

The above maximization selects a change-point location τ_j and a segmentation of $x_0, \ldots, x_{\tau_j - 1}$ leaving segment $j-1$ at time $\tau_j - 1$ with associated rank $r(\tau_j)$. This rank should then be incremented by one to prevent reselecting the same configuration.

In the standard statement of the algorithm, $\alpha_{J-1}^n(T-1)$ is the log-likelihood of the nth most probable segmentation from which the posterior probability of the nth most probable segmentation can directly be deduced as in (7) and (8). In the case of the Gaussian change in the mean model, the log-likelihood of the nth most probable segmentation is

$$-\frac{T}{2}\left\{\log - \frac{\alpha_{J-1}^n(T-1)}{T} + \log 2\pi + 1\right\}.$$

To retrieve the top L most probable segmentations, the recursion described above should be complemented by a backtracking procedure. The backtracking

procedure operates by jumps on the basis of two backpointers, the first giving the optimal preceding change point and the second the associated rank. An implementation of this algorithm is proposed in Appendix B in pseudo-code form. The complexity of this algorithm is $O(LJT^2)$-time and $O(LJT)$-space. This space complexity may become a limitation in the case of long sequences.

The forward-backward dynamic programming algorithm introduced in Section 3 for computing change-point or segment profiles, and the forward dynamic programming algorithm for computing the top L most probable segmentations, give two complementary points of view on the possible segmentations of a sequence. While the top L most probable segmentations are enumerated in one case, the output of the forward-backward dynamic programming algorithm can be viewed as the superposition (from the less probable to the most probable) of all the segmentations in a $J \times T$ array. The dynamic programming algorithm for computing the top L most probable segmentations is mainly useful when the cumulated posterior probability of the top L most probable segmentations is reasonably high, L being small. Recall that the number of possible segmentations is $\binom{T-1}{J-1}$. Hence, the application scope of this algorithm is far more restricted than these of the forward-backward dynamic programming algorithm.

5 Smoothing-type forward-backward algorithm

The smoothing-type forward-backward algorithm applies to a more restricted class of models than the dynamic programming algorithms presented in Sections 3 and 4. These dynamic programming algorithms rely on contrast functions which are additive in j, the segment index. The smoothing-type forward-backward algorithm relies on additive log-likelihood functions such as (1) and (3). In particular, the smoothing-type forward-backward algorithm does not apply to the Gaussian change in the mean model since the log-likelihood function (4) is not additive in j (a consequence of the fact that the estimated global variance depends on the estimated within-segment means). For models whose log-likelihood function is additive, we saw previously that the outputs of the dynamic programming algorithms can be normalized by the log-likelihood of all the possible J-segment models for the sequence $x_0, \ldots, x_{T-1}$ which is a byproduct of the smoothing-type forward-backward algorithm; see (7) and (8). This is a great advantage for the interpretation of the algorithm outputs which are expressed in terms of posterior probabilities; see illustrations in Section 7.

Contrarily to the change-point and segment profiles computed by the forward-backward dynamic programming algorithm (Section 3) where parts of individual segmentations and in particular the entire most probable segmentation are directly apparent, individual segmentations are not apparent in the change-point or segment profiles computed by the smoothing-type forward-backward algorithm. The idea here is to summarize all the possible segmentations (i.e. all the possible multiple change-point models) for each time t and each segment j. The smoothing-type forward-backward algorithm is based on the following decomposition

$$
\begin{aligned}
&\sum_{\tau_1,\ldots,\tau_{j-1},\tau_{j+1},\ldots,\tau_{J-1};\tau_j=t} \prod_{i=0}^{J-1} f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
=&\sum_{\tau_1,\ldots,\tau_{j-1};\tau_j=t} \prod_{i=0}^{j-1} f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
&\times \sum_{\tau_{j+1},\ldots,\tau_{J-1};\tau_j=t} \prod_{i=j}^{J-1} f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
=&\widetilde{F}_{j-1}\left(t-1\right)\widetilde{B}_j\left(t\right),
\end{aligned}
$$

where $\widetilde{F}_{j-1}\left(t-1\right)$ is computed in a forward recursion (i.e. from 0 to $T-1$) and $\widetilde{B}_j\left(t\right)$ is computed in a backward recursion (i.e. from $T-1$ to 0). The quantity $\widetilde{F}_j\left(t\right)$ is the likelihood of all the possible segmentations of $x_0,\ldots,x_t$, segment j ending at time t, $\widetilde{B}_j\left(t\right)$ is the likelihood of all the possible segmentations of $x_t,\ldots,x_{T-1}$, segment j beginning at time t, $\widetilde{F}_{j-1}\left(t-1\right)\widetilde{B}_j\left(t\right)$ is the likelihood of all the possible segmentations of $x_0,\ldots,x_{T-1}$, segment j beginning at time t and $f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right)$ is the likelihood of parameter $\widehat{\theta}_i$ for the segment $x_{\tau_i},\ldots,x_{\tau_{i+1}-1}$.
The forward recursion is given by,
$j=0:$

$$
\widetilde{F}_0\left(t\right)=f\left(x_0,\ldots,x_t;\widehat{\theta}_0\right),
$$

$j=1,\ldots,J-1:$

$$
\begin{aligned}
\widetilde{F}_j\left(t\right)&=\sum_{\tau_1,\ldots,\tau_j;\tau_{j+1}=t+1} \prod_{i=0}^{j} f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
&=\sum_{\tau_j=j}^{t} f\left(x_{\tau_j},\ldots,x_t;\widehat{\theta}_j\right)\widetilde{F}_{j-1}\left(\tau_j-1\right). \qquad (10)
\end{aligned}
$$

The backward recursion is given by,
$j=0,\ldots,J-2:$

$$
\begin{aligned}
\widetilde{B}_j\left(t\right)&=\sum_{\tau_{j+1},\ldots,\tau_{J-1};\tau_j=t} \prod_{i=j}^{J-1} f\left(x_{\tau_i},\ldots,x_{\tau_{i+1}-1};\widehat{\theta}_i\right) \\
&=\sum_{\tau_{j+1}=t+1}^{T-(J-j)+1} f\left(x_t,\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right)\widetilde{B}_{j+1}\left(\tau_{j+1}\right), \qquad (11)
\end{aligned}
$$

$j=J-1:$

$$
\widetilde{B}_{J-1}\left(t\right)=f\left(x_t,\ldots,x_{T-1};\widehat{\theta}_{J-1}\right).
$$

These forward and backward recursions are similar to the forward and backward dynamic programming recursions (maximization in (5) (6) replaced by summation in (10) (11) and contrast function in (5) (6) replaced by likelihood in (10) (11)). Unlike the dynamic programming algorithm where the forward quantities $\alpha_j(t)$ and the backward quantities $\beta_j(t)$ correspond to a single optimal segmentation, the forward quantities $\widetilde{F}_j(t)$ and the backward quantities $\widetilde{B}_j(t)$ here summarize all the possible segmentations whose combination rely necessarily on probabilities.

A naive implementation of the forward and backward recursions (10) (11) generates underflows. It is mandatory here to apply a log-transform for the computation of the log-likelihood attached to the current segment in both recursions. Hence, a naive implementation of the forward recursion (10) consists of computing $\log f\left(x_{\tau_j},\ldots,x_t;\widehat{\theta}_j\right)+\log\widetilde{F}_{j-1}(\tau_j-1)$ for each τ_j, then summing these terms after an exponential transform. It should be noted that unlike the case of hidden semi-Markov chains (Guédon, 2003), it is not possible to design a forward-backward algorithm purely in terms of posterior probabilities.

We here propose a scaling scheme inspired by that proposed by Levinson (1986) in the context of hidden semi-Markov chains. The practical forward recursion is given by,
$j=0$:

$$\begin{aligned}F_0(t)=&\frac{1}{N_t}\exp\left\{\log f\left(x_0,\ldots,x_t;\widehat{\theta}_0\right)-\sum_{u=0}^{t-1}\log N_u\right\}\\=&\frac{G_0(t)}{N_t},\end{aligned}$$

$j=1,\ldots,J-1$:

$$\begin{aligned}F_j(t)=&\frac{1}{N_t}\sum_{\tau_j=j}^{t}\exp\left\{\log f\left(x_{\tau_j},\ldots,x_t;\widehat{\theta}_j\right)-\sum_{u=\tau_j}^{t-1}\log N_u\right\}\\&\times F_{j-1}(\tau_j-1)\\=&\frac{G_j(t)}{N_t}.\end{aligned}\tag{12}$$

where $N_t=\sum_j G_j(t)$. The sums of the log-normalizing factors N_t are recursively computed during the forward recursion; see Appendix C.
Since

$$\log F_j(t)=\log\widetilde{F}_j(t)-\sum_{u=0}^{t}\log N_u,$$

the log-likelihood of all the possible J-segment models for the sequence $x_0,\ldots,x_{T-1}$ can be directly deduced as

$$\log f(x_0,\ldots,x_{T-1};J)=\log F_{J-1}(T-1)+\sum_{t=0}^{T-1}\log N_t.$$

The practical backward recursion is given by,

$j = 0, \ldots, J-2$:

$$\begin{aligned} B_j(t) =& \frac{1}{M_t} \sum_{\tau_{j+1}=t+1}^{T-(J-j)+1} \exp\left\{ \log f\left(x_t, \ldots, x_{\tau_{j+1}-1}; \widehat{\theta}_j\right) - \sum_{u=t+1}^{\tau_{j+1}-1} \log M_u \right\} \\ & \times B_{j+1}(\tau_{j+1}) \\ =& \frac{C_j(t)}{M_t}, \end{aligned} \tag{13}$$

$j = J-1$:

$$\begin{aligned} B_{J-1}(t) =& \frac{1}{M_t} \exp\left\{ \log f\left(x_t, \ldots, x_{T-1}; \widehat{\theta}_{J-1}\right) - \sum_{u=t+1}^{T-1} \log M_u \right\} \\ =& \frac{C_{J-1}(t)}{M_t}. \end{aligned}$$

where $M_t = \sum_j C_j(t)$.

The posterior probability of entering segment j at time t for $t > 0$ and consequently $j > 0$ can be directly extracted as

$$\begin{aligned} & \sum_{\tau_1, \ldots, \tau_{j-1}, \tau_{j+1}, \ldots, \tau_{J-1}; \tau_j = t} \prod_{i=0}^{J-1} f\left(x_{\tau_i}, \ldots, x_{\tau_{i+1}-1}; \widehat{\theta}_i\right) / f(x_0, \ldots, x_{T-1}; J) \\ =& F_{j-1}(t-1) B_j(t) \exp\left\{ \sum_{u=0}^{t-1} \log N_u + \sum_{u=t}^{T-1} \log M_u - \log f(x_0, \ldots, x_{T-1}; J) \right\} \\ =& \widetilde{F}_{j-1}(t-1) \widetilde{B}_j(t) / f(x_0, \ldots, x_{T-1}; J). \end{aligned}$$

The posterior probability of being in segment j at time $t < T-1$ (smoothed probability) is given by

$$\begin{aligned} & L_j(t) \\ =& \sum_{\tau_1, \ldots, \tau_{J-1}; \tau_j \le t < \tau_{j+1}} \prod_{i=0}^{J-1} f\left(x_{\tau_i}, \ldots, x_{\tau_{i+1}-1}; \widehat{\theta}_i\right) / f(x_0, \ldots, x_{T-1}; J) \\ =& F_j(t) B_{j+1}(t+1) \exp\left\{ \sum_{u=0}^{t} \log N_u + \sum_{u=t+1}^{T-1} \log M_u - \log f(x_0, \ldots, x_{T-1}; J) \right\} \\ & + L_j(t+1) - F_{j-1}(t) B_j(t+1) \\ & \times \exp\left\{ \sum_{u=0}^{t} \log N_u + \sum_{u=t+1}^{T-1} \log M_u - \log f(x_0, \ldots, x_{T-1}; J) \right\}, \end{aligned}$$

where the first term is the posterior probability of leaving segment j at time t and the last term is the posterior probability of entering segment j at time $t+1$. The complexity of this algorithm is $O(JT^2)$-time and $O(JT)$-space. An

implementation of this algorithm is proposed in Appendix C in pseudo-code form.

Both the change-point and the segment profiles for $K = 2, \ldots, J-1$ segments, can be obtained as an almost free byproduct of the forward-backward algorithm for J segments. For instance, the posterior probability of entering segment j at time t for $t > 0$ and consequently $j > 0$ for a K-segment model can be directly extracted as

$$F_{j-1}(t-1) B_{j+J-K}(t) \exp\left\{\sum_{u=0}^{t-1} \log N_u + \sum_{u=t}^{T-1} \log M_u - \log f(x_0, \ldots, x_{T-1}; K)\right\},$$

where the log-likelihood of all the possible K-segment models for the sequence $x_0, \ldots, x_{T-1}$ is given by

$$\log f(x_0, \ldots, x_{T-1}; K) = \log F_{K-1}(T-1) + \sum_{t=0}^{T-1} \log N_t.$$

The posterior probability of entering a segment at time t, irrespective of its rank, can directly be deduced as

$$\begin{aligned} & P(R_t = 1) \\ = & \left\{\sum_j F_{j-1}(t-1) B_j(t)\right\} \\ & \times \exp\left\{\sum_{u=0}^{t-1} \log N_u + \sum_{u=t}^{T-1} \log M_u - \log f(x_0, \ldots, x_{T-1}; J)\right\}. \end{aligned} \tag{14}$$

This univariate change-point profile provides a complementary point of view with reference to the multivariate change-point profile where only the uncertainty relative to the change-point location is reflected.

The sum of the entropies computed from the successive binary random variables R_t can be used as an indicator of overparameterization of the multiple change-point model (see Section 8 for an illustration)

$$\begin{aligned} \sum_t H(R_t) = & \sum_t \left[\{1 - P(R_t = 1)\} \log \{1 - P(R_t = 1)\}\right. \\ & \left. + P(R_t = 1) \log P(R_t = 1)\right]. \end{aligned} \tag{15}$$

6 Forward-backward algorithm for sampling segmentations

Instead of computing the top L most probable segmentations, it may be interesting to sample possible segmentations. The forward-backward algorithm for sampling segmentations decomposes into two passes, a forward pass which is the usual forward recursion (12) of the smoothing-type forward-backward algorithm, and a backward pass for sampling segmentations. This backward pass

can be seen as a stochastic backtracking procedure, in contrast to the optimal backtracking procedure of the classic forward dynamic programming algorithm and its generalization for computing the top L most probable segmentations.

In this backward pass, the length of segment j with $j > 0$, segment j ending at time t, is drawn from the distribution

$$\begin{aligned}\mathrm{Aux}_j\left(t, \tau_j\right) = \exp & \left\{ \log f\left(x_{\tau_j}, \ldots, x_t; \widehat{\theta}_j\right) - \sum_{u=\tau_j}^{t} \log N_u \right\} \\ & \times F_{j-1}\left(\tau_j - 1\right) / F_j\left(t\right), \qquad \tau_j = j, \ldots, t.\end{aligned}$$

The probabilities $\left(\mathrm{Aux}_j\left(t, \tau_j\right); \tau_j = j, \ldots, t\right)$ are simply the terms summed for each τ_j in the computation of the forward probability $F_j\left(t\right)$ (12) divided by this forward probability.

Similar algorithms were proposed as building blocks of Bayesian estimation methods for multiple change-point models. For instance, Fearnhead (2006) proposed a backward-forward algorithm for sampling segmentations. This algorithm starts with a backward recursion similar to (11) where the segment probabilities are obtained by integrating (and not maximizing as in our case) over the within-segment parameter space and that includes a supplementary term which is the prior probability of the segment length. Segmentations are then sampled in parallel in a forward pass.

7 Applications

7.1 Corsican pine growth phases

In this example, 6 trunks of approximately 70-year-old Corsican pines (*Pinus nigra* Arn. ssp. *laricio* Poir., Pinaceae) planted in two forest stands in the "Centre" region (France) were described by annual shoot (the morphological markers for approximately the first 2 years were not observable and these years were therefore not considered). An annual shoot is defined as the segment of stem established within a year. These 6 trees were taken from two stands with contrasting densities (1, 2, 3: 200 stems/ha; 4, 5, 6: 290 stems/ha) and correspond for each stand to three different diameter classes (1: 53 cm; 2: 45 cm; 3: 36 cm; 4: 48 cm; 5: 39 cm; 6: 29 cm). Diameters are conventionally measured at breast height. Two interval-scaled variables were recorded for each annual shoot, namely length (in cm) and number of branches per tier. Hence, information from both the parent entity (length of the annual shoot) and the offspring entities (number of branches per tier) were combined in the measurements. In the Corsican pine case, branches of roughly equivalent size are located at the top of the shoot just below the shoot limit and thus form a tier of branches.

The observed growth is the result of the modulation of the endogenous growth component by climatic factors. For the endogenous growth component, two assumptions can be made:

- the endogenous growth component takes the form of a trend,
- the endogenous growth component is structured as a succession of phases separated by marked transitions.

The aim of this study was to investigate this latter assumption.

We selected the median diameter individual of the 200 stems/ha plot (individual 2), considering only the lengths of the successive annual shoots, to illustrate the behavior of the methods for exploring the segmentation space. Segmentations of the other individuals, based in some cases on the original bivariate sequences, are presented in Guédon *et al.* (2007).

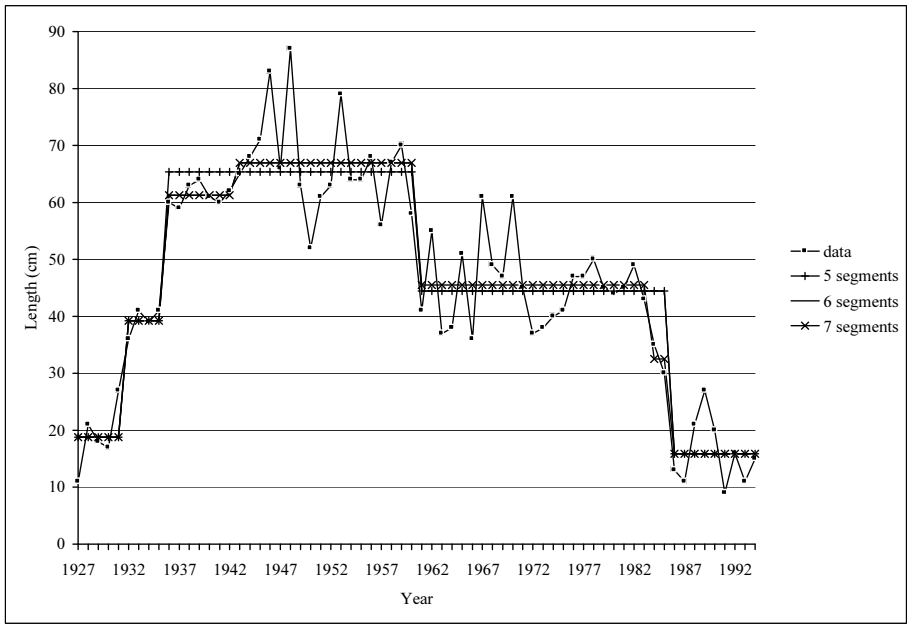

Figure 1: Corsican pine: Segmentations for $J = 5, 6, 7$.

The most probable segmentations using Gaussian change in the mean and variance models are shown in Figure 1 for $J = 5, 6, 7$. For $J = 5$, there are 766 480 possible segmentations but the cumulated posterior probability of the top 30 most probable segmentations exceeds 0.9 and the cumulated posterior probability of the top 216 most probable segmentations exceeds 0.99. The top 10 most probable segmentations (see Section 4) corresponding to the top 10 most likely change-point models are:

`1932, 1936, 1961, 1986` (0.17 0.17 68)
`1932, 1936, 1961, 1985` (0.145 0.315 69)
`1932, 1936, 1960, 1986` (0.061 0.376 70)
`1933, 1936, 1961, 1986` (0.053 0.429 71)
`1932, 1936, 1960, 1985` (0.053 0.482 71)
`1933, 1936, 1961, 1985` (0.045 0.527 71)
`1936, 1943, 1961, 1986` (0.041 0.568 81)
`1936, 1943, 1961, 1985` (0.035 0.603 81)
`1936, 1944, 1961, 1986` (0.033 0.636 82)
`1932, 1936, 1961, 1984` (0.031 0.667 83)

where the first indicator in parentheses is the posterior probability of the segmentation, the second indicator is the cumulated posterior probability of the top n most probable segmentations and the last indicator is the number of cells

used by the top n most probable segmentations in the $J \times T$ array. The number of cells can be interpreted as the beam size of the top n most probable segmentations. Since the number of cells does not increase for the 5th, 6th and 8th segmentations, these segmentations are masked in the segment profiles computed by the forward-backward dynamic programming algorithm (Section 3) shown in Figure 2a.

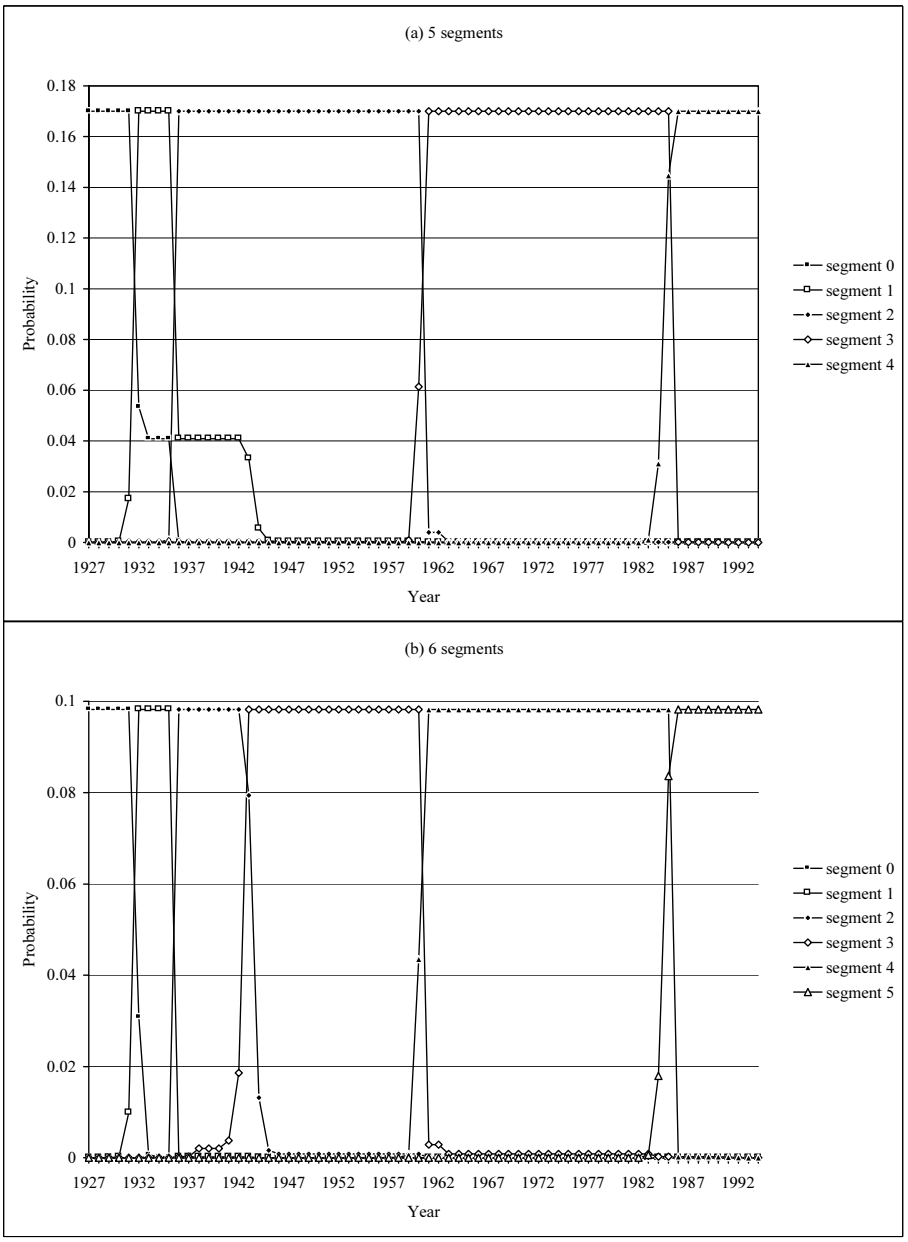

Figure 2: Corsican pine: Segment profiles computed by the forward-backward dynamic programming algorithm

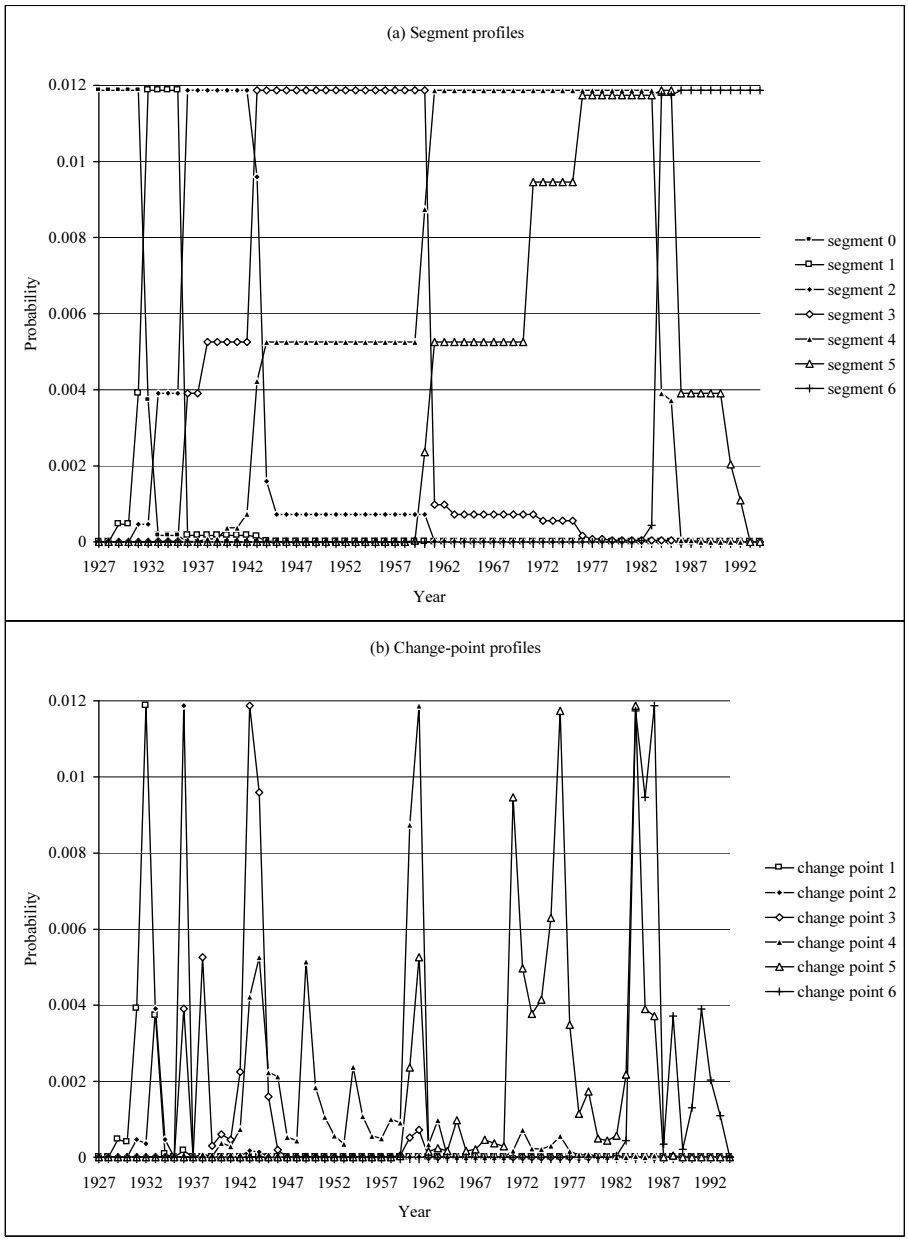

Figure 3: Corsican pine: Profiles computed for 7 segments by the forward-backward dynamic programming algorithm.

With reference to the most probable segmentation in 5 segments (posterior probability of 0.17), the two first segments are merged and the third is split (and this induces a shift of segments 1 and 2) in the 7th segmentation with posterior probability of 0.041 (Figure 2a). In the case of an underparameterization of the model, a missing change point may be detected by a single merge, shift and split (or split, shift and merge) of segments. The supplementary change point (1942 $\rightarrow$ 1943) predicted in the case of 5 segments (Figure 2a) with the

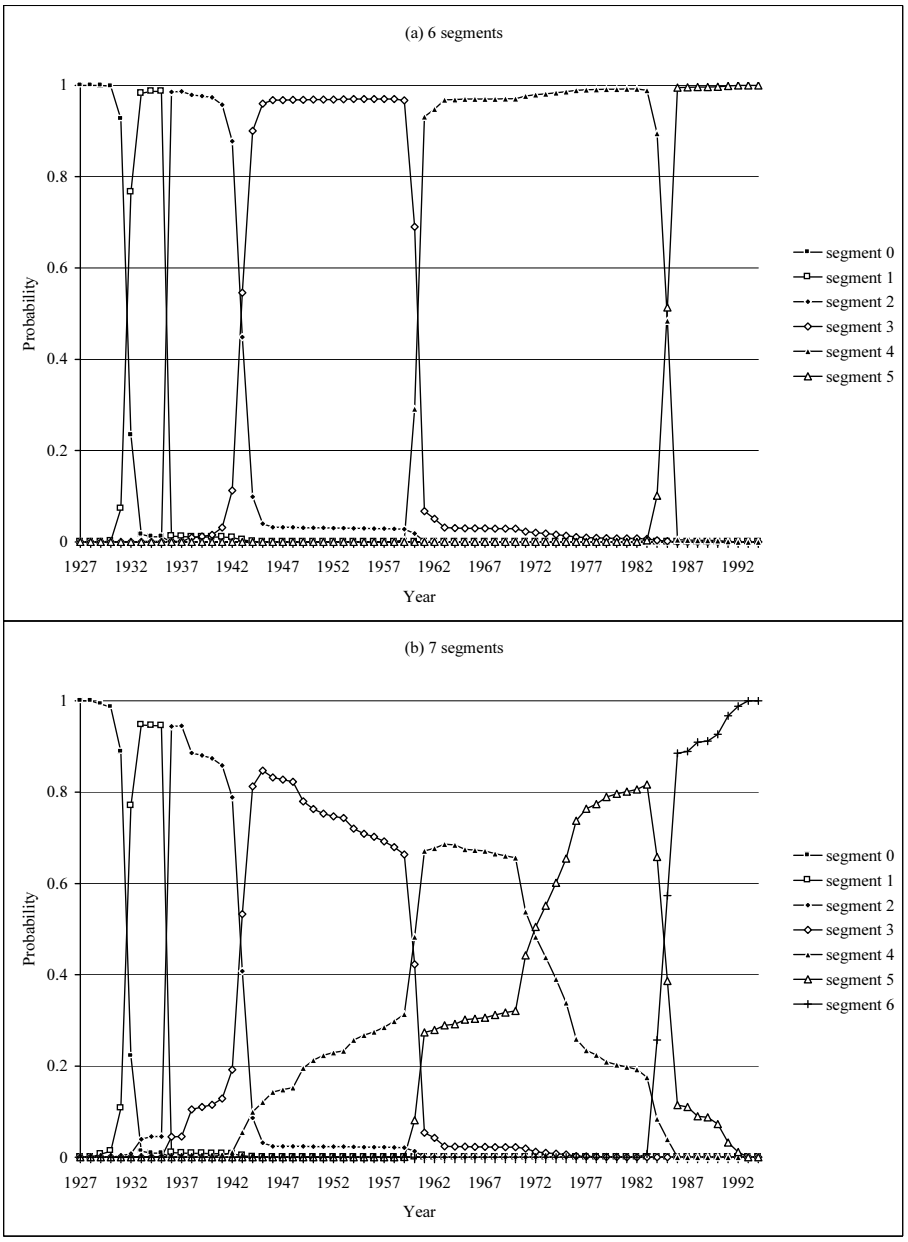

Figure 4: Corsican pine: Segment profiles computed by the smoothing-type forward-backward algorithm.

7th segmentation is an optimal change point in the case of 6 segments (Figure 2b), the 4 optimal change points found for 5 segments being conserved in the case of 6 segments (Figures 1 and 2). A too high number of segments may generate numerous merges, shifts and splits (or splits, shifts and merges) of segments as shown in Figure 3a for 7 segments. This overparameterization is also reflected in the posterior probability of the optimal segmentation which

is 0.012 for 7 segments instead of 0.098 for 6 segments; this point is further discussed in Section 8. In the case of an overparameterization of the model, far more alternative segmentations are apparent in the change-point profiles (Figure 3b) compared to the segment profiles (Figure 3a) both computed by the forward-backward dynamic programming algorithm. This overparameterization is also apparent in the segment profiles computed by the smoothing-type forward-backward algorithm (Section 5) where the uncertainty concerning the location of the 6 segments is low (Figure 4a) while it is far higher in the case of 7 segments (Figure 4b), particularly for the median segments.

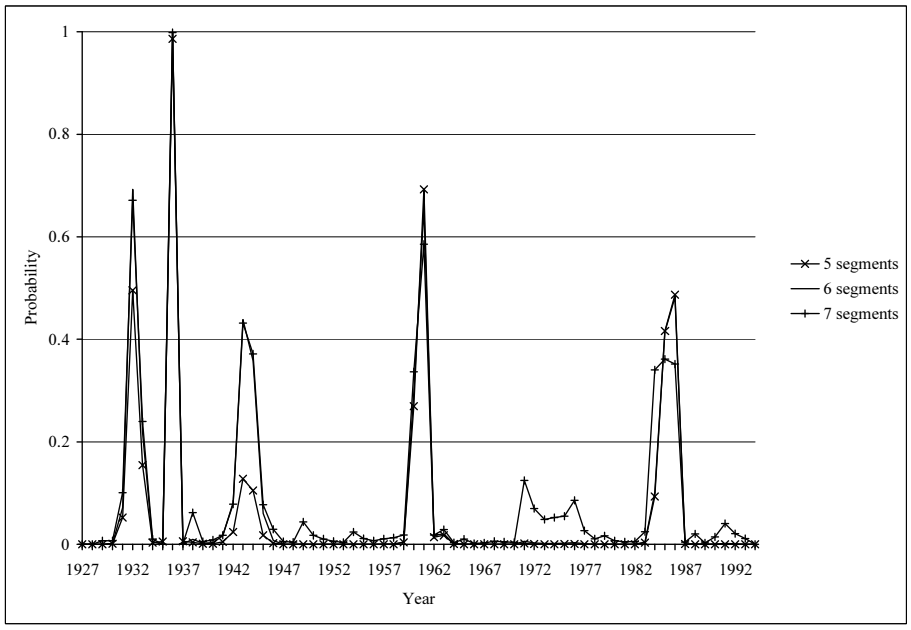

Figure 5: Corsican pine: Posterior change-point probabilities.

Another summary of all the possible segmentations is given by the change-point profile (the change-point posterior probabilities are cumulated for all change-point ranks; see (14)) also computed by the smoothing-type forward-backward algorithm. It should be recalled that the three change-point profiles shown in Figure 5 are computed by a single application of the smoothing-type forward-backward algorithm for 7 segments due to the transdimensional character of this algorithm. There is an agreement between the possible segmentations in 5, 6 and 7 segments concerning the 4 change points (1931 $\rightarrow$ 1932, 1935 $\rightarrow$ 1936, 1960 $\rightarrow$ 1961, 1985 $\rightarrow$ 1986) while the change point (1942 $\rightarrow$ 1943) is simply predicted in the case of 5 segments (posterior probability of 0.71 for the change point centered on 1931 $\rightarrow$ 1932 and 0.28 for the change point centered on 1942 $\rightarrow$ 1943). The noise component particularly in the range 1971 $\rightarrow$ 1976 is an indicator of overparameterization of the 7-segment model.

Another way to explore the behavior of multiple change-point models consists of generating a large number of segmentations using the forward-backward algorithm for sampling segmentations (Section 6) and then extracting empirical segment length distributions. The segment length distributions for $J = 5, 6, 7$

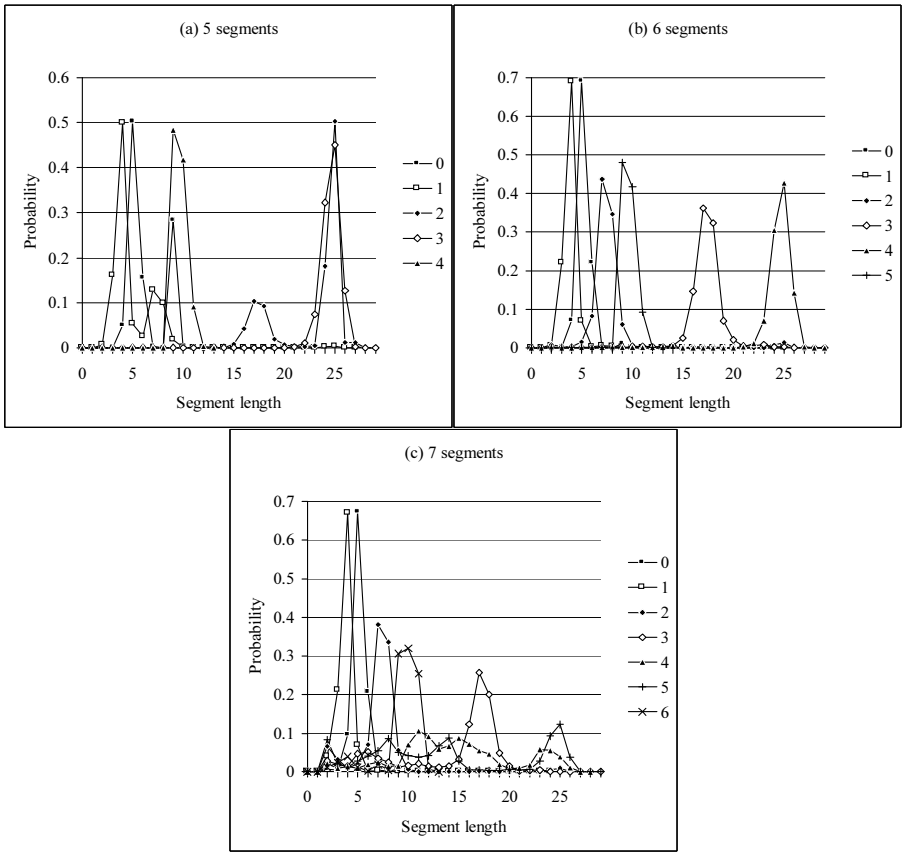

Figure 6: Corsican pine: Segment length distributions.

extracted on the basis of 10000 segmentations are shown in Figure 6. For 5 segments (Figure 6a), the alternative segmentations corresponding to the merge, shift and split transform translate into bimodal distributions for segments 0, 1 and 2. For 6 segments (Figure 6b), all the segment length distributions are unimodal with small dispersions, while for 7 segments (Figure 6c), length distributions for segments 3, 4 and 5 are strongly dispersed.

Sub-optimal segmentations may be helpful to compare alternative multiple change-point models (Gaussian model with piecewise constant variances or global variance in this example). For 5 segments, there is an agreement for 3 change-point locations (1931 → 1932, 1935 → 1936, 1960 → 1961) between the two Gaussian models. The only difference lies in a shift of one year for the last change point (1984 → 1985 for the change in the mean model instead of 1985 → 1986 for the change in the mean and variance model). The most probable segmentation for the change in the mean model is the second most probable segmentation for the change in the mean and variance model (with a likelihood ratio of 0.85 with reference to the most probable segmentation; see above) and conversely, the most probable segmentation for the change in the mean and variance model is the third most probable segmentation for the change in the mean model (with a likelihood ratio of 0.67 with reference to the most probable

segmentation). For 6 segments, there is an agreement for 4 change-point locations (1931 → 1932, 1935 → 1936, 1960 → 1961, 1985 → 1986) between the two Gaussian models. The remaining change point is 1983 → 1984 for the change in the mean model and 1942 → 1943 for the change in the mean and variance model. The 1983 → 1984 change point identified with the 6-segment change in the mean model becomes an optimal change point with the 7-segment change in the mean and variance model; see Figure 1. The most probable segmentation for the change in the mean model is the 80th most probable segmentation for the change in the mean and variance model (with a likelihood ratio of 0.009 with reference to the most probable segmentation) and conversely, the most probable segmentation for the change in the mean and variance model is the 16th most probable segmentation for the change in the mean model (with a likelihood ratio of 0.19 with reference to the most probable segmentation). Hence, in the case of 5 segments, the segmentation difference between the two Gaussian models corresponds to uncertainty concerning the location of a change point while, in the case of 6 segments, the segmentation difference between the two Gaussian models is a true structural difference.

7.2 Apple tree branching structure

Twenty apple trees (*Malus domestica* Borkh, *Rosaceae*) cultivar reinet, grafted on rootstock M.7, were planted in a field near Montpellier (south of France) and cut back to one bud one year after transplantation. The trees were then allowed to develop without pruning. The location of the immediate offspring shoots (offspring shoots developed without delay with respect to the parent node establishment date) was recorded after one year of growth while the location of 1-year-delayed offspring shoots was recorded after two years of growth. Among these 1-year-delayed offspring shoots, short shoots, long shoots and flowering shoots were distinguished. The first annual shoot of the trunks was described by node from the top to the base where, for each node, the type of axillary production chosen among latent bud (0), 1-year-delayed short shoot (1), 1-year-delayed long shoot (2), 1-year-delayed flowering shoot (3), and immediate shoot (4) was recorded. The branching structure of the first annual shoot of the trunk after two years of growth is assumed to be a good predictor of the adult tree structure. The branching structure of these apple tree trunks were previously analyzed using hidden semi-Markov chains (Guédon *et al.*, 2001; Guédon, 2003). To present the most probable segmentation in 6 segments of the selected individual, we adopted the following convention: the upper row consists of the most probable segmentation while the lower row consists of the observed sequence. The symbol '\' indicates continuation of the bivariate sequence defined in this way on the next two rows.

```
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3\
2 2 2 3 0 3 3 3 0 3 0 3 3 0 0 3 0 4 4 4 4 4 4 4 4 4 4 0 4 0 0 0 0 0\
3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5
0 0 0 0 0 0 1 2 1 1 0 1 1 0 2 2 1 2 1 1 0 2 0 0 0 0 0 0 0 0 0 0 0 0
```

We estimated multinomial multiple change-point models on the basis of this sequence for $J = 5, 6, 7$. The results are fairly similar to those obtained for the Corsican pine sequence. In the case of 5 segments, a supplementary change point is predicted between ranks 2 and 3 by a single split, shift and merge,

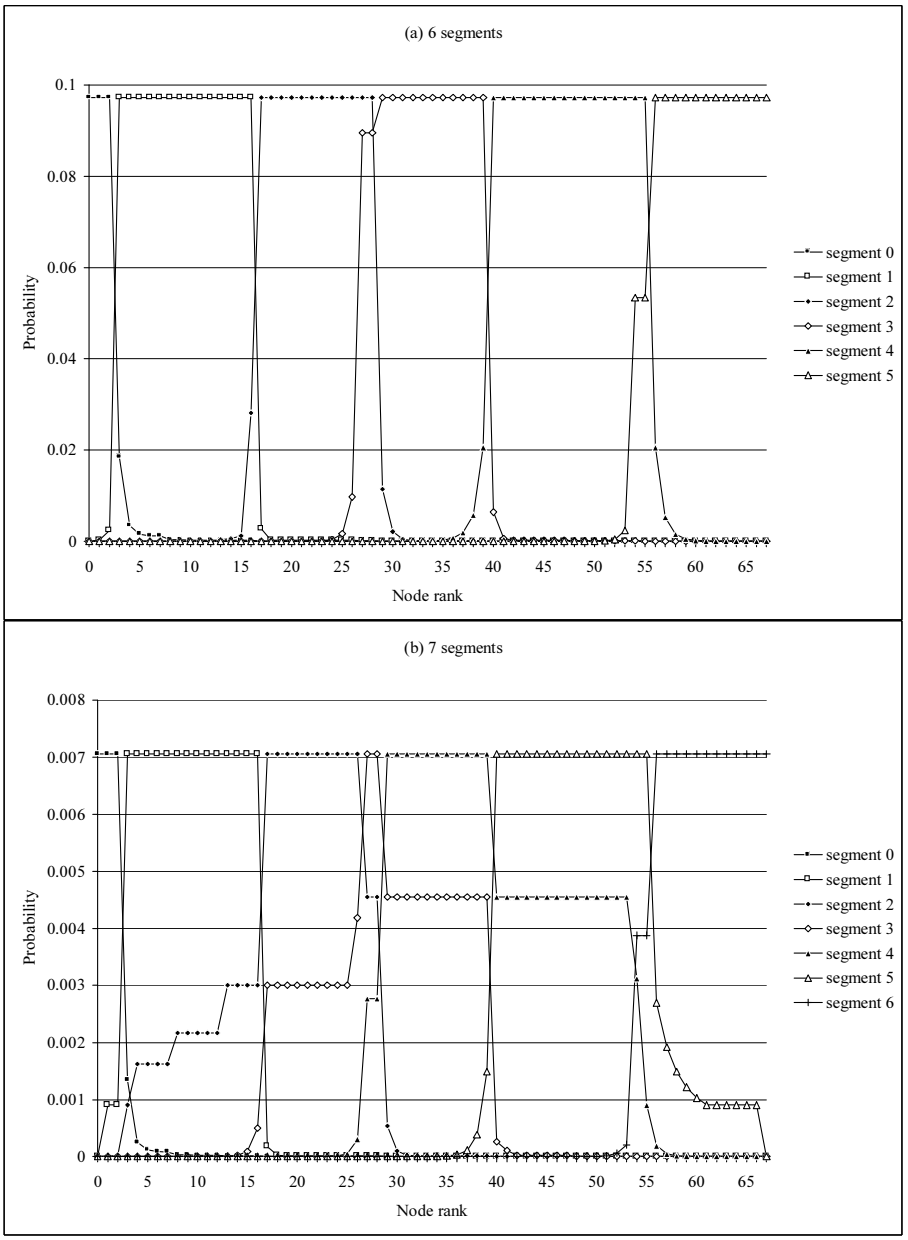

Figure 7: Apple tree: Segment profiles computed by the forward-backward dynamic programming algorithm.

with posterior probability of 0.005 to be compared with the posterior probability of 0.114 of the most probable segmentation (segment profiles not shown). The 6-segment model seems to provide an adequate compromise between the fit to the data and the parsimony of the model (Figure 7a) while the 7-segment model is clearly overparameterized (Figure 7b). The main interest of this example is illustrated in Figure 8 by the change-point profiles, computed by the

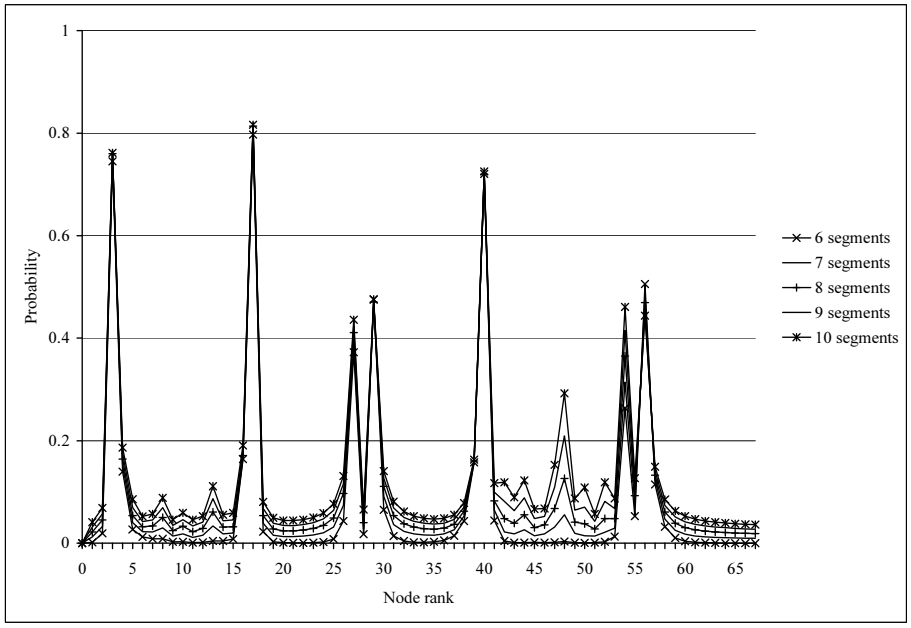

Figure 8: Apple tree: Posterior change-point probabilities.

smoothing-type forward-backward algorithm, where the change-point posterior probabilities are cumulated for all change-point ranks; see (14). These change-point profiles show a roughly uniform noise component from 7 segments up to 10 segments. This means that the five change points are strongly defined (with two possible locations at two nodes apart for change points 3 and 5).

8 Concluding remarks

The dynamic programming and smoothing-type forward-backward algorithms (Sections 3 and 5) can be equally well designed as backward-forward algorithms. The dynamic programming algorithm for computing the top L segmentations (Section 4) can be designed with an initial backward recursion followed by a forward tracking pass to retrieve the top L most probable segmentations. The algorithm for sampling segmentations (Section 6) can be designed with the initial backward recursion (13) followed by a forward pass for sampling segmentations. One of the advantages of the proposed algorithms is their impartiality since they do not require subjective choice of prior, for instance for the segment length, in order to compute posterior change-point probabilities.

Exploring the segmentation space may consist in extracting segmentations, whose probabilities are not negligible, that differ from the most probable segmentation and from other segmentations previously extracted on the basis of the same criteria. These structural differences between segmentations are naturally highlighted in the change-point and segment profiles computed by the forward-backward dynamic programming algorithm. In particular, the merge, shift and split (or split, shift and merge) transform is highlighted by the seg-

ment profiles computed by forward-backward dynamic programming algorithm. This behavior is specific to change-point models where the change points and the parameters attached to each segment are jointly estimated. In particular, such behavior cannot be observed with hidden Markovian models.

Our aim here is to propose possible uses of the outputs of the proposed algorithms for model selection. Our focus will be mainly on the overparameterization of the multiple change-point models for which different indicators can be proposed. This point of discussion is rather prospective and the proposal of new model selection criteria for determining the number of change points is beyond the scope of this paper. We computed the modified BIC values (Zhang and Siegmund, 2007) for the different examples for purposes of comparison. The log-likelihood of the optimal J-segment model for the sequence $x_0, \ldots, x_{T-1}$ is given by

$$\begin{aligned}
&\log f_J\left(x_0,\ldots,x_{T-1};\widehat{\tau}_1,\ldots,\widehat{\tau}_{J-1},\hat{\theta}\right)\\
=&\max_{0<\tau_1<\cdots<\tau_{J-1}<T}\sum_{j=0}^{J-1}\log f\left(x_{\tau_j},\ldots,x_{\tau_{j+1}-1};\widehat{\theta}_j\right)\\
=&\alpha_{J-1}\left(T-1\right).
\end{aligned}$$

It should be recalled that the BIC usage is not theoretically justified for change-point models where the likelihood functions do not satisfy the required regularity conditions and BIC thus tends to overestimate the number of change points. Zhang and Siegmund (2007) proposed a modified BIC in the case of the Gaussian change in the mean model. Transposed to the change in the mean and variance model, this criterion is given by

$$\begin{aligned}
\mathrm{mBIC}_J=&2\log f_J\left(x_0,\ldots,x_{T-1};\widehat{\tau}_1,\ldots,\widehat{\tau}_{J-1},\hat{\theta}\right)-(3J-1)\log T\\
&-\sum_{j=0}^{J-1}\log\left(\widehat{\tau}_{j+1}-\widehat{\tau}_j\right),
\end{aligned}$$

where

$$\begin{aligned}
\min_{0<\tau_1<\cdots<\tau_{J-1}<T}\sum_{j=0}^{J-1}\log\left(\widehat{\tau}_{j+1}-\widehat{\tau}_j\right)=&\log\left(T-J+1\right)\\
\approx&J\log T-(J-1)\log T \qquad \text{if } J\ll T,\\
\max_{0<\tau_1<\cdots<\tau_{J-1}<T}\sum_{j=0}^{J-1}\log\left(\widehat{\tau}_{j+1}-\widehat{\tau}_j\right)=&J\log\frac{T}{J}\\
=&J\log T-J\log J.
\end{aligned}$$

Hence each change point contributes between 1 and 2 dimensions to the penalty term (instead of systematically 1 dimension for each mean or variance parameter) and this penalty term is maximized when the change points are evenly spaced.

The posterior probability of the J-segment model, given by

$$P\left(\mathcal{M}_J|x_0,\ldots,x_{T-1}\right)=\frac{\exp\left(\frac{1}{2}\Delta\text{mBIC}_J\right)}{\sum_{K=1}^{J_{\max}}\exp\left(\frac{1}{2}\Delta\text{mBIC}_K\right)},$$

with

$$\Delta\text{mBIC}_J=\text{mBIC}_J-\max_K\text{mBIC}_K,$$

can be interpreted as the weight of evidence in favor of the J-segment model (among the $J_{\max}$ models).

Both the posterior probability of the most probable segmentation $\exp\left\{\alpha_{J-1}\left(T-1\right)-\log f\left(x_0,\ldots,x_{T-1};J\right)\right\}$ and the sum over time of the entropies computed from the posterior change-point distributions (see (15)), referred to as change-point entropy in the following, can be used to determine a limit for model overparameterization. It should be noted that the posterior probability of the most probable segmentation and the change-point entropy for $K=2,\ldots,J$ can be globally computed by the classic forward dynamic programming algorithm and the smoothing-type forward-backward algorithm for J segments.

Table 1: Corsican pine: indicators extracted for different Gaussian change in the mean and variance models - P_J abbreviates $P\left(\mathcal{M}_J|x_0,\ldots,x_{T-1}\right)$ -.

J	$2\log f_J$	posterior probability	change-point entropy	free param.	mBIC_J	P_J
1	-594.17	1		2	-606.83	0
2	-550.51	0.568	1.75	5	-577.89	0
3	-518.12	0.165	4.99	8	-559.76	0
4	-470.22	0.311	3.89	11	-527.47	0.01
5	-449.38	0.17	5.69	14	-520.09	0.12
6	-431.39	0.098	6.99	17	-516.37	0.77
7	-422.88	0.012	11.27	20	-521.12	0.07
8	-410.48	0.013	13.27	23	-523.62	0.02
9	-398.56	0.015	14.43	26	-524.86	0.01
10	-389.35	0.008	16.17	29	-528.3	0

For the Corsican pine, the models are overparameterized from seven segments inclusive. This is quite obvious regarding the posterior probability of the optimal segmentation and the change-point entropy as a function of the number of segments; see Table 1. The proposed indicators exhibit changes of values of high magnitude around the optimal number of segments. The behavior of the proposed indicators is similar for the apple tree example (results not shown). This behavior may be exploited to select a model among different models with close values for a penalized likelihood criterion. These indicators provide an intuitive translation (and possibly a graphical translation with the change-point and segment profiles) of the under- or overparameterization of a change-point model. A key difference between the proposed indicators and the usual penalized likelihood criteria is that these indicators rely on all the possible segmentations for a given number of segments J while the penalized likelihood criteria rely solely on the most probable segmentation. The formal relations

between these two complementary approaches may be an interesting topic for future investigation.

Methods for exploring the segmentation space for multiple change-point models are fully implemented in the VPlants software which is freely available at http://www-sop.inria.fr/virtualplants.

Acknowledgments

The author thanks Yves Caraglio and Céline Méredieu for providing the Corsican pine data and Evelyne Costes for providing the apple tree data.

References

Auger, I. E. and Lawrence, C. E. (1989). Algorithms for the optimal identification of segment neighborhoods. *Bulletin of Mathematical Biology* 51, 39-54.

Braun, J. V. and Müller, H.-G. (1998). Statistical methods for DNA sequence segmentation. *Statistical Science* 55(2), 142-162.

Dobigeon, N., Tourneret, J.-Y. and Scargle, J. D. (2007). Joint segmentation of multivariate astronomical time series: Bayesian sampling with a hierarchical model. *IEEE Transactions on Signal Processing* 55(2), 414-423.

Fearnhead, P. (2006). Exact and efficient Bayesian inference for multiple changepoint problems. *Statistics and Computing* 16(2), 203-213.

Guédon, Y. (2003). Estimating hidden semi-Markov chains from discrete sequences. *Journal of Computational and Graphical Statistics* 12(3), 604-639.

Guédon, Y. (2007). Exploring the state sequence space for hidden Markov and semi-Markov chains. *Computational Statistics & Data Analysis* 51(5), 2379-2409.

Guédon, Y., Barthélémy, D., Caraglio, Y. and Costes, E. (2001). Pattern analysis in branching and axillary flowering sequences. *Journal of Theoretical Biology* 212(4), 481-520.

Guédon, Y., Caraglio, Y., Heuret, P., Lebarbier, E. and Meredieu, C. (2007). Analyzing growth components in trees. *Journal of Theoretical Biology* 248(3), 418-447.

Hawkins, D. M. (2001). Fitting multiple change-point models to data. *Computational Statistics & Data Analysis* 37(3), 323-341.

Hawkins, D. M. and Zamba, K. D. (2005). Statistical process control for shifts in mean or variance using a changepoint formulation. *Technometrics* 47(2), 164-173.

Lavielle, M. (2005). Using penalized contrasts for the change-point problem. *Signal Processing* 85(8), 1501-1510.

Lebarbier, E. (2005). Detecting multiple change-points in the mean of Gaussian process by model selection. *Signal Processing* 85(4), 717-736.

Levinson, S. E. (1986). Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language* 1, 29-45.

Liu, J. S. and Lawrence, C. E. (1999). Bayesian inference on biopolymer models. *Bioinformatics* 15, 38-52.

Picard, F., Robin, S., Lavielle, M., Vaisse, C. and Daudin, J.-J. (2005). A statistical approach for array CGH data analysis. *BMC Bioinformatics* 6(27).

Zhang, N. R. and Siegmund, D. O. (2007). A modified Bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics* 63, 22-32.

Appendix A: Pseudo-code of the forward-backward dynamic programming algorithm

The following convention is adopted in the presentation of this pseudo-code: The operator ':=' denotes the assignment of a value to a variable (or the initialization of a variable with a value) and the working variables SegmentLogLikelihood(t) and Output$_j$ (t) are introduced for this implementation. The expression ($t < T-1$? $J-2 : J-1$) is a shorthand for **if** $t < T-1$ **then** $J-2$ **else** $J-1$.

Forward recursion

for $t := 0$ to $T-1$ **do**

log-likelihood computation for segments ending at time t (Gaussian model)
SumSquaredDeviation := 0
Sum := x_t
SegmentLogLikelihood(t) := $-\infty$
for $u := t-1$ to 0 **do**
SumSquaredDeviation := SumSquaredDeviation $+(t-u)/(t-u+1)$
$\times\{x_u -$ Sum$/(t-u)\}^2$
Sum := Sum $+x_u$
SegmentLogLikelihood(u) := $-(t-u+1)/2\,[\log\{$SumSquaredDeviation
$/(t-u+1)\} + \log 2\pi + 1]$
end for

for $j := 0$ to $J-1$ **do**
$\alpha_j(t) := -\infty$
end for

for $j :=$ max$\{0, J-(T-t)\}$ to min$\{(t < T-1\,?\,J-2 : J-1), t\}$ **do**
if $j = 0$ **then**
$\alpha_0(t) :=$ SegmentLogLikelihood(0)
else $\{j > 0\}$
for $u := t$ to j **do**
if SegmentLogLikelihood(u) $+ \alpha_{j-1}(u-1) > \alpha_j(t)$ **then**
$\alpha_j(t) :=$ SegmentLogLikelihood(u) $+ \alpha_{j-1}(u-1)$
end if
end for
end if
end for
end for

In a first step, the log-likelihoods for segments ending at time t, SegmentLogLikelihood(u) are computed for each segment length. Then in a second step, the quantities $\alpha_j\,(t)$ are computed for each segment j. The quantities SegmentLogLikelihood(u) should be stored for each time u while the quantities $\alpha_j\,(t)$ should be stored for each time t and each segment j.

Backward recursion

for $t := T-1$ to 0 **do**

log-likelihood computation for segments beginning at time t (Gaussian model)

SumSquaredDeviation := 0
Sum := x_t
SegmentLogLikelihood(t) := $-\infty$
for $u := t+1$ to $T-1$ **do**
  SumSquaredDeviation := SumSquaredDeviation $+(u-t)/(u-t+1)$
    $\times\{\mathrm{x}_u - \mathrm{Sum}/(u-t)\}^2$
  Sum := Sum $+x_u$
  SegmentLogLikelihood(u) := $-(u-t+1)/2\,[\log\{$SumSquaredDeviation
    $/(u-t+1)\} + \log 2\pi + 1]$
end for

for $j := 0$ to $J-1$ **do**
  Output$_j(t) := -\infty$
end for

for $j := \max\{(t=0\,?\,0:1), J-(T-t)\}$ to $\min(J-1, t)$ **do**
  if $j < J-1$ **then**
    $\beta_j(t) := -\infty$
    for $u := T-(J-j)$ to t **do**
      if SegmentLogLikelihood(u) $+\beta_{j+1}(u+1) > \beta_j(t)$ **then**
        $\beta_j(t)$:= SegmentLogLikelihood(u) $+\beta_{j+1}(u+1)$
      end if

      for segment profiles
      if $u > t$ **then**
        if $t = 0$ **then**
          if $\beta_0(0) >$ Output$_0(u)$ **then**
            Output$_0(u) := \beta_0(0)$
          end if
        else $\{t > 0\}$
          if $\alpha_{j-1}(t-1) + \beta_j(t) >$ Output$_j(u)$ **then**
            Output$_j(u) := \alpha_{j-1}(t-1) + \beta_j(t)$
          end if
        end if
      end if
    end for

  else $\{j = J-1\}$
    $\beta_{J-1}(t)$:= SegmentLogLikelihood($T-1$)

    for segment profiles
    for $u := T-1$ to $t+1$ **do**
      if $\alpha_{J-2}(t-1) + \beta_{J-1}(t) >$ Output$_{J-1}(u)$ **then**
        Output$_{J-1}(u) := \alpha_{J-2}(t-1) + \beta_{J-1}(t)$
      end if
    end for

```
      end if

      if t = 0 then
        Output_0(0) := β_0(0)
      else {t > 0}
        Output_j(t) := α_{j-1}(t-1) + β_j(t)
      end if
    end for
  end for
```

In a first step, the log-likelihoods for segments beginning at time t, SegmentLogLikelihood(u) are computed for each segment length. Then in a second step, the quantities $\beta_j(t)$ are computed for each segment j and the quantities $\gamma_j(t)$ are extracted in the case of the computation of segment profiles. The quantities SegmentLogLikelihood(u) should be stored for each time u while the quantities $\beta_j(t)$ and $\text{Output}_j(t)$ should be stored for each time t and each segment j.

Appendix B: Pseudo-code of the forward dynamic programming algorithm for computing the top L most probable segmentations in J segments

For each time t, each segment j and each rank n, two backpointers must be recorded, the first $\text{Length}_j^n(t)$ giving the optimal length of the segment and the second $\text{Rank}_j^n(t)$ giving the associated rank.

Forward recursion

```
for t := 0 to T − 1 do

  log-likelihood computation for segments ending at time t (see Appendix A)
  L_0(t) := 1
  for j := 1 to min{(t < T − 1 ? J − 2 : J − 1), t} do
    L_j(t) := L_{j-1}(t)(t − j + 1)/j
  end for
  for j := 1 to min{(t < T − 1 ? J − 2 : J − 1), t} do
    if L_j(t) > L then
      L_j(t) = L
    end if
  end for

  for j := max{0, J − (T − t)} to min{(t < T − 1 ? J − 2 : J − 1), t} do
    if j = 0 then
      α_0^1(t) := SegmentLogLikelihood(0)
      Length_0^1(t) := t + 1

    else {j > 0}
      for u := t to j do
        r(u) := 1
```

```
      end for

      for n := 1 to L_j(t) do
        α_j^n(t) := −∞
        for u := t to j do
          if SegmentLogLikelihood(u) + α_{j−1}^{r(u)}(u − 1) > α_j^n(t) then
            α_j^n(t) := SegmentLogLikelihood(u) + α_{j−1}^{r(u)}(u − 1)
            Length_j^n(t) := t − u + 1
            Rank_j^n(t) := r(u)
          end if
        end for

        r{t− Length_j^n(t) + 1} := r{t− Length_j^n(t) + 1} + 1
      end for
    end if

    for n := L_j(t) + 1 to L do
      α_j^n(t) := −∞
    end for
  end for
end for
```

In a first step, the log-likelihoods for segments ending at time t, SegmentLogLikelihood(u) are computed for each segment length; see Appendix A. Then in a second step, the quantities $\alpha_j^n(t)$ are computed for each segment j and each rank n. The increasing number of segmentations for small values of t is taken into account by initializing at $-\infty$ $\alpha_j^n(t)$ for each segment j and for $n = L_j(t)+1, \ldots, L$. The quantities SegmentLogLikelihood(u) should be stored for each time u while the quantities $\alpha_j^n(t)$ and the backpointers $\text{Length}_j^n(t)$ and $\text{Rank}_j^n(t)$ should be stored for each time t, each segment j and each rank n.

Backtracking

```
for n := 1 to L_{J−1}(T − 1) do
  t := T − 1
  m := n

  for j := J − 1 to 0 do
    for u := t to t− Length_j^m(t) + 1 do
      s_u^n := j
    end for
    if j > 0 then
      q := Rank_j^m(t)
      t := t− Length_j^m(t)
      m := q
    end if
  end for
end for
```

Appendix C: Pseudo-code of the smoothing-type forward-backward algorithm

The working variables Norm(t), ForwardNorm(t) and BackwardNorm(t) are introduced for this implementation.

Forward recursion

for $t := 0$ to $T-1$ **do**

log-likelihood computation for segments ending at time t (see Appendix A): values stored in the quantities (SegmentLikelihood$(u); u = 0, \ldots, t)$
SegmentLikelihood(t) := exp{SegmentLikelihood(t)}
SegmentNorm := 0
for $u := t-1$ to 0 **do**
SegmentNorm := SegmentNorm + Norm(u)
SegmentLikelihood(u) := exp{SegmentLikelihood(u)− SegmentNorm}
end for

for $j := 0$ to $J-1$ **do**
$F_j(t) := 0$
end for
Norm(t) := 0

for $j := \max\{0, J-(T-t)\}$ to $\min\{(t < T-1\,?\,J-2 : J-1), t\}$ **do**
if $j = 0$ **then**
$F_0(t)$:= SegmentLikelihood(0)
else $\{j > 0\}$
for $u := t$ to j **do**
$F_j(t) := F_j(t)+$ SegmentLikelihood$(u) F_{j-1}(u-1)$
end for
end if
Norm(t) := Norm(t) + $F_j(t)$
end for

for $j := \max\{0, J-(T-t)\}$ to $\min\{(t < T-1\,?\,J-2 : J-1), t\}$ **do**
$F_j(t) := F_j(t)/$Norm(t)
end for

Norm(t) := log{Norm(t)}
ForwardNorm(t) := SegmentNorm + Norm(t)
end for

$\log f(x_0, \ldots, x_{T-1}; J) = \log F_{J-1}(T-1)+$ ForwardNorm$(T-1)$

The minimum segment index should be 0 instead of $\max\{0, J-(T-t)\}$ for the transdimensional generalization.

In a first step, the log-likelihoods for segments ending at time t, SegmentLikelihood(u) are computed for each segment length; see Appendix A. Then in a second step, the quantities $F_j(t)$ are computed for each segment j.

The variable SegmentNorm is used to compute $\sum_{v=u}^{t-1} \log N_v$ for each $u < t$. The quantities $\sum_{u=0}^{t} \log N_u$ are stored in the auxiliary quantity ForwardNorm(t) for each time t for use in the backward recursion. The quantities SegmentLikelihood(u) should be stored for each time u while the quantities $F_j\,(t)$ should be stored for each time t and each segment j.

Backward recursion
for $t := T-1$ to 0 **do**

log-likelihood computation for segments beginning at time t (see Appendix A): values stored in the quantities (SegmentLikelihood(u); $u = t, \ldots, T-1$)

SegmentLikelihood(t) := exp{SegmentLikelihood(t)}
SegmentNorm := 0
for $u := t+1$ to $T-1$ **do**
  SegmentNorm := SegmentNorm + Norm(u)
  SegmentLikelihood(u) := exp{SegmentLikelihood(u)− SegmentNorm}
end for

for $j := 0$ to $J-1$ **do**
  $B_j(t) := 0$
  Output$_j(t) := 0$
end for
Norm(t) := 0

for $j := \max\{(t = 0\,?\,0 : 1), J-(T-t)\}$ to $\min(J-1, t)$ **do**
  if $j < J-1$ **then**
    for $u := t$ to $T-(J-j)$ **do**
      $B_j(t) := B_j(t)+$ SegmentLikelihood(u)$B_{j+1}(u+1)$
    end for
  else $\{j = J-1\}$
    $B_{J-1}(t) :=$ SegmentLikelihood($T-1$)
  end if
  Norm(t) := Norm(t) + $B_j(t)$
end for

for $j := \max\{(t = 0\,?\,0 : 1), J-(T-t)\}$ to $\min(J-1, t)$ **do**
  $B_j(t) := B_j(t)/$Norm(t)
end for

Norm(t) := log{Norm(t)}
BackwardNorm(t) := SegmentNorm + Norm(t)

for segment profiles
if $t < T-1$ **then**
  for $j := \max\{0, J-(T-t)\}$ to $\min(J-1, t)$ **do**
    Output$_j(t)$:= Output$_j(t+1)$
    if $j < J-1$ **then**
      Output$_j(t)$:= Output$_j(t) + F_j(t)B_{j+1}(t+1)$ SequenceNorm

```
        end if
        if j > 0 then
          Output_j(t) := Output_j(t) − F_{j−1}(t)B_j(t + 1) SequenceNorm
        end if
      end for
    else {t = T − 1}
      Output_{J−1}(T − 1) := 1
    end if

    if t > 0 then
      SequenceNorm = exp{ForwardNorm(t − 1)+ BackwardNorm(t)
                         − log f(x_0, ..., x_{T−1}; J)}
    end if

    for change-point profiles
    if t = 0 then
      Output_0(0) := 1
    else {t > 0}
      for j := max{1, J − (T − t)} to min(J − 1, t) do
        Output_j(t) := F_{j−1}(t − 1)B_j(t) SequenceNorm
      end for
    end if
  end for
```

The maximum segment index should be $J-1$ instead of $\min(J-1,t)$ for the transdimensional generalization.

In a first step, the log-likelihoods for segments beginning at time t, SegmentLikelihood(u) are computed for each segment length; see Appendix A. Then in a second step, the quantities $B_j(t)$ are computed for each segment j. The variable SegmentNorm is used to compute $\sum_{v=t+1}^{u} \log M_v$ for each $u > t$. The quantities SegmentLikelihood(u) should be stored for each time u while the quantities $B_j(t)$ and $\text{Output}_j(t)$ should be stored for each time t and each segment j and only the current quantity BackwardNorm(t) should be stored.

Contents

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399