Stat Comput
DOI 10.1007/s11222-011-9248-x

On-line changepoint detection and parameter estimation with application to genomic data

François Caron · Arnaud Doucet · Raphael Gottardo

Received: 2 June 2010 / Accepted: 15 March 2011

Abstract An efficient on-line changepoint detection algorithm for an important class of Bayesian product partition models has been recently proposed by Fearnhead and Liu (in J. R. Stat. Soc. B 69, 589–605, 2007). However a severe limitation of this algorithm is that it requires the knowledge of the static parameters of the model to infer the number of changepoints and their locations. We propose here an extension of this algorithm which allows us to estimate jointly on-line these static parameters using a recursive maximum likelihood estimation strategy. This particle filter type algorithm has a computational complexity which scales linearly both in the number of data and the number of particles. We demonstrate our methodology on a synthetic and two real-world datasets from RNA transcript analysis. On simulated data, it is shown that our approach outperforms standard techniques used in this context and hence has the potential to detect novel RNA transcripts.

Keywords Sequential Monte Carlo · Particle filtering · Changepoint models · Product partition models · Recursive parameter estimation · Tiling arrays

F. Caron (✉)
INRIA Bordeaux Sud-Ouest and Institut de Matématiques de Bordeaux, Université de Bordeaux, Talence, France
e-mail: Francois.Caron@inria.fr

A. Doucet
Departments of Statistics & Computer Science, University of British Columbia, Vancouver, BC, Canada

R. Gottardo
Vaccine and Infectious Disease and Public Health Sciences Divisions, Fred Hutchinson Cancer Research Center, 1100 Fairview Avenue North, M2-C200 PO Box 19024, Seattle, WA 98109-1024, USA

Published online: 06 April 2011

1 Introduction

Many time series, such as DNA sequences, stock prices or electricity load, exhibit temporal heterogeneity (Carlin et al. 1992; Fearnhead 2006; Johnson et al. 2003). In this context, a popular approach consists of segmenting the sequence of observations $z_1, z_2, \ldots, z_T$ by choosing a sequence of changepoint locations $0 < \tau_1 < \tau_2 < \cdots < \tau_m < T$ such that the observations are homogeneous within segments and heterogeneous across segments. We focus here on a Bayesian product partition changepoint model (Barry and Hartigan 1992) which assigns a joint prior distribution over the number and locations of change-points. We further assume that the observations across segments are statistically independent.

When the parameters of the changepoint model are known, exact inference about the number of changepoints and their locations can be performed using an algorithm of complexity T^2 (Fearnhead 2006). This is far too computationally expensive when analyzing real-world data sets with tens or hundreds of thousands observations as discussed in this paper. To reduce this computational complexity, a particle filtering algorithm for product partition models was proposed by Fearnhead and Liu (2007). It admits a computational complexity linear both in the number of data and the number of particles. We propose here an extension of this algorithm to infer jointly online the model parameters using a recursive maximum likelihood technique which builds upon the recent work by Poyiadjis et al. (2011). A standard Bayesian alternative would consist of assigning a prior to these parameters and to sample from the joint posterior distribution of the changepoints and the parameters using Markov chain Monte Carlo (MCMC) methods; see for example Stephens (1994), Chib (1998), De Iorio et al. (2005). However, MCMC techniques are far too computationally

expensive for huge data sets. Moreover standard MCMC strategies in this context update the changepoints and model parameters independently. This can lead to slow mixing Markov chains due to the strong correlations between these parameters.

We apply our algorithm to the detection of novel RNA transcripts from tiling arrays—an important problem that has received little attention from the statistics community. A common approach to address this problem is to combine probe measurements via a sliding window (SW) statistic computed over neighboring probes, and then to apply a thresholding on the resulting statistics to call transcript regions (Kapranov et al. 2002; Bertone et al. 2004; Cheng et al. 2005). A more sophisticated method has been proposed recently which models the underlying signal as piecewise constant (Huber et al. 2006) but it fixes the number of segments in advance and requires specifying a threshold in order to classify segments as transcript or non-transcript. The model proposed here addresses these pitfalls, builds on previous approaches used in gene expression analysis (Newton et al. 2001; Gottardo et al. 2003, 2006) and uses a mixture of normal and skew t distributions to model non-Gaussianity and classify segments as transcript or non-transcript, hence no thresholding is necessary.

The rest of this paper is organized as follows. In Sects. 2 and 3, we present our changepoint model and parameter estimation procedure, respectively. In Sect. 4, we apply our method to simulated and real-world data and compare it to the changepoint model of Huber et al. (2006) and the SW approach used by Cheng et al. (2005). Finally, in Sect. 5 we discuss possible extensions of our work. All the calculations are detailed in Appendices.

2 Statistical model

Assume we have a (possibly multi- dimensional) sequence of observations $\{z_t, t = 1, \ldots, T\}$ observed at consecutive points in time. We will denote by $z_{t_1:t_2} = \{z_{t_1}, \ldots, z_{t_2}\}$ the vector of observations from t_1 to t_2. Given a sequence of changepoint locations $0 < \tau_1 < \tau_2 < \cdots < \tau_m < T$, the sequence of observations $z_1, \ldots, z_T$ can be partitioned into $m + 1$ contiguous segments $z_{1:\tau_1}, z_{\tau_1+1:\tau_2}, \ldots, z_{\tau_m:T}$. A Bayesian changepoint model is defined by a joint distribution over the number of changepoints, their locations and the data. We consider a specific changepoint model which is such that the changepoint positions are modeled as a Markov process

$$\Pr(\text{“next changepoint at } t_2\text{”}|\text{“changepoint at } t_1\text{”}) = h(t_2 - t_1) \tag{1}$$

i.e. the probability of a changepoint only depends on the index distance to the previous one. This model is a special case of a product partition model for changepoints (Barry and Hartigan 1992; Fearnhead and Liu 2007). The function h can be any distribution with support the set of positive integers. Finally, we denote by H, $H(l) = \sum_{i=1}^{l} h(i)$ the cumulative distribution associated with h, which will be used in Sect. 3 when we describe our estimation procedure.

Additionally, we make the following conditional independence assumption (Barry and Hartigan 1992; Fearnhead and Liu 2007): “*given the position of a changepoint, the data before that changepoint is independent of the data after the changepoint*”. Finally for a segment of observations $z_{\tau_i+1:\tau_{i+1}}$, $i = 0, \ldots, m$, we assume that there are M possible models. To each model $r \in \{1, \ldots, M\}$ of prior probability $p(r)$ is associated a set of unknown parameters Ψ_r with some prior distribution π_r which is such that for $i \leq j$

$$P(i, j|r) := \int p(z_{i:j}|r, \Psi_r)\pi_r(\Psi_r)d\Psi_r \tag{2}$$

and the marginal likelihood

$$P(i, j) = \sum_{r=1}^{M} P(i, j|r)\, p\,(r) \tag{3}$$

can be computed analytically. This is typically possible when conjugate priors are used as in Fearnhead and Liu (2007), Fearnhead (2006), Chib (1998). If this marginal likelihood is not analytically tractable, Gaussian quadrature or Laplace approximation can be used to approximate (2); see e.g. Kass and Raftery (1995).

3 Changepoint detection and parameter estimation

3.1 Exact inference

3.1.1 Filtering recursions

We can rewrite this changepoint model as a state-space model (Chopin 2007; Fearnhead and Liu 2007). Let C_t denote the time of the most recent changepoint prior to t (with $C_t = 0$ if there has been no changepoint before time t). Conditional on $C_{t-1} = j$, either $C_t = j$, i.e. there is no changepoint at time t, or $C_t = t - 1$ if there is a changepoint. Using (1), it is easy to establish that

$$f(C_t = j|C_{t-1} = i) = \begin{cases} \frac{1-H(t-i-1)}{1-H(t-i-2)} & \text{if } j = i \\ \frac{H(t-i-1)-H(t-i-2)}{1-H(t-i-2)} & \text{if } j = t-1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

and

$$g(z_t|C_t = j, z_{1:t-1}) = \begin{cases} \frac{P(j,t)}{P(j,t-1)} & \text{if } j < t-1 \\ P(t-1, t) & \text{if } j = t-1 \end{cases}$$

where $P(\cdot,\cdot)$ is given by (3).

The so-called filtering distributions $p(C_t|z_{1:t})$ can be computed recursively in time using the recursions presented in Fearnhead and Liu (2007)

$$p(C_t|z_{1:t}) = \frac{\xi\,(C_t, z_{1:t})}{\sum_{i=0}^{t-1}\xi\,(C_t = i, z_{1:t})}$$

where

$$\xi(C_t, z_{1:t}) := g(z_t|C_t, z_{1:t-1})p(C_t|z_{1:t-1})$$

satisfies the following recursion

$$\xi(C_t, z_{1:t}) = \begin{cases} g(z_t|C_t = j, z_{1:t-1})f(C_t = j|C_{t-1} = j) \\ \quad \times p(C_{t-1} = j|z_{1:t-1}) \quad \text{if } j < t-1, \\ g(z_t|C_t = j, z_{1:t-1})\sum_{i=0}^{t-2} f(C_t = j|C_{t-1} = i) \\ \quad \times p(C_{t-1} = i|z_{1:t-1}) \quad \text{if } j = t-1. \end{cases} \tag{5}$$

Once the filtering distributions $p(C_t|z_{1:t})$ are stored for all $t = 1,\ldots,T$, we can simulate from the joint posterior distribution of the changepoints at time T (Chopin 2007; Fearnhead and Liu 2007), as follows.[1]

Simulation of changepoints from the joint posterior distribution

- Simulate τ_1 from $p(C_T|z_{1:T})$. Set $k = 1$.
- While $\tau_k > 0$
  - Sample τ_{k+1} proportionally to $f(C_{\tau_k+1} = \tau_k|C_{\tau_k})p(C_{\tau_k}|z_{1:\tau_k})$ and set $k = k+1$.

3.1.2 MAP recursions

An on-line Viterbi algorithm can be designed for calculating the maximum a posteriori (MAP) estimate of the changepoints and model labels (Fearnhead and Liu 2007). Let $\mathcal{M}_j$ be the event that given a changepoint at time j, the MAP estimate of changepoints and model has occurred prior to time j. Then for $t = 1,\ldots,n$, $j = 0,\ldots,t-1$ and $r = 1,\ldots,M$, we define

$$P_t(j,r) = \Pr(C_t = j, \text{model } r, \mathcal{M}_j, z_{1:t}),$$

$$P_t^{MAP} = \Pr(\text{Changepoint at } t, \mathcal{M}_t, z_{1:t}).$$

At time t, the MAP estimate $\widehat{c}_t$ of C_t and the current model are given by the values of j and r which maximise $P_t(j,r)$.

[1] For notational convenience, we use a reversed ordering of the changepoints compared to the definition in Sect. 1.

The following recursions can be established

$$P_t(j,r) = (1 - H(t-j-1))P(j,t|r)p(r)P_j^{MAP},$$
$$P_t^{MAP} = \max_{j,r}\left(\frac{P_t(j,r)h(t-j)}{1-H(t-j-1)}\right) \tag{6}$$

where $P(j,t|r)$ is the marginal distribution of the observations $z_{j+1:t}$ assumed to be in the same segment following the model r.

3.1.3 Recursive parameter estimation

The previous recursions assume that the transition probability $f(C_t|C_{t-1})$ and the conditional predictive density $g(z_t|C_t, z_{1:t-1})$ are known. However they usually depend on some parameters $\boldsymbol{\theta}$ in $\mathbb{R}^{n_\theta}$ which need to be estimated from the data. We propose here a recursive maximum likelihood approach. We introduce a subscript $\boldsymbol{\theta}$ to emphasize the dependence on parameters $\boldsymbol{\theta}$ of the filtering density $p_{\boldsymbol{\theta}}(C_t|z_{1:t})$, the transition probability $f_{\boldsymbol{\theta}}(C_t|C_{t-1})$, the conditional predictive density $g_{\boldsymbol{\theta}}(z_t|C_t, z_{1:t-1})$ and $\xi_{\boldsymbol{\theta}}(C_t, z_{1:t}) = g_{\boldsymbol{\theta}}(z_t|C_t, z_{1:t-1})p_{\boldsymbol{\theta}}(C_t|z_{1:t-1})$. These quantities are assumed to be continuously differentiable with respect to $\boldsymbol{\theta}$.

The log-likelihood of the data $z_{1:t}$ is given by

$$l_t(\boldsymbol{\theta}) = \log p_{\boldsymbol{\theta}}(z_1) + \sum_{k=2}^{t}\log p_{\boldsymbol{\theta}}(z_k|z_{1:k-1}) \tag{7}$$

where

$$p_{\boldsymbol{\theta}}(z_t|z_{1:t-1}) = \sum_{j=0}^{t-1}\xi_{\boldsymbol{\theta}}(C_t = j, z_{1:t}). \tag{8}$$

As $t \to \infty$, we have

$$\lim_{t\to\infty}\frac{l_t(\boldsymbol{\theta})}{t} = l(\boldsymbol{\theta}).$$

This follows from the fact that (1), (21) and (22) define an (asymptotically) stationary process with 'good' mixing properties. Moreover, $l(\boldsymbol{\theta})$ admits the true parameter $\boldsymbol{\theta}^*$ as a global maximum. To find a local maximum of $l(\boldsymbol{\theta})$, we use a stochastic approximation algorithm (Benveniste et al. 1990)

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \gamma_t \nabla \log p_{\boldsymbol{\theta}_{1:t-1}}(z_t|z_{1:t-1}) \tag{9}$$

where the stepsize sequence $\{\gamma_t\}$ is a positive non-increasing sequence such that $\sum\gamma_t = \infty$ and $\sum\gamma_t^2 < \infty$ whereas $\nabla \log p_{\boldsymbol{\theta}_{1:t-1}}(z_t|z_{1:t-1})$ is the gradient of the predictive log-likelihood w.r.t. $\boldsymbol{\theta}$. The subscript $\boldsymbol{\theta}_{1:t-1}$ indicates that this gradient is computed using the filtering recursions updated with $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ at time $k+1$. Under regularity conditions (Benveniste et al. 1990), it can be shown that $\boldsymbol{\theta}_t$ will converge to a

local maximum of $l(\boldsymbol{\theta})$. To improve the convergence rate of this algorithm, we can also use a Newton or quasi-Newton stochastic gradient algorithm by computing the Hessian of the log-likelihood; see Poyiadjis et al. (2011) for an application of this approach in a general state-space model context.

To compute the gradient term appearing in (9), we note that

$$\nabla \log p_{\boldsymbol{\theta}}(z_t|z_{1:t-1}) = \frac{\nabla p_{\boldsymbol{\theta}}(z_t|z_{1:t-1})}{p_{\boldsymbol{\theta}}(z_t|z_{1:t-1})} = \frac{\sum_{j=0}^{t-1} \nabla \xi_{\boldsymbol{\theta}}(C_t = j, z_{1:t})}{\sum_{j=0}^{t-1} \xi_{\boldsymbol{\theta}}(C_t = j, z_{1:t})}. \tag{10}$$

By taking the derivative of $p_{\boldsymbol{\theta}}(C_t|z_{1:t})$ with respect to $\boldsymbol{\theta}$, we obtain

$$\nabla p_{\boldsymbol{\theta}}(C_t|z_{1:t}) = \frac{\nabla \xi_{\boldsymbol{\theta}}(C_t, z_{1:t})}{\sum_{i=0}^{t-1} \xi_{\boldsymbol{\theta}}(C_t = i, z_{1:t})} - p_{\boldsymbol{\theta}}(C_t|z_{1:t}) \frac{\sum_{i=0}^{t-1} \nabla \xi_{\boldsymbol{\theta}}(C_t = i, z_{1:t})}{\sum_{i=0}^{t-1} \xi_{\boldsymbol{\theta}}(C_t = i, z_{1:t})}. \tag{11}$$

The term $\nabla \xi_{\boldsymbol{\theta}}(C_t, z_{1:t})$ is obtained by taking the derivative of (5)

$$\nabla \xi_{\boldsymbol{\theta}}(C_t = j, z_{1:t}) = \begin{cases} g_{\boldsymbol{\theta}}(z_t|C_t = j, z_{1:t-1}) f_{\boldsymbol{\theta}}(C_t = j|C_{t-1} = j) \\ \quad \times p_{\boldsymbol{\theta}}(C_{t-1} = j|z_{1:t-1}) \pi_t^{(j,j)} \quad \text{if } j < t-1 \\ g_{\boldsymbol{\theta}}(z_t|C_t = j, z_{1:t-1}) \\ \quad \times \sum_{i=0}^{t-2} f_{\boldsymbol{\theta}}(C_t = j|C_{t-1} = i) \\ \quad \times p_{\boldsymbol{\theta}}(C_{t-1} = i|z_{1:t-1}) \pi_t^{(i,j)} \quad \text{if } j = t-1 \end{cases} \tag{12}$$

where

$$\pi_t^{(i,j)} := \nabla \log g_{\boldsymbol{\theta}}(z_t|C_t = j, z_{1:t-1}) + \nabla \log f_{\boldsymbol{\theta}}(C_t = j|C_{t-1} = i) + \nabla \log p_{\boldsymbol{\theta}}(C_{t-1} = i|z_{1:t-1}).$$

3.2 Approximate inference

The computational cost of the recursion for computing $p_{\boldsymbol{\theta}}(C_t|z_{1:t})$ and $\nabla \log p_{\boldsymbol{\theta}}(z_t|z_{1:t-1})$ at each time t is proportional to t. This procedure is thus not appropriate for large datasets. We propose a deterministic approximation scheme to numerically approximate these quantities. Our approximation of $p_{\boldsymbol{\theta}}(C_t|z_{1:t})$ is inspired by the work of Fearnhead and Liu (2007) and relies on the following idea. At time t, the exact algorithm stores the set of probabilities $p_{\boldsymbol{\theta}}(C_t = j|z_{1:t})$ for $j = 0, 1, \ldots, t-1$. Given many of these probabilities are negligible, we can reasonably approximate the filtering distribution by a fewer set of N_t support points $c_t^{(1)}, \ldots, c_t^{(N_t)}$, called particles, with associated probability mass $w_t^{(1)}, \ldots, w_t^{(N_t)}$, called weights. To limit the number of particles N_t at time t, we adopt a simple adaptive deterministic selection scheme where all the particles whose weights are below a given threshold ε are discarded; see below. In simulations, we have found that this deterministic selection step was performing a little bit better in terms of average mean square error than the random stratified optimal resampling proposed in Fearnhead and Liu (2007). However, the algorithm in Fearnhead and Liu (2007) could also be used.

At time $t-1$, suppose that $\xi_{\boldsymbol{\theta}}(C_t, z_{1:t})$ and $p_{\boldsymbol{\theta}}(C_{t-1}|z_{1:t-1})$ are approximated through

$$\widehat{\xi}_{\boldsymbol{\theta}}(C_{t-1}, z_{1:t-1}) = \sum_{i=1}^{N_{t-1}} \widetilde{w}_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1})$$
$$\widehat{p}_{\boldsymbol{\theta}}(C_{t-1}|z_{1:t-1}) = \sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1}) \tag{13}$$

where $\delta_{c_{t-1}^{(i)}}(C_{t-1}) = 1$ if $C_{t-1} = c_{t-1}^{(i)}$ and 0 otherwise. That is $\widetilde{w}_{t-1}^{(i)}$ resp. $w_{t-1}^{(i)}$ is an approximation of $\xi_{\boldsymbol{\theta}}(C_{t-1} = c_{t-1}^{(i)}, z_{1:t-1})$ resp. $p_{\boldsymbol{\theta}}(C_{t-1} = c_{t-1}^{(i)}|z_{1:t-1})$ and $w_{t-1}^{(i)} \propto \widetilde{w}_{t-1}^{(i)}$ with $\sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} = 1$. We propose to approximate $\nabla p_{\boldsymbol{\theta}}(C_{t-1}|z_{1:t-1})$ through

$$\widehat{\nabla p_{\boldsymbol{\theta}}}(C_{t-1}|z_{1:t-1}) = \sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \beta_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1}) \tag{14}$$

where $\sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \beta_{t-1}^{(i)} = 0$; that is we are using the same particles $\{c_{t-1}^{(i)}\}$. Here $w_{t-1}^{(i)} \beta_{t-1}^{(i)}$ is an approximation of $\nabla p_{\boldsymbol{\theta}}(C_{t-1} = c_{t-1}^{(i)}|z_{1:t-1})$ so $\beta_{t-1}^{(i)}$ can be thought of as an approximation of $\nabla \log p_{\boldsymbol{\theta}}(C_{t-1} = c_{t-1}^{(i)}|z_{1:t-1})$.

At time t, let $\widetilde{c}_t^{(i)} = c_{t-1}^{(i)}$ and $\widetilde{c}_t^{(N_{t-1}+1)} = t-1$ for each particle $i = 1, \ldots, N_{t-1}$. To compute an approximation of $p_{\boldsymbol{\theta}}(C_{t-1}|z_{1:t-1})$, we plug our approximation (14) into (5) to obtain the unnormalized weights for $i = 1, \ldots, N_{t-1}$

$$\widetilde{w}_t^{(i)} = g_{\boldsymbol{\theta}}(z_t|C_t = \widetilde{c}_t^{(i)}, z_{1:t-1}) \times f_{\boldsymbol{\theta}}(C_t = \widetilde{c}_t^{(i)}|C_{t-1} = \widetilde{c}_t^{(i)}) w_{t-1}^{(i)}, \tag{15}$$

and

$$\widetilde{w}_t^{(N_{t-1}+1)} = g_{\boldsymbol{\theta}}(z_t|C_t = t-1, z_{1:t-1}) \times \sum_{i=0}^{N_{t-1}} f_{\boldsymbol{\theta}}(C_t = t-1|C_{t-1} = \widetilde{c}_t^{(i)}) w_{t-1}^{(i)}. \tag{16}$$

Similarly, by plugging (14) into (12), we obtain an approximation $\widetilde{\alpha}_t^{(i)}$ of $\nabla \xi_{\boldsymbol{\theta}}(C_t = \widetilde{c}_t^{(i)}, z_{1:t})$ which satisfies for

$i = 1, \ldots, N_{t-1}$

$$\widetilde{\alpha}_t^{(i)} = g_{\boldsymbol{\theta}}(z_t|C_t = \widetilde{c}_t^{(i)}, z_{1:t-1}) f_{\boldsymbol{\theta}}(C_t = \widetilde{c}_t^{(i)}|C_{t-1} = \widetilde{c}_t^{(i)}) w_{t-1}^{(i)} \times \left[\nabla \log g_{\boldsymbol{\theta}}(z_t|C_t = \widetilde{c}_t^{(i)}, z_{1:t-1}) + \nabla \log f_{\boldsymbol{\theta}}(C_t = \widetilde{c}_t^{(i)}|C_{t-1} = \widetilde{c}_t^{(i)}) + \beta_{t-1}^{(i)}\right], \quad (17)$$

and

$$\widetilde{\alpha}_t^{(N_{t-1}+1)} = g_{\boldsymbol{\theta}}(z_t|C_t = t-1, z_{1:t-1}) \times \sum_{i=0}^{N_{t-1}} f_{\boldsymbol{\theta}}(C_t = t-1|C_{t-1} = \widetilde{c}_t^{(i)}) w_{t-1}^{(i)} \times \left[\nabla \log g_{\boldsymbol{\theta}}(z_t|C_t = t-1, z_{1:t-1}) + \nabla \log f_{\boldsymbol{\theta}}(C_t = t-1|C_{t-1} = \widetilde{c}_t^{(i)}) + \beta_{t-1}^{(i)}\right]. \quad (18)$$

Using (10), we obtain

$$\widehat{\nabla \log p_{\boldsymbol{\theta}}}(z_t|z_{1:t-1}) = \frac{\sum_{i=1}^{N_{t-1}+1} \widetilde{\alpha}_t^{(i)}}{\sum_{i=1}^{N_{t-1}+1} \widetilde{w}_t^{(i)}}. \quad (19)$$

If we were to iterate this algorithm, the computational complexity would increase without bound with t. We only keep the particles $\widetilde{c}_t^{(i)}$ such that $\overline{w}_t^{(i)} > \varepsilon$ where $\overline{w}_t^{(i)} \propto \widetilde{w}_t^{(i)}$, $\sum_{i=1}^{N_{t-1}+1} \overline{w}_t^{(i)} = 1$ and discard the others. We then renormalize the weights of the surviving N_t particles and denote them $w_t^{(i)}$. Finally, using (11) we obtain

Particle filter for on-line changepoints and parameter estimation

At time $t = 1$

- Set $\boldsymbol{\theta}_0$, $c_1^{(1)} = 0$, $w_1^{(1)} = 1$, $w_1^{(1)}\beta_1^{(1)} = 0$ and $N_1 = 1$.

At time $t \geq 2$

- For $i = 1, \ldots, N_{t-1}$ let $\widetilde{c}_t^{(i)} = c_{t-1}^{(i)}$. Set $\widetilde{c}_t^{(N_{t-1}+1)} = t-1$.
- For $i = 1, \ldots, N_{t-1}+1$, compute $\widetilde{w}_t^{(i)}$ using (15)–(16) using $\boldsymbol{\theta}_{t-1}$.
- For $i = 1, \ldots, N_{t-1}+1$, compute $\widetilde{\alpha}_t^{(i)}$ using (17)–(18) using $\boldsymbol{\theta}_{t-1}$.
- Update the parameter vector using (9) and (19), that is

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \gamma_t \frac{\sum_{i=1}^{N_{t-1}+1} \widetilde{\alpha}_t^{(i)}}{\sum_{i=1}^{N_{t-1}+1} \widetilde{w}_t^{(i)}}$$

- Adaptive selection step: let N_t be the number of selected particles and $w_t^{(i)}$, $c_t^{(i)}$, $i = 1, \ldots, N_t$ be resp. the normalized weights and the associated support points and $\varphi : \{1, \ldots, N_t\} \to \{1, \ldots, N_{t-1}+1\}$ the injective function such that $w_t^{(i)} = \widetilde{w}_t^{(\varphi(i))}$ for $i = 1, \ldots, N_t$.
- For $i = 1, \ldots, N_t$, compute the weights $w_t^{(i)}\beta_t^{(i)}$ using (20).

$$w_t^{(i)}\beta_t^{(i)} = \frac{\widetilde{\alpha}_t^{(\varphi(i))}}{\sum_{j=1}^{N_t} \widetilde{w}_t^{(\varphi(j))}} - w_t^{(i)} \frac{\sum_{j=1}^{N_t} \widetilde{\alpha}_t^{(\varphi(j))}}{\sum_{j=1}^{N_t} \widetilde{w}_t^{(\varphi(j))}} \quad (20)$$

for $i = 1, \ldots, N_t$ where $\varphi : \{1, \ldots, N_t\} \to \{1, \ldots, N_{t-1}+1\}$ is the injective function such that $w_t^{(i)} = \overline{w}_t^{(\varphi(i))}$.

To summarize, the particle filter for joint changepoints and parameter estimation proceeds as follows.

If the number of particles is fixed to N, the overall computational complexity of the algorithm is in $O(n_\theta N \log(N) T)$ for T observations if a deterministic resampling step is used, and $O(n_\theta N T)$ if an optimal stratified resampling step is used.[2] Note that a particle filter for joint state and parameter estimation relying on recursive maximum likelihood has also been proposed in Poyiadjis et al. (2011) for the class of general non-linear non-Gaussian state-space models. However, the cost of the algorithm in Poyiadjis et al. (2011) is quadratic in the number of particles whereas it is linear in our case.

The parameter estimate typically converges before time T for large T. For smaller datasets, we can run the particle filter $K > 1$ times on the dataset, using $\boldsymbol{\theta}_1^{(j)} = \boldsymbol{\theta}_T^{(j-1)}$ and $\gamma_1^{(j)} = \gamma_T^{(j-1)}$ as the initial values for parameter estimates and step size for runs $j = 2, \ldots, K$, so as to obtain convergence. In this case, the algorithm can be interpreted as a stochastic approximation algorithm maximizing $l_T(\boldsymbol{\theta})$ given by (7). Then the particle filter may be applied to obtain the MAP and full posterior of changepoints using the final parameter estimate $\gamma_T^{(K-1)}$.

4 Genome-wide transcriptome analysis

4.1 Description of the dataset

We use two publicly available datasets to demonstrate our methodology. In the first, David et al. (2006) use high density Affymetrix tiling arrays with 25-mer oligonucleotides spaced every 4 bps on average to interrogate both strands of the full *Saccharomyces cerevisiae* genome. We will refer to these data as the *yeast* data. In the second, Cheng et al. (2005) use tiling arrays to map the sites of transcription for approximately 30% of the human genome encoded in 10 human chromosomes (6, 7, 13, 14, 19, 20, 21, 21, X, and Y). Similar to David et al. (2006), Cheng et al. (2005) use Affymetrix high density tiling arrays with 25-mer oligonucleotides spaced every 5 bps on average. These data, which we will refer to as the *human* data, also contain experimentally verified transcripts which will allow us to validate our methodology.

[2] As N is typically below 100, the computational time used for resampling is negligible in our experiments.

Similar to oligonucleotide gene expression arrays (Lockhart et al. 1996), Affymetrix tiling arrays query each sequence of interest with a perfect match (PM) and a mismatch (MM) probe, where the MM probe is complementary to the sequence of interest except at the central base, which is replaced with its complementary base. The difference is that the probes used on tiling arrays do not necessarily belong to genes, which allows for an unbiased mapping of RNA transcripts. Following the idea that MM intensities are poor measures of non-specific hybridization (Irizarry et al. 2003), we only used the PM intensities. In the case of the yeast data, the data were normalized using the procedure of David et al. (2006) and described in Huber et al. (2006), which is part of the `tilingArray` package available from Bioconductor (Gentleman et al. 2004). In the case of the human data, the data were normalized by quantile normalization (Bolstad et al. 2003), as in Cheng et al. (2005). After normalization, the data take the form $\{y_{tr} : t = 1, \ldots, T; r = 1, \ldots, R\}$, where y_{tr} is the normalized intensity of probe (also called time in the following) t from replicate r. Here, we assume that the probes are ordered by genomic positions, where we denote by $\{x_t : t = 1, \ldots, T\}$ the corresponding positions arranged in increasing order, that is $x_t < x_{t'}$ for $t < t'$. Finally, we will summarize each probe measurement by the mean of its normalized intensities across replicates, and we will denote the resulting summaries by $\{z_t : t = 1, \ldots, T\}$. Such summaries are often used in microarray studies to facilitate modeling, reduce the computational burden, and avoid across-array normalization issues; see for example Efron (2004) and Do et al. (2005).

4.2 Changepoint model

We use a changepoint model (1) where h is chosen to be a negative binomial distribution with parameters ρ and d, such that

$$h(x) = \text{Negbin}(x - u; \rho, d) = \begin{cases} \frac{\Gamma(d+x-u)}{\Gamma(x-u+1)\Gamma(d)} \rho^d (1-\rho)^{x-u} & \text{if } x \geq u \\ 0 & \text{otherwise.} \end{cases}$$

Here u controls the smallest distance between two changepoints, d controls the shape of the distribution, which has a mode greater than u for $d > 1$, and ρ controls the average length of the segments. This distribution generalizes the geometric distribution (implicitly assumed by HMMs) and was shown to give a better fit for changepoint data on the two real data sets. In these applications we will fix u to 15, corresponding to a minimum segment size of approximately 100 bps based on our biological prior knowledge, and d to 2, allowing for a positive mode. The parameter ρ is estimated from the data.

4.3 Marginal likelihood

We assume that each segment may be either *transcribed* or *non-transcribed*. Let us denote by λ the probability that a segment is transcribed and $r_i \in \{0, 1\}$ the associated latent variable indicating if segment i is transcribed ($r_i = 1$) or not ($r_i = 0$). It follows that $r_i \sim \text{Ber}(\lambda)$, that is Bernoulli with parameter λ. If $r_i = 1$ (transcript), then the data $z_{\tau_i+1:\tau_{i+1}}$, are assumed to be distributed from a normal/normal-inverse gamma compound distribution, as follows,

$$\begin{aligned} &(z_t|\mu_i, \sigma_i^2, r_i = 1) \sim \mathcal{N}(\mu_i, \sigma_i^2) \quad \text{for } t = \tau_i + 1, \ldots \tau_{i+1} \\ &(\mu_i, \sigma_i^2|r_i = 1) \sim \mathcal{NIG}(m_1, s_1, \nu_1, \gamma_1) \end{aligned} \tag{21}$$

where $\mathcal{NIG}(m_1, s_1, \nu_1, \gamma_1)$ is the normal-inverse gamma distribution, defined in Appendix A, with parameters m_1, s_1, ν_1 and γ_1 and $\mathcal{N}(\mu_i, \sigma_i^2)$ is the normal distribution with mean μ_i and variance σ_i^2. If $r_i = 0$ (not a transcript), the data $z_{\tau_i+1:\tau_{i+1}}$, are assumed to arise from a mixture of a skew t-distribution and a normal-normal inverse gamma compound distribution. If we introduce another latent variable $q_i \sim \text{Ber}(p_0)$, $p_0 \in [0, 1]$, we can write

$$\begin{aligned} &(z_t|r_i = 0, q_i = 1) \sim \mathcal{N}(\mu_i, \sigma_i^2) \quad \text{for } t = \tau_i + 1, \ldots \tau_{i+1} \\ &(\mu_i, \sigma_i^2|r_i = 0, q_i = 1) \sim \mathcal{NIG}(m_0, s_0, \nu_0, \gamma_0) \end{aligned} \tag{22}$$

and

$$(z_t|r_i = 0, q_i = 0) \sim st(\varphi_0, \psi_0, \zeta_0, \xi_0)$$

where m_0, s_0, ν_0, γ_0, φ_0 and ψ_0 are unknown parameters that will be estimated while ζ_0 and ξ_0 will be fixed in advanced. Note that in (22) resp. (21), the unknown parameters are shared across non-transcript segments, resp. transcripts, which allows us to borrow strength across segments when estimating segment boundaries. The skew t-distribution, $st(\varphi_0, \psi_0, \zeta_0, \xi_0)$, is as defined in Azzalini and Capitanio (2003) and whose density is given in Appendix A. The parameters φ_0, ψ_0, ζ_0 and ξ_0 represent the location, scale, degrees of freedom, and skewness parameters. In the example explored in this paper, we will use $\zeta_0 = 4$ for the degrees of freedom parameter to provide for robustness against outliers, and $\xi_0 = 10$ for the skewness parameter, which seems to be enough to deal with the skewness observed for non-transcript segments. Even though these parameters could be estimated, we have chosen to fix them for simplicity. However, the exact value of these parameters is not crucial; experimentation showed that different values give similar results. For the non-transcribed segments, we have found it necessary to introduce a skew t-distribution to deal with frequent outliers and the skewed nature of low-intensity observations. We have experimented with a single normal/normal-inverse gamma compound distribution for the baseline and the results were not as good

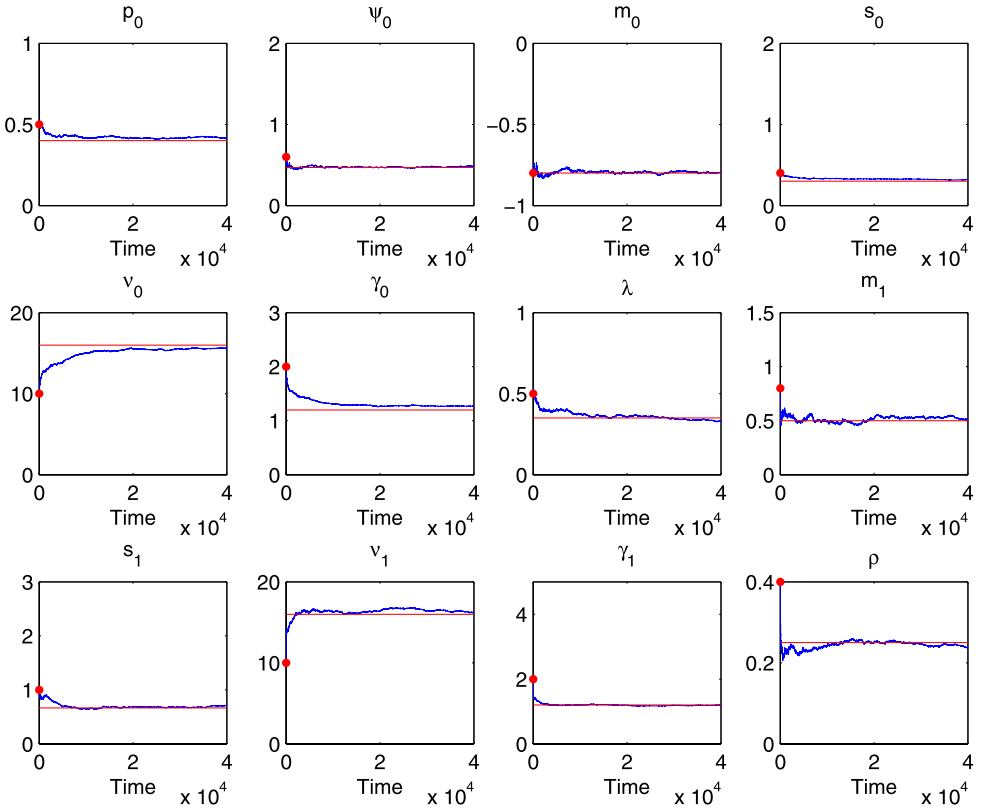


Fig. 1 Parameter estimates for the simulated data. The value of each parameter is shown as a function of iterations. The true value for each parameter is shown with a *horizontal line*

in terms of goodness of fit and segmentation results (data not shown). In order to have the same mean value for the non-transcribed segments, we assume that the mean of the skew t-distribution is equal to m_0, that is we set $m_0 = \varphi_0 + \psi_0\xi_0/\sqrt{1+\xi_0^2}\sqrt{\zeta_0/\pi}\,\Gamma((\zeta_0-1)/2)/\,\Gamma(\zeta_0/2)$. Note that we use the same generic variables μ_i and σ_i^2 in (21) and (22) for ease of notation even though these are different parameters. In any case, these variables are nuisance parameters which will be integrated out later on.

The marginal likelihood $P(\tau_i, \tau_{i+1}) := p(z_{\tau_i+1:\tau_{i+1}})$ conditional on two consecutive changepoints τ_i and τ_{i+1} and the unknown parameters, which are omitted below for ease of notation, is given in Appendix B.

4.4 Results

4.4.1 Synthetic dataset

In order to evaluate our approach and compare it to Huber's method (Huber et al. 2006), we present a simulation study on synthetic data for which the ground truth is available.. We have simulated 40,000 observations from our changepoint model with the following parameters $p_0 = 0.4$, $\psi_0 = 0.47$, $\zeta_0 = 4$, $\xi_0 = 10$, $m_0 = -0.8$, $\varphi_0 = -1.27$, $s_0 = 0.3$, $\nu_0 = 16$, $\gamma_0 = 1.2$, $m_1 = 0.5$, $s_1 = 0.67$, $\nu_1 = 16$, $\gamma_1 = 1.2$, $\lambda = 0.35$, $\rho = 0.25$, $\alpha = 10^{-6}$, $d = 2$, $u = 15$. These values were chosen to be within the range of the estimated parameters on real data. The parameters $\boldsymbol{\theta}$ were first estimated on the whole dataset using our on-line algorithm. The evolution of the parameter estimates over time are represented in Fig. 1. The algorithm manages to correctly estimate this set of parameters. Based on the final estimated value, the particle filter is then run again on the whole dataset. The MAP, posterior of changepoints, and number of particles for a portion of the data are represented in Fig. 2. The true transcribed segments are represented by red patches. The number of particles varies over time adaptively. It increases as long as there is no changepoint, and decreases when evidence of a changepoint occurs. Even with a few number of particles (20 on average), the algorithm manages to estimate the model parameters (including the segment boundaries) very well. This shows that our approximation is good even when the number of particles is low and that an increase in the number of particles does not imply significant improvements in the estimation of the changepoints and parameters.

We also compared our method with the dynamic programming approach used in Huber et al. (2006). We took the same parameters with p_0 varying from 0 (skew-t distribution for the baseline) to 1 (normal distribution for the baseline). We have simulated for each value of p_0 200 datasets of size 1,000 and we compared the error on the number of estimated changepoints, the number of false positive and the number of false negative for both methods, using MAP es-

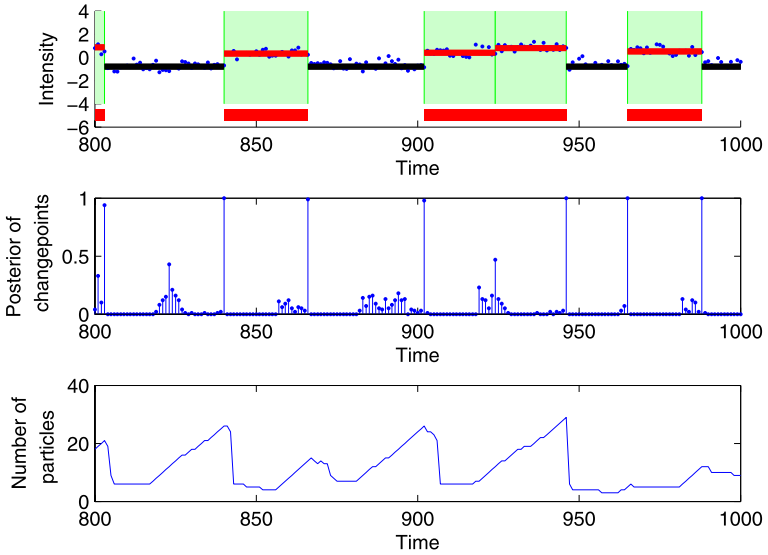
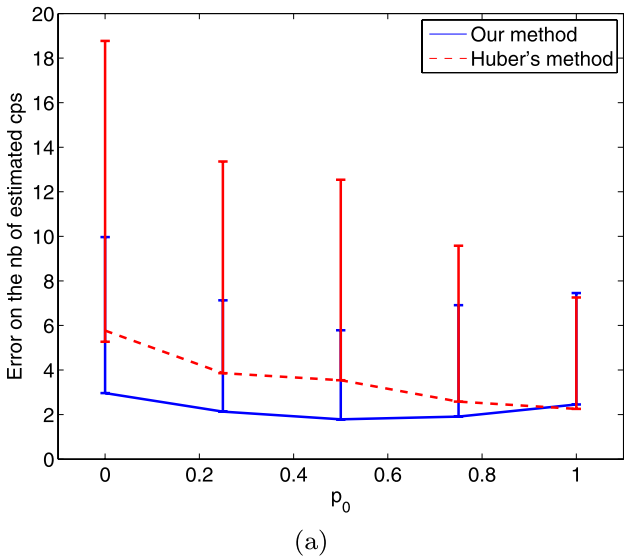


Fig. 2 (*top*) MAP, (*middle*) posterior of changepoints and (*bottom*) number of particles for the simulated dataset. *On the top figure*, the true transcribed segments are represented by *red patches*



(a)

Fig. 3 Mean absolute error and 90% confidence bounds on the estimated number of changepoints for Huber's and our method

timates for our method. The results are reported in Figs. 3 and 4. When the baseline data are normally distributed, both methods perform similarly. As p_0 increases, the number of false positive for Huber's method increases while it remains roughly the same for our method. The number of false negative are equivalent for both methods whatever the value of p_0 is. Note that contrary to Huber's method, ours is also able to estimate the model label of each segment.

4.4.2 *Yeast dataset*

We fitted our changepoint model to the positive strand of the first chromosome of the yeast data, using $u = 15$, $d = 2$, $\zeta_0 = 4$ and $\xi_0 = 10$, as explained earlier.

The parameters $\boldsymbol{\theta} = \{p_0, \psi_0, m_0, s_0, \nu_0, \gamma_0, \lambda, m_1, s_1, \nu_1, \gamma_1, \rho\}$ were estimated by running the particle filter $K = 20$ times with $\varepsilon = 10^{-6}$ on the full dataset, hence requiring 4×10^5 iterations. Evolution of each parameter with respect to iterations is shown in Fig. 5. Although we have used $K = 20$ passes over the whole dataset in order to show the convergence, most parameters had converged after only two passes. In terms of segmentation and classification, the results obtained using the parameter after 2 passes were very similar to the results obtained after 20 passes. Note that, as stated in the previous section, for a larger number of probes the parameter estimates would typically converge more quickly as there is more information available. The final parameter value $\hat{\boldsymbol{\theta}}$ obtained after 20 passes over the full dataset, shown in Table 1, is used as the parameter values and the particle filter is then ran with $\varepsilon = 10^{-6}$ in order to obtain the segmentation. The MAP estimate of the changepoints for a portion of the whole chromosome is represented in Fig. 6 (top). The associated number of changepoints for the whole chromosome is 299. Figure 6 (bottom) also shows the results using the algorithm of Huber et al. (2006), with 153 segments over chromosome 1. The number 153 was estimated using previous biological knowledge as explained in David et al. (2006). Overall, both segmentations show similar results and clearly agree with known coding sequence (CDS) annotations. This said, the advantage of our methodology over Huber's is obvious when looking at the segmentation results as we get a direct classification of the segments into transcripts and non-transcripts. Additionally, no thresholding is necessary. Using our method, one can easily see that some of the detected transcripts (green background) do not overlap with know annotations. This confirms the findings of David et al. (2006) that even this well-studied genome has transcriptional complexity far beyond current annotations.

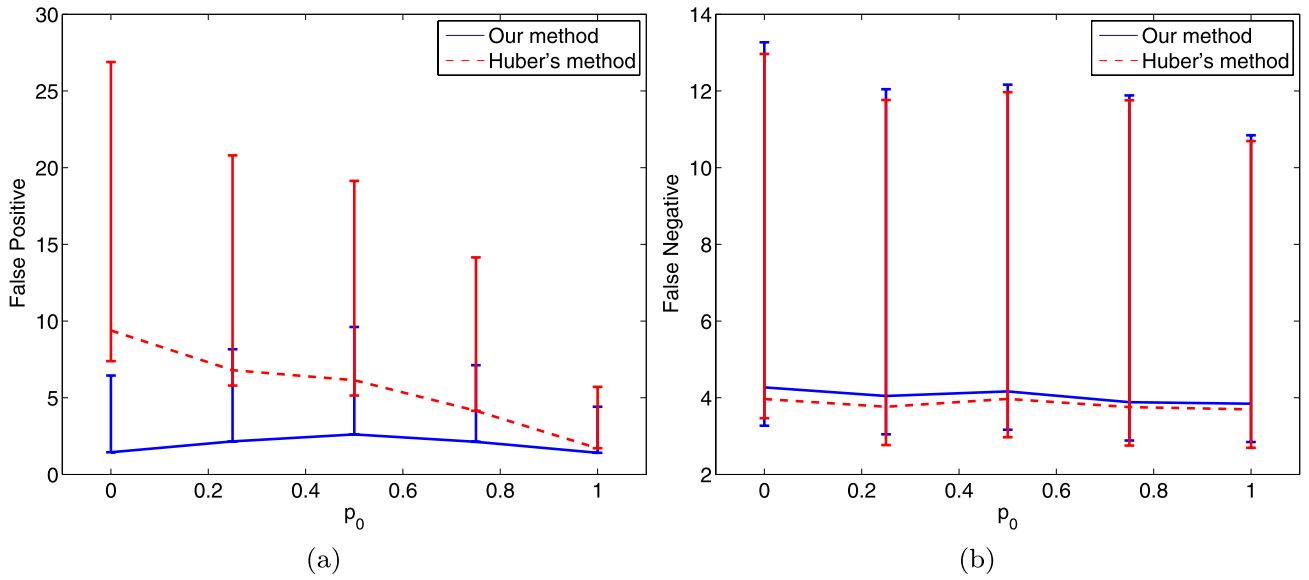


Fig. 4 (**a**) Mean number of false positive and 90% confidence bounds for Huber's and our method. (**b**) Mean number of false negative and 90% confidence bounds for Huber's and our method

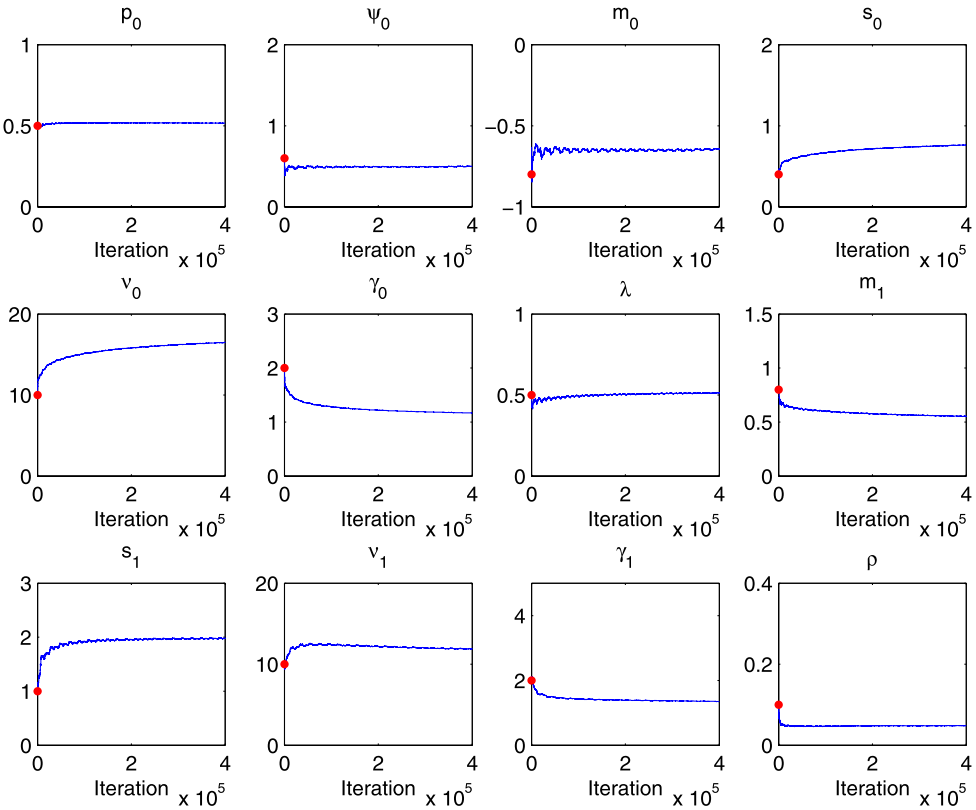


Fig. 5 Online estimation convergence for the yeast data. The value of each parameter is shown as a function of iterations. The *dot* on the left-hand side of each plot represents the initial value of the hyperparameter

Table 1 Summary of parameter estimates for both the yeast and human data, using the online estimation procedure

Parameter	p_0	ψ_0	m_0	s_0	ν_0	γ_0	λ	m_1	s_1	ν_1	γ_1	ρ
Yeast	.52	.50	−0.64	.76	16.5	1.17	.51	.55	1.98	11.88	1.35	.05
Human	.23	.93	−0.37	.96	10.83	1.84	.63	1.18	.83	6.02	4.32	.21

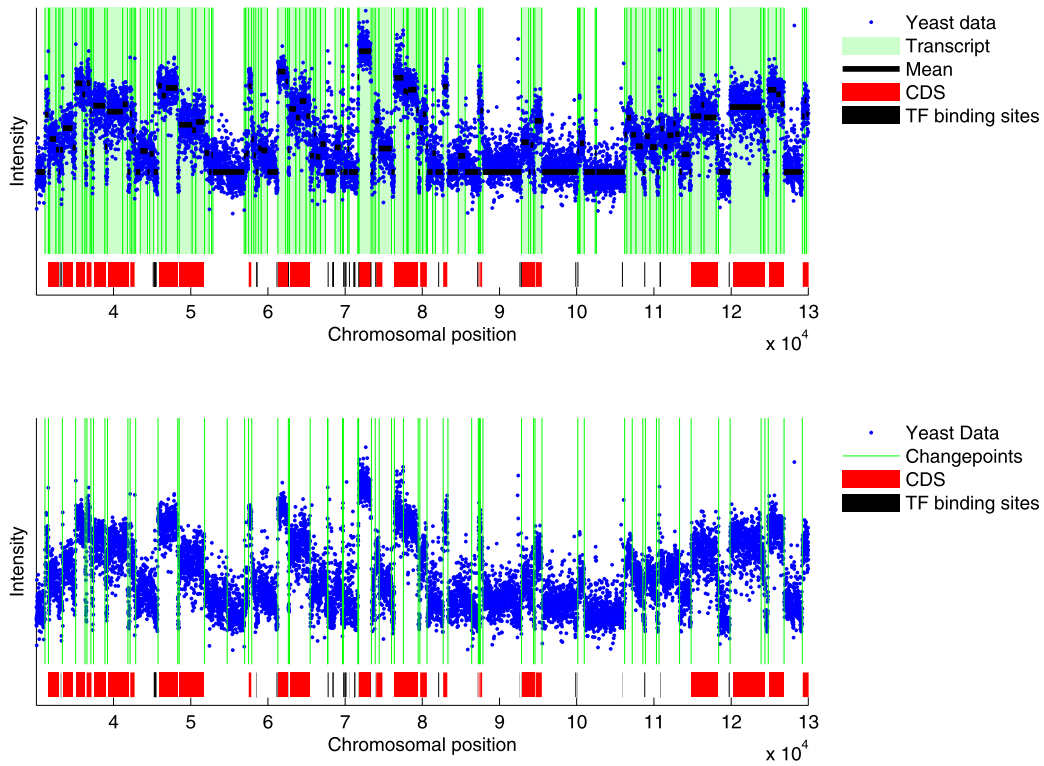


Fig. 6 (Color online) Segmentation results for part of chromosome 1 for the yeast data using our algorithm (*top*) and Huber et al.'s algorithm (*bottom*). For the *top graph*, the MAP estimate is displayed with transcript segments (*green background*), non-transcript segments (*white background*) segments and *black* segments for the segment intensity levels. For both *top* and *bottom graphs*, segments boundaries are represented with *green vertical lines*. Transcript annotations are shown below with *red rectangles* representing coding sequences and *black segments* representing TF binding sites

Note that, using our method, the number of segments is estimated whereas in Huber's it has to be fixed in advance. Our estimated number of segments is significantly larger than the number used by David et al. (2006), but a closer look at the segmentation results suggests that such a larger number is necessary to explain changes in intensity along the chromosome; see Fig. 7 where we have zoomed onto two specific regions. For example, the left parts of Fig. 7(a) (around 6.9×10^4) show a clear jump in the observed intensities, which is detected as a separate transcript by our method (top) but not Huber's (bottom). Similar observations can be made for the left parts of Fig. 7(b) (around 1.14×10^5), where our method detects a putative transcript not detected by Huber's. Even though David et al. (2006) decided to fix the number of segments to 153 using previous biological knowledge, Huber et al. (2006) also provide a method for estimating the number of segments based on AIC or BIC. For the data used here, the estimated number of segments using AIC and BIC are 307 and 232, respectively, which are closer to our estimate. Finally, Fig. 7(a) also shows the marginal posterior probabilities of changepoints, which provide nice measures of uncertainty for the corresponding changepoints. These marginal probabilities are obtained with 1,000 draws distributed from the approximated joint posterior distribution of the changepoints; see the algorithm in Sect. 3.1.1.

Overall, using the yeast data, we have shown that our changepoint model is a compelling method for RNA transcript segmentation using tiling arrays as it automatically estimates the number of segments along with their classification while also estimating important tuning parameters. We now turn to a more complex human dataset (Cheng et al. 2005).

4.4.3 Human dataset

As with the yeast data, we fitted our changepoint model to the chromosome 6 of the human data with the same fixed parameters, namely $u = 15$, $d = 2$, $\zeta_0 = 4$ and $\xi_0 = 10$. For ease of comparison with Huber's segmentation algorithm, we have only selected a subset of chromosome 6 which contains 20,000 probes with many known annotations and verified transcript regions. For comparison, we have also ran our algorithm on the whole chromosome 6, and the results were very similar. The parameters $\boldsymbol{\theta} = \{p_0, \psi_0, m_0, s_0, \nu_0, \gamma_0, \lambda, m_1, s_1, \nu_1, \gamma_1, \rho\}$ are first estimated, running the particle filter with $\varepsilon = 10^{-6}$ twenty times on the full dataset, hence 4×10^5 iterations. Evolution of $\boldsymbol{\theta}$

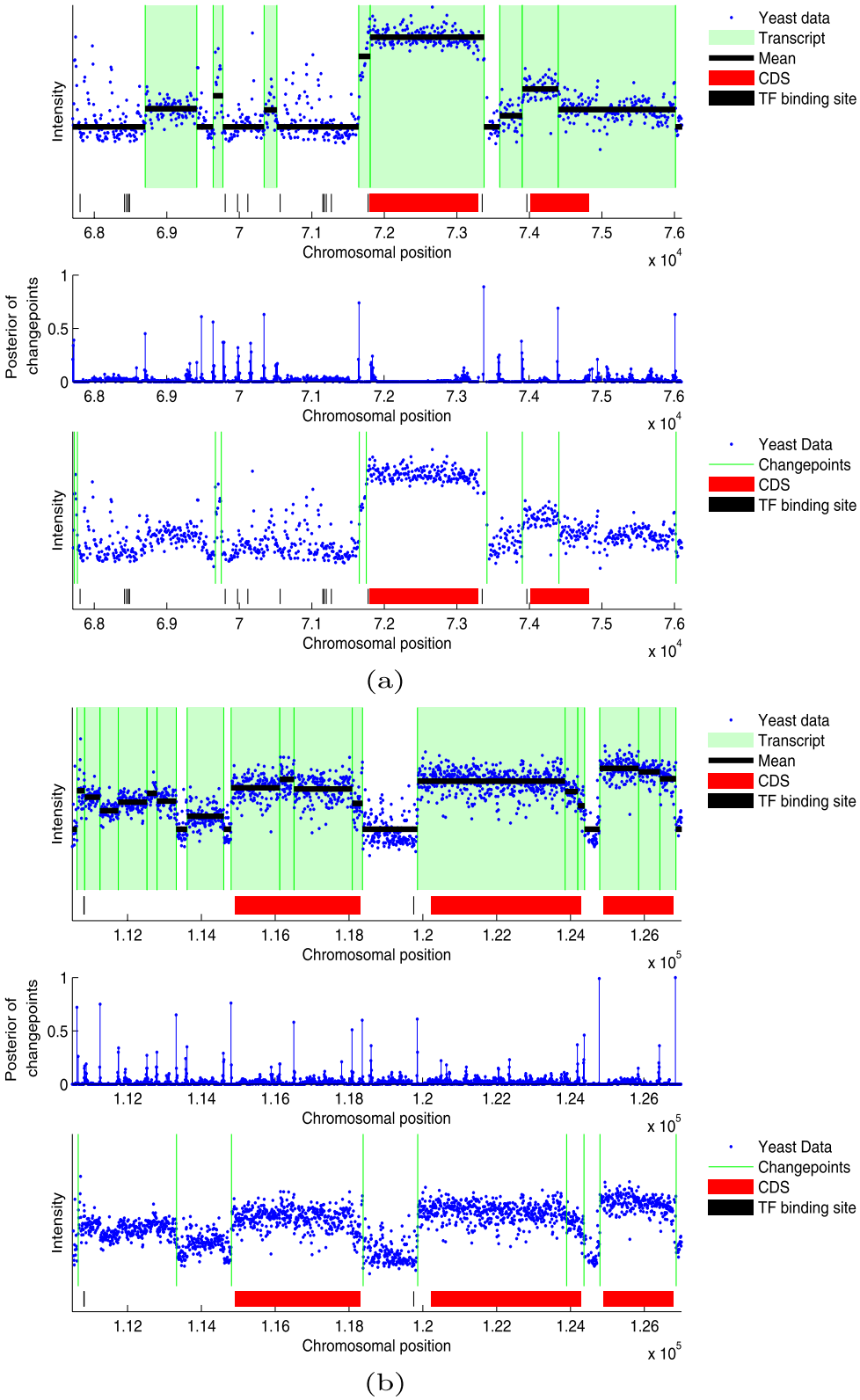


Fig. 7 Segmentation results for two close up regions from the yeast data. Our MAP segmentation (*top*) provides a better fit to the data by segmenting a few jumps in the data not detected as segments by Huber's method (*bottom*). The posterior probabilities of changepoints are represented in the *middle plots*

with respect to iterations is shown in Fig. 8. Most of the parameters have converged. The parameters associated to the skew t-distribution converge slowly due to the small probability of this mixture component (around 0.09). Using more iterations for the parameter estimation has shown very little difference for the changepoint results.

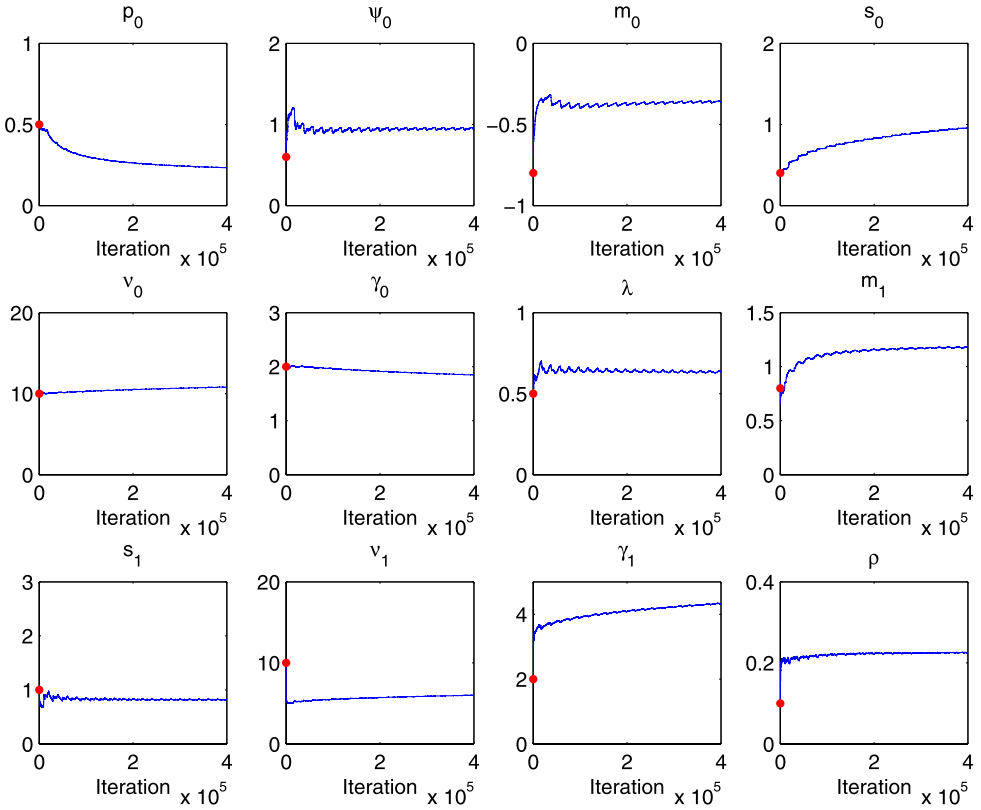


Fig. 8 Parameter estimates for the human dataset. The value of each parameter is shown as a function of iterations. The *dot* on the left-hand side of each plot represents the initial parameter estimates

The estimate of the parameter $\boldsymbol{\theta}$ obtained after 20 passes (Table 1), is used as the parameter value, then the particle filter algorithm is applied with $\varepsilon = 10^{-6}$ in order to obtain the segmentation. The MAP segmentation estimate of the changepoints for a portion of the whole chromosome is represented in Fig. 9 (top). The associated number of changepoints is 824, which is significantly higher than for the yeast data used previously, even though it contains roughly the same number of probes. This is not surprising as the human genome is far more complex than its yeast counterpart as it contains many exons (Fig. 9). In addition, the experiment of Cheng et al. (2005) was not strand-specific (hence the presence of both $+/-$ annotations on Fig. 9) which could lead to more transcripts being detected. Finally, Cheng et al. (2005) did not use a control sample to normalize their data as did David et al. (2006), which could potentially lead to the detection of false transcripts due to sequence specific biases. Figure 9 (bottom) also shows the results using the algorithm of Huber et al. (2006), fixing the number of segments to 204 using BIC. Using AIC, the optimal number of segments is 5448, which seems a bit large.

Because of the large number of changepoints, it is hard to compare our approach with that of Huber et al. (2006) based on Fig. 9 alone. This said, the advantage of our methodology over Huber's is once again obvious when looking at the segmentation results as we get a direct classification of segments into transcripts and non-transcripts. In addition, using our method, the number of segments is estimated automatically. As with the yeast data, Fig. 9 shows that many of the detected transcripts (green background) do not overlap with known annotations. This confirms the findings of Cheng et al. (2005), where the authors have noted that most of the detected transcripts were previously unannotated.

The number of segments estimated by our method is somewhat larger than the number estimated by Huber's segmentation combined with BIC, but a closer look at some specific regions suggests that such a number is necessary to explain changes in intensity along the chromosome; see Fig. 10 where we have zoomed onto two specific regions. For example the left parts of Fig. 10(a) (around 7.1715×10^6) show many jumps in the observed intensities which are detected as separate transcripts by our method (top) but not Huber's (bottom). In fact, Huber's method fails to properly segment one validated region (mark as verified transcript). Figure 10 also shows the regions detected as transcripts by the sliding window approach of Cheng et al. (2005). In general, our method and Huber's lead to precise estimates of the transcript boundaries whereas the sliding window approach tends to smooth out the boundaries, confirming previous observations made by Huber et al. (2006). In addition, the sliding window approach requires one to derive a threshold in order to call transcript regions, which can be difficult without prior knowledge. Cheng et al. (2005) used negative control measurements to derive the threshold used to de-

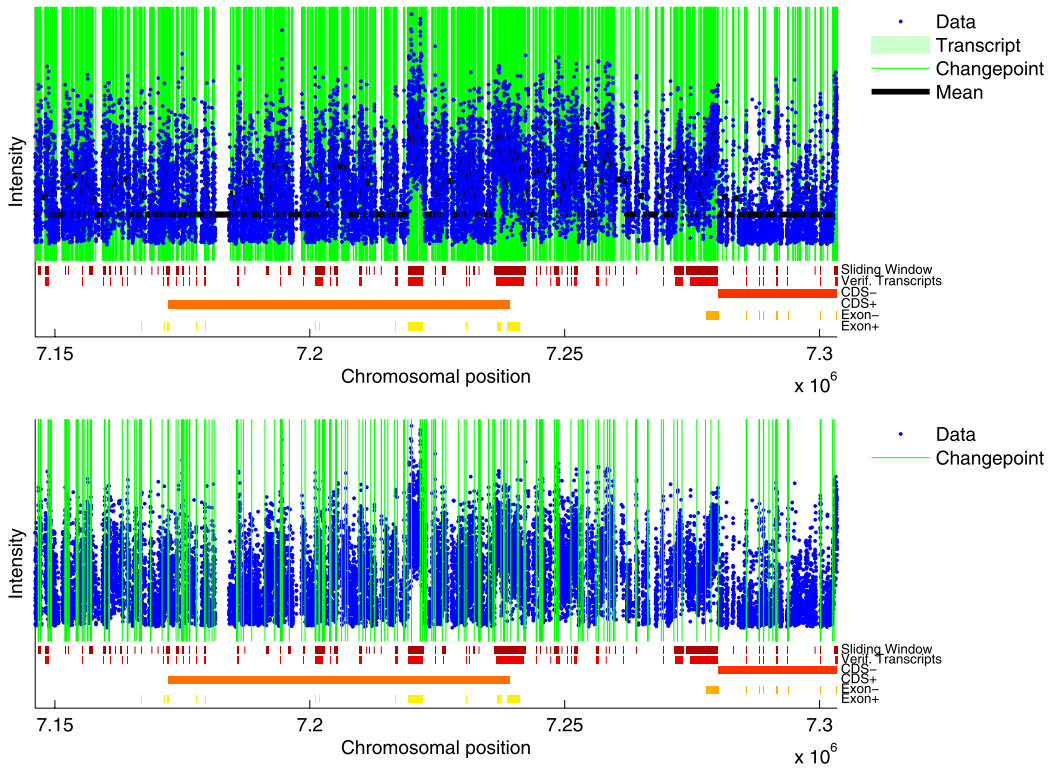


Fig. 9 (Color online) Segmentation results for part of chromosome 6 for the human dataset using our algorithm (*top*) and Huber et al.'s algorithm (*bottom*). For the *top graph*, the MAP estimate is displayed with transcript segments (*green background*) and non-transcript segments (*white background*), and *black segments* for the segment intensity levels. For both *top* and *bottom graphs*, segment boundaries are represented with *green vertical lines*. Transcript annotations are shown with coding sequences and Exon for both strands. We also show the transcript regions found by the sliding window method of Cheng et al. (2005), and the subset of these that were experimentally verified

tect transcripts, but such controls are not always available. Using our method, we simultaneously estimate the number of segments along with their classification (transcript/non-transcript). In particular, our method correctly classifies all of the verified transcripts. Note that such classification is not possible with Huber's method.

4.4.4 Model checking

In order to check model assumptions for both datasets, we now look at the predictive cumulative distribution $\Pr(Z_t \leq z_t|z_{1:t-1})$ evaluated at z_t. If the model assumptions are correct, these values should be uniformly distributed between 0 and 1 and $\Phi^{-1}(\Pr(Z_t \leq z_t|z_{1:t-1}))$, where Φ^{-1} is the inverse Gaussian cdf, should be normally distributed. The histogram of the predictive distribution and the associated qq-plot are represented in Fig. 11. Although the model is slightly overconfident, the histogram and qq-plot show that our model fits the data quite well for the yeast dataset. For the human dataset, the qq-plot and histogram are not as good, which is not surprising as the data are more noisy than the yeast data. Nonetheless, there is no evidence of severe mis-specification.

5 Discussion

This paper has presented an original algorithm to perform jointly on-line changepoint detection and parameter estimation. This algorithm has a computational complexity which is linear in the number of data and does not suffer from degeneracy problems of standard particle methods for static parameter estimation (Andrieu et al. 2005; Fearnhead 2002). Let N be the number of particles then the computational complexity is only in N (optimal stratified resampling) or $N \log N$ (deterministic resampling) compared to N^2 for general state-space models (Poyiadjis et al. 2011).

The model relies on the assumption that the marginal likelihood (4.3) can be computed analytically. This assumption, which leads to very efficient algorithms, is made in several other papers on Bayesian analysis of changepoint models (Fearnhead and Liu 2007; Fearnhead 2006; Xuan and Murphy 2007) and other Bayesian modeling frameworks (Gottardo et al. 2003; Colella et al. 2007; Kendziorski et al. 2003) to cite a few.

In a genomic application, we have demonstrated that our approach provides a powerful framework for detecting RNA transcripts from tiling array experiments. In this context, it

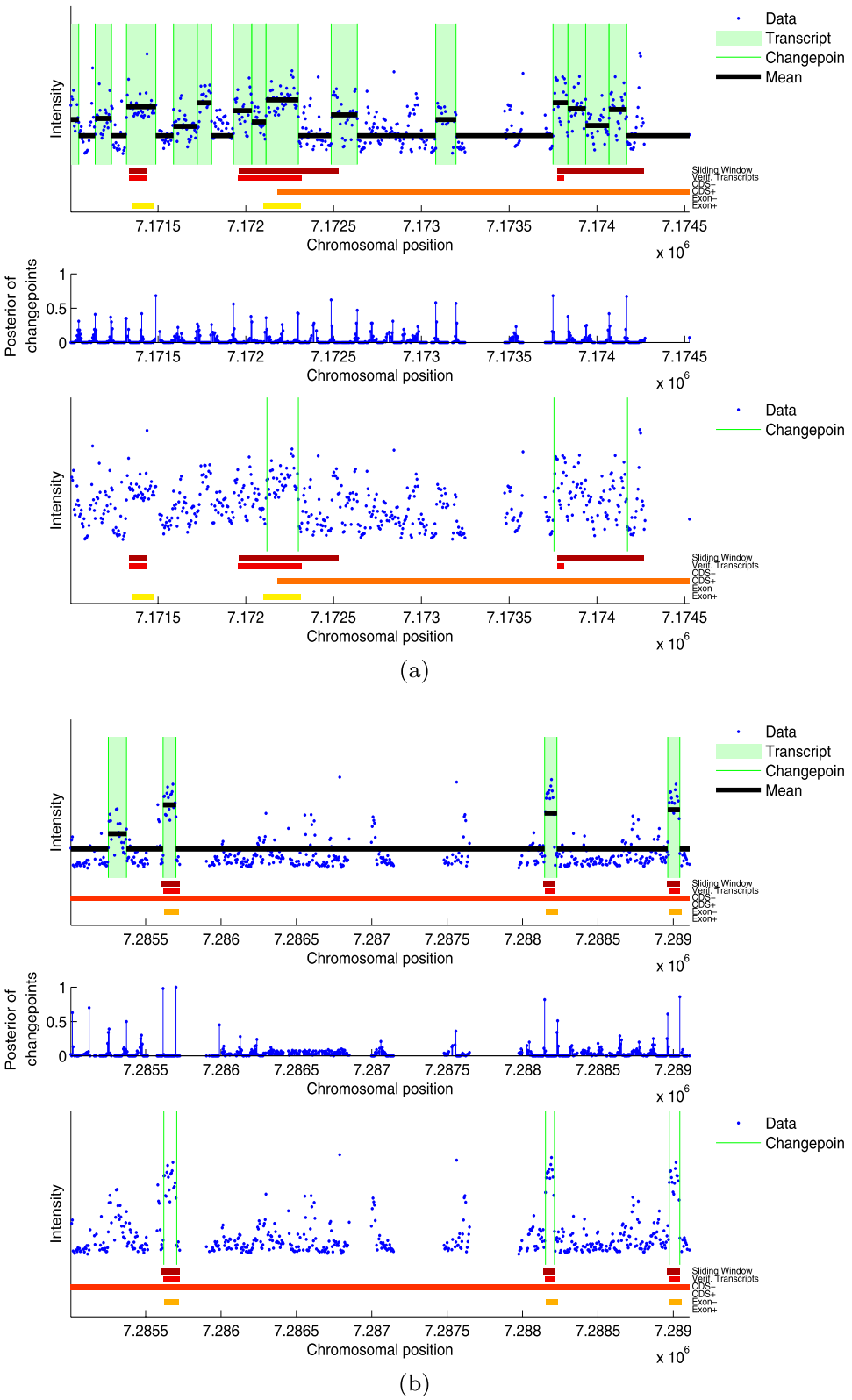


Fig. 10 Segmentation results for two close-up regions from the human data. Our MAP segmentation (*top*) provides a better fit to the data and properly detect a verified transcript not detected by Huber's method (*bottom*)

presents several advantages over current approaches. It can automatically detect the number of transcripts and classify them as transcribed/not-transcribed. Using two experiments on Affymetrix tiling arrays, we have shown that our algorithm can provide powerful detection of RNA transcript compared to a sliding window approach or a simple seg-

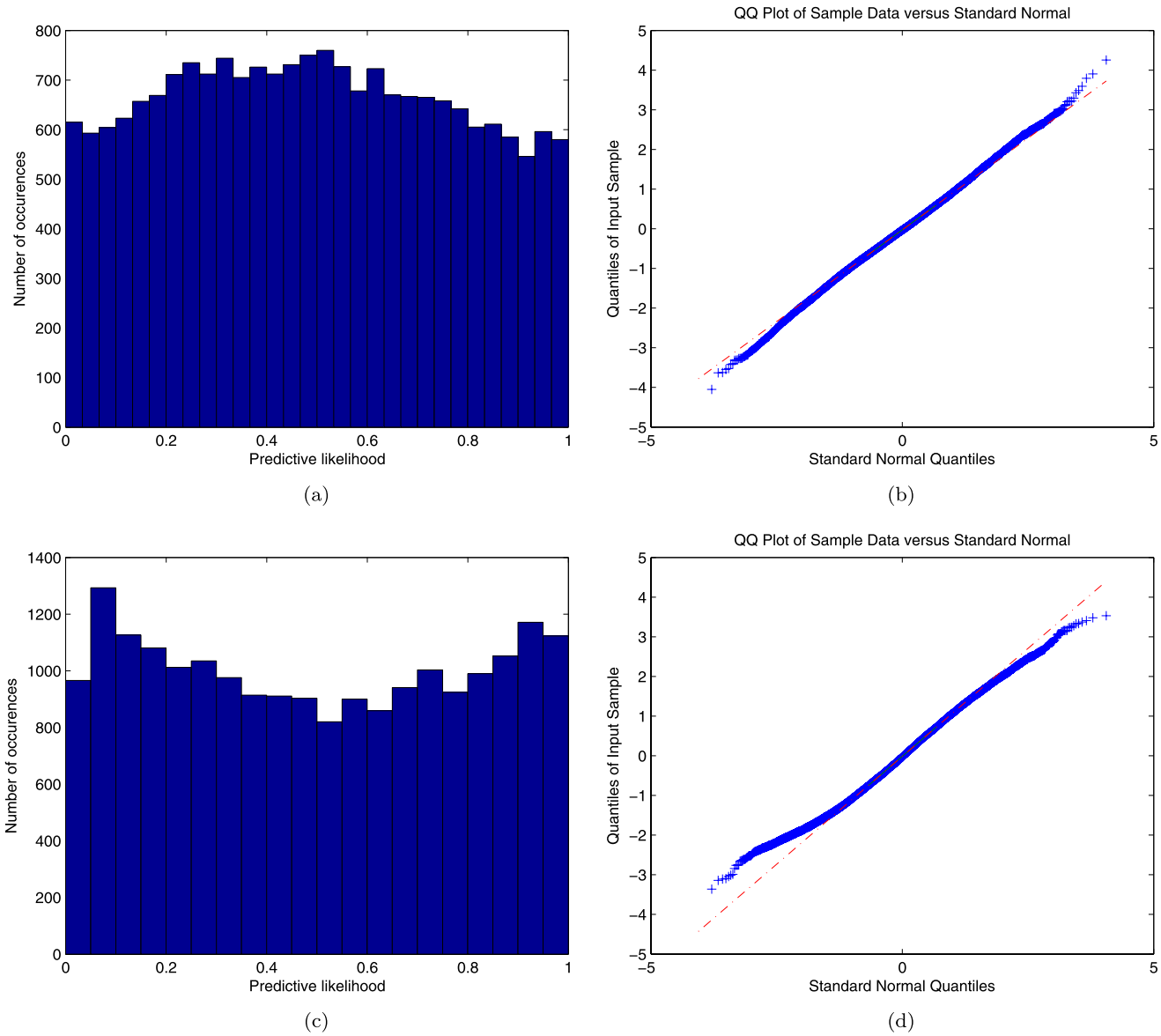


Fig. 11 Histogram of $\Pr(Z_t \leq z_t | z_{1:t-1})$ and qq-plot for the Yeast (**a**–**b**) and Human (**c**–**d**) datasets

mentation algorithm. This is particularly true of the human dataset were we have detected all of the verified transcripts. In addition, we have performed a simulation study which showed that our estimation procedure provides good estimates of the unknown parameters, including the unknown changepoints.

It is possible to propose various straightforward extensions of the model and associated algorithm. Although our applications deal with univariate time series, it can be used directly for multivariate time series; see Xuan and Murphy (2007) for some interesting examples. Additionally, instead of focusing on piecewise constant signals, we could for example consider switching linear regression or switching autoregressive models (Fearnhead 2005). This could be useful in a biological context. Here we have assumed that the biological process of transcription can be described by piecewise constant expression levels as in Huber et al. (2006) and David et al. (2006). In reality, the actual biological process could lead to more complex hybridization profiles than the piecewise constant shape assumed here.

Acknowledgements The authors are grateful to Luke Bornn for helpful comments and Philipp Kapranov for helpful discussion about the human data. François Caron was funded by a postdoctoral fellowship from the Pacific Institute for the Mathematical Sciences (PIMS) and the Centre National de la Recherche Scientifique (CNRS). Raphael

Gottardo was funded by National Institutes of Health (NIH) grant no. R01-HG005692.

Appendix A: Distributions

The probability distribution of the skew t-distribution with parameters φ_0, ψ_0, ζ_0 and ξ_0 is given by

$$\frac{2}{\psi_0} t\left(\frac{x-\varphi_0}{\psi_0}, \zeta_0\right) T\left(\xi_0\left(\frac{x-\varphi_0}{\psi_0}\right) \times \sqrt{\frac{\zeta_0+1}{\left(\frac{x-\varphi_0}{\psi_0}\right)^2+\zeta_0}}, \zeta_0+1\right) \tag{23}$$

where t and T are the standard centered student t density and cumulative density function, respectively. The parameters φ_0, ψ_0, ζ_0 and ξ_0 represent the location, scale, degrees of freedom and skewness parameters. The normal inverse gamma distribution $(\mu, \sigma^2) \sim \mathcal{N}i\mathcal{G}(m_1, s_1, \nu_1, \gamma_1)$ is defined by

$$(\mu_i|\sigma_i^2) \sim \mathcal{N}(m_1, s_1^2\sigma_i^2), \quad \sigma_i^2 \sim i\mathcal{G}\left(\frac{\nu_1}{2}, \frac{\gamma_1}{2}\right)$$

and the resulting joint pdf is given by

$$\begin{aligned}&\mathcal{N}i\mathcal{G}(\mu_i, \sigma_i^2|m_1, s_1, \nu_1, \gamma_1)\\ &\quad= (2\pi s_1^2\sigma_i^2)^{-1/2}\exp\left(-\frac{1}{2s_1^2\sigma_i^2}(x-m_1)^2\right)\\ &\qquad\times \frac{(\frac{\gamma_1}{2})^{\frac{\nu_1}{2}}}{\Gamma(\frac{\nu_1}{2})}(\sigma_i^2)^{-\frac{\nu}{2}-1}\exp\left(-\frac{\gamma_1}{2\sigma_i^2}\right).\end{aligned}$$

Appendix B: Marginal likelihoods

B.1 Genome-wide transcriptome analysis

Conditioning on two consecutive changepoints τ_i and τ_{i+1} and the unknown parameters, which are omitted below for ease of notation, the marginal likelihood is given by

$$P(\tau_i, \tau_{i+1}) := p(z_{\tau_i+1:\tau_{i+1}}) \tag{24}$$

$$= (1-\lambda)p(z_{\tau_i+1:\tau_{i+1}}|r_i=0) + \lambda p(z_{\tau_i+1:\tau_{i+1}}|r_i=1) \tag{25}$$

B.1.1 Transcribed segments

The marginal likelihood $p(z_{\tau_i+1:\tau_{i+1}}|r_i=1)$ is

$$\begin{aligned}&p(z_{\tau_i+1:\tau_{i+1}}|r_i=1)\\ &\quad= \int p(z_{\tau_i+1:\tau_{i+1}}|r_i=1, \mu_i, \sigma_i^2)p(\mu_i, \sigma_i^2)d\mu_i d\sigma_i\\ &\quad= \pi^{-n/2}(1+ns_1^2)^{-1/2}\\ &\qquad\times\left(s^2+\frac{n(m-m_1)^2}{1+ns_1^2}+\gamma_1\right)^{-(n+\nu_1)/2}\\ &\qquad\times\gamma_1^{\nu_1/2}\frac{\Gamma((n+\nu_1)/2)}{\Gamma(\nu_1/2)}\end{aligned}$$

where $m = \frac{1}{n}\sum_{k=\tau_i+1}^{\tau_{i+1}} z_k$, $s^2 = \sum_{k=\tau_i+1}^{\tau_{i+1}}(z_k-m)^2$ and $n = \tau_{i+1}-\tau_i$.

B.1.2 Non-transcribed segments

The marginal likelihood $p(z_{\tau_i+1:\tau_{i+1}}|r_i=0)$ is

$$\begin{aligned}p(z_{\tau_i+1:\tau_{i+1}}|r_i=0) &= (1-p_0)p(z_{\tau_i+1:\tau_{i+1}}|r_i=0, q_i=0)\\ &\quad+ p_0 p(z_{\tau_i+1:\tau_{i+1}}|r_i=0, q_i=1)\end{aligned}$$

where

$$\begin{aligned}&p(z_{\tau_i+1:\tau_{i+1}}|r_i=0, q_i=0)\\ &\quad= \prod_{k=\tau_i+1}^{\tau_{i+1}} st(z_k; \varphi_0, \psi_0, \zeta_0, \xi_0)\\ &p(z_{\tau_i+1:\tau_{i+1}}|r_i=0, q_i=1)\\ &\quad= \pi^{-n/2}(1+ns_0^2)^{-1/2}\\ &\qquad\times\left(s^2+\frac{n(m-m_0)^2}{1+ns_0^2}+\gamma_0\right)^{-(n+\nu_0)/2}\\ &\qquad\times\gamma_0^{\nu_0/2}\frac{\Gamma((n+\nu_0)/2)}{\Gamma(\nu_0/2)}\end{aligned}$$

where again $m = \frac{1}{n}\sum_{k=\tau_i+1}^{\tau_{i+1}} z_k$, $s^2 = \sum_{k=\tau_i+1}^{\tau_{i+1}}(z_k-m)^2$ and $n = \tau_{i+1}-\tau_i$.

References

Andrieu, C., Doucet, A., Tadic, V.: On-line parameter estimation in general state-space models. In: Proc. 44th IEEE Conference on Decision and Control (2005)

Azzalini, A., Capitanio, A.: Distributions generated by perturbation of symmetry with emphasis on a multivariate skew t-distribution. J. R. Stat. Soc. B **65**, 367–389 (2003)

Barry, D., Hartigan, J.: Product partition models for change point problems. Ann. Stat. **20**, 260–279 (1992)

Benveniste, A., Metivier, M., Priouret, P.: Adaptive Algorithms and Stochastic Approximations. Springer, Berlin (1990)

Bertone, P., Stolc, V., Royce, T.E., Rozowsky, J.S., Urban, A.E., Zhu, X., Rinn, J.L., Tongprasit, W., Samanta, M., Weissman, S., Gerstein, M.B., Snyder, M.: Global identification of human transcribed sequences with genome tiling arrays. Science **306**(5705), 2242–2246 (2004)

Bolstad, B.M., Irizarry, R.A., Astrand, M., Speed, T.P.: A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics **19**(2), 185–193 (2003)

Carlin, B., Gelfand, A., Smith, A.: Hierarchical Bayesian analysis of changepoint problems. Appl. Stat. **41**, 389–405 (1992)

Cheng, J., Kapranov, P., Drenkow, J., Dike, S., Brubaker, S., Patel, S., Long, J., Stern, D., Tammana, H., Helt, G., Sementchenko, V., Piccolboni, A., Bekiranov, S., Bailey, D.K., Ganesh, M., Ghosh, S., Bell, I., Gerhard, D.S., Gingeras, T.R.: Transcriptional maps of 10 human chromosomes at 5-nucleotide resolution. Science **308**(5725), 1149–1154 (2005)

Chib, S.: Estimation and comparison of multiple change-point models. J. Econom. **86**, 221–241 (1998)

Chopin, N.: Dynamic detection of change points in long time series. Ann. Inst. Math. Sci. **59**, 349–366 (2007)

Colella, S., Yau, C., Taylor, J., Mirza, G., Butler, H., Clouston, P., Bassett, A., Seller, A., Holmes, C., Ragoussis, J.: QuantiSNP: an objective Bayes hidden-Markov model to detect and accurately map copy number variation using SNP genotyping data. Nucleic Acids Res. **35**, 2013–2025 (2007)

David, L., Huber, W., Granovskaia, M., Toedling, J., Palm, C.J., Bofkin, L., Jones, T., Davis, R.W., Steinmetz, L.M.: A high-resolution map of transcription in the yeast genome. Proc. Natl. Acad. Sci. USA **103**(14), 5320–5325 (2006)

De Iorio, M., de Silva, E., Stumpf, M.: Recombination hotspots as a point process. Philos. Trans. R. Soc. B **360**, 1597–1603 (2005)

Do, K., Muller, P., Tang, F.: A Bayesian mixture model for differential gene expression. J. R. Stat. Soc. C **54**, 627–644 (2005)

Efron, B.: Large-scale simultaneous hypothesis testing: the choice of a null hypothesis. J. Am. Stat. Assoc. **99**(465), 96–104 (2004)

Fearnhead, P.: MCMC, sufficient statistics and particle filter. J. Comput. Graph. Stat. **11**, 848–862 (2002)

Fearnhead, P.: Exact Bayesian curve fitting and signal segmentation. IEEE Trans. Signal Process. **53**, 2160–2166 (2005)

Fearnhead, P.: Exact and efficient Bayesian inference for multiple changepoint problems. Stat. Comput. **16**, 203–213 (2006)

Fearnhead, P., Liu, Z.: On-line inference for multiple change points problems. J. R. Stat. Soc. B **69**, 589–605 (2007)

Gentleman, R.C., Carey, V.J., Bates, D.M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R.A., Leisch, F., Li, C., Maechler, M., Rossini, A.J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J.Y.H., Zhang, J.: Bioconductor: open software development for computational biology and bioinformatics. Genome Biol. **5**(10), R80 (2004)

Gottardo, R., Pannucci, J.A., Kuske, C.R., Brettin, T.S.: Statistical analysis of microarray data: a Bayesian approach. Biostatistics **4**(4), 597–620 (2003)

Gottardo, R., Raftery, A.E., Yeung, K.Y., Bumgarner, R.E.: Bayesian robust inference for differential gene expression in microarrays with multiple samples. Biometrics **62**(1), 10–18 (2006)

Huber, W., Toedling, J., Steinetz, L.M.: Transcript mapping with high-density oligonucleotide tiling arrays. Bioinformatics **22**(16), 1963–1970 (2006)

Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U., Speed, T.P.: Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics **4**(2), 249–264 (2003)

Johnson, T., Elashoff, R., Harkema, S.: A Bayesian changepoint analysis of electromyographic data: detecting muscle activation patterns and associated applications. Biostatistics **4**, 143–164 (2003)

Kapranov, P., Cawley, S.E., Drenkow, J., Bekiranov, S., Strausberg, R.L., Fodor, S.P.A., Gingeras, T.R.: Large-scale transcriptional activity in chromosomes 21 and 22. Science **296**(5569), 916–919 (2002)

Kass, R., Raftery, A.: Bayes factors. J. Am. Stat. Assoc. **90**, 773–795 (1995)

Kendziorski, C.M., Newton, M.A., Lan, H., Gould, M.N.: On parametric empirical Bayes methods for comparing multiple groups using replicated gene expression profiles. Stat. Med. **22**(24), 3899–3914 (2003)

Lockhart, D.J., Dong, H., Byrne, M.C., Follettie, M.T., Gallo, M.V., Chee, M.S., Mittmann, M., Wang, C., Kobayashi, M., Horton, H., Brown, E.L.: Expression monitoring by hybridization to high-density oligonucleotide arrays. Nat. Biotechnol. **14**(13), 1675–1680 (1996)

Newton, M.A., Kendziorski, C.M., Richmond, C.S., Blattner, F.R., Tsui, K.W.: On differential variability of expression ratios: improving statistical inference about gene expression changes from microarray data. J. Comput. Biol. **8**(1), 37–52 (2001)

Poyiadjis, G., Doucet, A., Singh, S.S.: Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. Biometrica **98**(1), 65–80 (2011)

Stephens, D.: Bayesian retrospective multiple-changepoint identification. Appl. Stat. **43**, 159–178 (1994)

Xuan, X., Murphy, K.: Modeling changing dependency structure in multivariate time series. In: International Conference on Machine Learning (2007)