An EM and a stochastic version of the EM algorithm for nonparametric Hidden semi-Markov models

Sonia Malefaki, Samis Trevezas* and Nikolaos Limnios

Laboratoire de Mathématiques Appliquées de Compiègne, UTC, France

Abstract

The Hidden semi-Markov models (HSMMs) have been introduced to overcome the constraint of a geometric sojourn time distribution for the different hidden states in the classical hidden Markov models. Several variations of HSMMs have been proposed that model the sojourn times by a parametric or a nonparametric family of distributions. In this article, we concentrate our interest on the nonparametric case where the duration distributions are attached to transitions and not to states as in most of the published papers in HSMMs. Therefore, it is worth noticing that here we treat the underlying hidden semi–Markov chain in its general probabilistic structure. In that case, Barbu and Limnios (2008) proposed an Expectation–Maximization (EM) algorithm in order to estimate the semi-Markov kernel and the emission probabilities that characterize the dynamics of the model. In this paper, we consider an improved version of Barbu and Limnios' EM algorithm which is faster than the original one. Moreover, we propose a stochastic version of the EM algorithm that achieves comparable estimates with the EM algorithm in less execution time. Some numerical examples are provided which illustrate the efficient performance of the proposed algorithms.

Key words and phrases: Hidden semi-Markov models; Maximum likelihood estimation; EM algorithm; Stochastic EM algorithm.

1 Introduction

The last decades, the Hidden Markov Models (HMMs) have become one of the most powerful and popular techniques in several scientific fields such as speech recognition [Rabiner (1989), Rabiner and Juang (1993)], biology [Krogh et al. (1994a,b)], image processing [Li and Gray (2000)] and several other fields [Bhar and Hamori (2004), Sansom (1998)]. The interested reader can find a well-documented review in Ephraim and Merhav (2002). For an overview of recent advances in HMMs see Cappé et al. (2005).

The first who made statistical inference for HMMs were Baum and Petrie (1966) and since then they are widely used. In the original setting a HMM consists of a bidimensional stochastic process $(Z_n, Y_n)_{n\in\mathbb{N}}$, where the first component forms a finite Markov chain (MC) not directly observed and the second one conditioned on the MC forms a sequence of conditionally independent random variables (r.v.) with a finite alphabet. By observing a trajectory of a certain length the main goal is to estimate the transition probabilities of the MC and the conditional distributions that characterize the relationship between the observable and the hidden process.

Although the HMMs can be proved useful in several cases, they suffer from an important limitation. They do not allow other distribution for the sojourn times in the states of the hidden process than the geometric

*Corresponding author; Address: Laboratoire de Mathématiques Appliquées de Compiègne, UTC, France, e–mail: samis.trevezas@utc.fr one. Such a model can be proved improper to describe several real data problems [see Levinson (1986)]. A natural generalization of the HMMs are the Hidden Semi-Markov Models (HSMMs). In this case, the hidden process $(Z_n)_{n\in\mathbb{N}}$, is no longer a MC but a semi-Markov chain (SMC). Contrary to HMMs, the HSMMs allow any distribution (beyond the geometric) for the sojourn times in the different states of the hidden process, in order to achieve a better description of a real problem dataset.

HSMMs were introduced by Ferguson (1980). Since then, several variations of HSMMs have been proposed that model the sojourn times by a parametric or a nonparametric family of distributions. In any case we are interested in estimating the semi-Markov kernel (SM kernel) that governs the evolution of the hidden semi-Markov chain and the conditional distributions that generate the observed data. In order to obtain the maximum likelihood estimator (MLE) of the parameters of the HSMMs, an Expectation–Maximization (EM) algorithm can be employed, due to the incompleteness of the data. Ferguson (1980) proposed an EM algorithm for a HSMM. Since then, a lot of research has been done in this direction. Levinson (1986), Guédon and Cocozza-Thivent (1990), Durbin et al. (1998), Sansom and Thomson (2001), Guédon (2003, 2005), Bulla and Bulla (2006) and Barbu and Limnios (2008) are few typical references that presented EM algorithms for parametric and nonparametric cases. Moreover, Barbu and Limnios (2006, 2008) proved consistency and asymptotic normality of the MLE for the nonparametric HSMM. Trevezas and Limnios (2009) proved consistency and asymptotic normality of the MLE for the general HSMM with backward recurrence time dependence, where the emission law of every observed state, conditioned on the SMC, depend not only on the underlying state of the SMC but also on the time that has elapsed from the last jump of the SMC.

In this paper we concentrate our interest on the nonparametric MLE of the HSMMs in their full generality, where the sojourn (duration) times are attached to transitions and not to states. We will present an improved EM algorithm that runs in significantly less time than the original Barbu and Limnios' EM algorithm [Barbu and Limnios (2008)]. Moreover, we propose stochastic versions of the EM, that give additional computational advantages. In order to illustrate the performance of the proposed algorithms we used simulated data. The stochastic version of the EM algorithm gives almost the same estimates as EM at the same level of accuracy but it converges faster and needs even less CPU time per iteration than the proposed EM algorithm. This offers the opportunity to use even longer trajectories in order to have better accuracy in a reasonable time period, something really useful especially for real data problems.

The rest of this paper is organized as follows: In Section 2, we introduce the mathematical notations and we state the conditions in order to specify the subclass of HSMMs to be considered. In Section 3, we present an EM algorithm for the nonparametric HSMMs. The proofs of the Forward-Backward algorithm are presented in Section 4. In Section 5, we present a class of stochastic EM algorithms for the nonparametric HSMMs. In Section 6, the performance of the proposed algorithms is illustrated in several examples. Finally, we conclude by providing a short discussion.

2 Preliminaries and assumptions

Let the couple $(\mathbf{Z},\mathbf{Y}) := (Z_n, Y_n)_{n\in\mathbb{N}}$ be a hidden semi-Markov chain defined on a probability space $(\Omega, \mathscr{A}, \mathbb{P}_{\boldsymbol{\theta}})$, where $\boldsymbol{\theta} \in \Theta$, and Θ is a euclidean subset that parametrizes our model and will be specified in the sequel. We assume that the SMC $\mathbf{Z}$ has finite state space $E = \{1, 2, \dots, s\}$ and SM kernel

$\boldsymbol{q}^{\boldsymbol{\theta}} := (q_{ij}^{\boldsymbol{\theta}}(k))_{i,j\in E,k\in\mathbb{N}}$. If we denote by $(\mathbf{J},\mathbf{S}) := (J_l, S_l)_{l\in\mathbb{N}}$ the associated Markov renewal process to $\mathbf{Z}$, then for each $l \in \mathbb{N}$,

$$q_{ij}^{\boldsymbol{\theta}}(k) = \mathbb{P}_{\boldsymbol{\theta}}(J_{l+1} = j, S_{l+1} - S_l = k \mid J_l = i).$$

The process $\mathbf{S}$ keeps track of the successive time points that changes of states in $\mathbf{Z}$ occur (jump times), and $\mathbf{J}$ records the visited states at these time points (embedded MC). By convention, we take $S_0 = 0$, i.e., the initial observation is taken on a jump time, and also $q_{ii}^{\boldsymbol{\theta}}(k) = 0$, i.e., we do not allow direct transitions of $\mathbf{J}$ to the same state (self-transitions). We note here that the exclusion of self-transitions (virtual transitions) is always possible when we have irreducible SMCs, by considering a simple transformation of the semi-Markov kernel [Pyke (1961)]. Let also $\mathbf{N} := (N(n))_{n\in\mathbb{N}}$ be the discrete time counting process of the number of jumps of $\mathbf{Z}$ until time n, where

$$N(n) = \max\{l \in \mathbb{N} \mid S_l \leq n\}.$$

It is clear that $S_{N(n)}$ expresses the time that the last jump before time n took place. The sequence $\mathbf{X} := (X_l)_{l\in\mathbb{N}}$, where $X_0 = 0$ and $X_l = S_l - S_{l-1}$ for $l \in \mathbb{N}^*$ represents the sojourn times in the successively visited states. Additionally, we will use the notation $\mathbf{Z}_{n_1}^{n_2}$ to denote the vector $(Z_{n_1}, Z_{n_1+1}, \ldots, Z_{n_2})$, for $n_1 \leq n_2$, and $\boldsymbol{i}$ for a vector that every component equals to the element $i \in E$. The dimension of $\boldsymbol{i}$ is implied by the dimension of $\mathbf{Z}_{n_1}^{n_2}$. The observable process $\mathbf{Y}$ has finite state space $A = \{1, \ldots, d\}$ and given the process $\mathbf{Z}$ and the observed vector $\mathbf{Y}_0^{n-1}$, the conditional probability function of Y_n depends only on Z_n, that is, for any $n \in \mathbb{N}$, $\alpha \in A$ and $i \in E$, we have

$$\mathbb{P}_{\boldsymbol{\theta}}(Y_n = \alpha \mid Z_n = i, \mathbf{Z}_0^{n-1}, \mathbf{Y}_0^{n-1}) = \mathbb{P}_{\boldsymbol{\theta}}(Y_n = \alpha \mid Z_n = i) := R_{i;\alpha}^{\boldsymbol{\theta}}.$$

For each $i \in E$, we denote by $\boldsymbol{R}_i^{\boldsymbol{\theta}} := (R_{i;\alpha}^{\boldsymbol{\theta}})_{\alpha\in A}$ the vector of emission probabilities from the hidden states to the observed states, and $\boldsymbol{R}^{\boldsymbol{\theta}} := (\boldsymbol{R}_i^{\boldsymbol{\theta}})_{i\in E}$.

We state the following conditions concerning the subclass of HSMMs to be considered:

(A1) The parameter of the model is selected to be $\boldsymbol{\theta} = (\boldsymbol{q}, \boldsymbol{R})$, and therefore we choose a nonparametric framework and identify $\boldsymbol{q}^{\boldsymbol{\theta}}$ and $\boldsymbol{R}^{\boldsymbol{\theta}}$ with $\boldsymbol{q}$ and $\boldsymbol{R}$ respectively.

(A2) If we denote by $\boldsymbol{\theta}^{(0)}$ the initial value of the EM and the stochastic version of EM algorithm that we present in the sequel, then under $\mathbb{P}_{\boldsymbol{\theta}^{(0)}}$ the corresponding constants $\tilde{n}_{ij} := \max\{k \in \mathbb{N} : q_{ij}^{(0)}(k) > 0\}$, should not exceed the observation length M.

(A3) The embedded MC $\mathbf{J}$ is irreducible.

Remarks: i) By condition (A1) we make clear our intention to deal only with nonparametric cases, where no knowledge is available for the underlying distributions. Condition (A2) implies that the conditional sojourn time distributions for the different hidden states, conditioned on the next visited state, are considered initially as discrete distributions concentrated on finite sets of time points. Condition (A3) renders the SMC $\mathbf{Z}$ irreducible as well. By using conditions (A1) and (A2) we obtain the dependence relations for the parameters, where for each $i \in E$ we have :

$$\sum_{j\neq i}\sum_{k=1}^{\tilde{n}_{ij}} q_{ij}(k) = 1 \quad \text{and} \quad \sum_{\alpha\in A} R_{i;\alpha} = 1. \tag{1}$$

ii) If we denote by $\tilde{n}_i := \max_j \tilde{n}_{ij}$, then for each $i \in E$ the constants $\tilde{n}_i$ express the maximum time period of the sojourn times in state i.

We pinpoint that the majority of works that concern HSMMs treat the case where the sojourn times are attached to states [see for example Guédon (2003, 2007), Bulla and Bulla (2006)]. We illustrate the difference by describing the dynamics of the first transition for the SMC $\mathbf{Z}$ from a state i to a state j in k time units. Let us define p_{ij} the corresponding transition probability of the embedded MC $\mathbf{J}$ and $f_{ij}(k)$ the conditional probability to have a transition from state i to state j in k time units. Then, we have

$$p_{ij} = \mathbb{P}_{\boldsymbol{\theta}}\left(Z_{S_1} = j \mid Z_0 = i\right) \quad \text{and} \quad f_{ij}(k) = \mathbb{P}_{\boldsymbol{\theta}}\left(S_1 = k \mid Z_{S_1} = j, Z_0 = i\right), \tag{2}$$

where we recall that $S_0 = 0$.

When sojourn times are attached to transitions the following decomposition holds for the elements of the SM kernel:

$$\begin{aligned} q_{ij}(k) &= \mathbb{P}_{\boldsymbol{\theta}}\left(Z_k = j, \mathbf{Z}_1^{k-1} = \boldsymbol{i} \mid Z_0 = i\right) = \mathbb{P}_{\boldsymbol{\theta}}\left(Z_{S_1} = j, S_1 = k \mid Z_0 = i\right) \\ &= \mathbb{P}_{\boldsymbol{\theta}}\left(Z_{S_1} = j \mid Z_0 = i\right) \mathbb{P}_{\boldsymbol{\theta}}\left(S_1 = k \mid Z_{S_1} = j, Z_0 = i\right) = p_{ij} f_{ij}(k), \end{aligned}$$

whereas in the case that sojourn times are attached to states the probability $f_{ij}(k)$ does not depend on the next visited state j, therefore leading to the decomposition

$$q_{ij}(k) = \mathbb{P}_{\boldsymbol{\theta}}\left(Z_{S_1} = j \mid Z_0 = i\right) \mathbb{P}_{\boldsymbol{\theta}}\left(S_1 = k \mid Z_0 = i\right) = p_{ij} f_i(k).$$

Note that in this framework when a sojourn time in a state i expires, we can determine the next visited state j by using only the probabilities of the embedded MC, whereas in the framework of attached transitions, the next visited state depends also on the duration of this time. This shows that the HSMMs with duration times attached to transitions are more general. Nevertheless, for many applications the dependence on the next visited state could be a restriction since we have an important increase in the number of unknown parameters.

3 An improved EM algorithm for nonparametric HSMMs

The EM algorithm is a very popular estimation method that has been applied in several scientific fields. Its present general form was given by Dempster et al. (1977) although it has already been mentioned earlier in the literature as a method for estimation in HMMs [Baum et al. (1970)]. It consists an iterative, deterministic method designed to find the MLE in incomplete data problems. Each iteration of the EM algorithm comprises two steps. The first step is the expectation step (E–step), where the conditional expectation of the complete data log–likelihood conditioned on the data and the current parameter estimate (called Q–function) is calculated. The second step is the maximization step (M–step), where the parameters are updated by maximizing the Q–function of the E–step. Thus, by starting from an arbitrary initial value, under some regularity conditions, the EM algorithm produces a convergent sequence of parameter estimates [Boyles (1983), Wu (1983)]. For an extensive literature on the EM, see McLachlan and Krishnan (2008) and the references therein.

In this section we present an EM algorithm for the nonparametric HSMMs, which are defined in Section 2 and satisfy conditions (A1)–(A3), that consists an improvement of the corresponding algorithm of Barbu and Limnios (2008). This improvement can be summarized as follows:

i) This version of the EM is significantly faster. Indeed, the complexity of our proposed Forward–Backward algorithm is $O(M^2s^2)$ in time in the worst case, instead of $O(M^3s^2)$ in Barbu and Limnios' algorithm. In the case of HSMMs with duration times attached to states, Guédon (2003) derived an EM algorithm with complexity $O(M^2s + s^2M)$. The difference in time complexity from the proposed algorithm is due to the additional dependence of the sojourn time in the hidden states on the next visited state, something that can not be relaxed by the nature of our model, since we treat a more general dependence structure.

ii) Instead of using an approached MLE by taking the approached likelihood function, we derive the exact MLE that we obtain directly from the likelihood function without neglecting the part that corresponds to the sojourn time in the last visited state. For the derivation of the exact MLE in the case of HSMMs that duration times are attached to states see Guédon (2003) and Bulla (2006).

iii) The algorithm of Barbu and Limnios does not take into account the support of the conditional distributions that appear in the initial value $\boldsymbol{\theta}^{(0)}$. A direct transition from a state i to a state j of the SMC $\mathbf{Z}$, occurs at a maximum period of $\tilde{n}_{ij}$ time units. Therefore, the support of the transition laws should be limited by this barrier during all iterations. This fact reduces considerably the execution time of the algorithm.

We note also here that the construction of the initial algorithm of Barbu and Limnios is based on the parameterization $(\boldsymbol{p}, \boldsymbol{f})$ of the SM kernel $\boldsymbol{q}$, where $\boldsymbol{p}$ is the vector of transition probabilities of the embedded MC $\boldsymbol{J}$ and $\boldsymbol{f}$ is the probability vector that characterize the conditional sojourn times in the different hidden states and are given by (2). In our analysis, we use directly the SM kernel.

More specifically, in the HSMMs case, based on a sample sequence $\{\mathbf{Y}_0^M = \mathbf{y}_0^M\}$, our main goal is to find the MLE of the parameter $\boldsymbol{\theta} = (\mathbf{q}, \mathbf{R})$ of the model. The likelihood function $\mathcal{L}_M(\boldsymbol{\theta}) := \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_0^M, \mathbf{Y}_0^M)$ of the complete data has the form

$$\mathcal{L}_M(\boldsymbol{\theta}) = a(Z_0) \left(\prod_{l=1}^{N(M)} q_{J_{l-1}J_l}(X_l) \right) \left(\prod_{n=0}^{M} R_{Z_n;Y_n} \right) \bar{H}_{J_{N(M)}}(M - S_{N(M)}),$$

where $a(\cdot)$ is the initial distribution and $\bar{H}_i(\cdot)$ is the survival function in state i for the SMC $\mathbf{Z}$, and for $u \in \mathbb{N}$ is given by

$$\bar{H}_i(u) = 1 - \sum_{j \neq i} \sum_{k=1}^{u} q_{ij}(k). \tag{3}$$

Now, we define the following counting processes:

$$\begin{aligned} N_{ij}(k, M) \quad &:= \quad \sum_{n=k-1}^{M-1} \mathbb{1}_{\{Z_{n+1}=j, \mathbf{Z}_{n-k+1}^{n}=\boldsymbol{i}, Z_{n-k}\neq i\}},\ i, j \in E,\ i \neq j,\ 1 \leq k \leq \widetilde{n}_{ij}, \\ N_{i;\alpha}(M) \quad &:= \quad \sum_{n=0}^{M} \mathbb{1}_{\{Z_n=i, Y_n=\alpha\}},\ i \in E,\ \alpha \in A, \end{aligned}$$

where by convention $\{Z_0 = i, Z_{-1} \neq i\} = \{Z_0 = i\}$, well adapted to our assumption that $S_0 = 0$. We can reformulate the likelihood function as follows:

$$\mathcal{L}_M(\boldsymbol{\theta}) = a(Z_0) \left(\prod_{\substack{i,j,k \\ i \neq j}} q_{ij}(k)^{N_{ij}(k,M)} \right) \left(\prod_{i,\alpha} R_{i;\alpha}^{N_{i;\alpha}(M)} \right) \bar{H}_{Z_M}(M - S_{N(M)}),$$

where $S_{N(M)}$ is the time of the last jump of $\mathbf{Z}$ before time M.

Thus, the log–likelihood $\ell_M(\boldsymbol{\theta}) := \log \mathcal{L}_M(\boldsymbol{\theta})$ of the complete data is then equal to

$$\begin{aligned} \ell_M(\boldsymbol{\theta}) = \quad & \log a(Z_0) + \sum_{\substack{i,j,k \\ i \neq j}} N_{ij}(k,M) \log q_{ij}(k) + \sum_i \sum_\alpha N_{i;\alpha}(M) \log R_{i;\alpha} \\ & + \log \bar{H}_{Z_M}(M - S_{N(M)}). \end{aligned}$$

Instead of maximizing the log–likelihood of the complete data which is not known, the EM algorithm maximizes the conditional expectation of the log–likelihood of the complete data conditioned on the event $\{\mathbf{Y}_0^M = \mathbf{y}_0^M\}$ and on the current value $\boldsymbol{\theta}^{(m)}$ of the parameter. For simplicity we refer to the above event by $\mathbf{y}$ when it causes no confusion. The first term corresponds to the initial distribution and since in our context does not depend on $\boldsymbol{\theta}$ it can be excluded from the maximization function. Consequently, the function that has to be maximized is given by:

$$\begin{aligned} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) \quad = \quad & \sum_{\substack{i,j,k \\ i \neq j}} \log q_{ij}(k) \sum_{n=k-1}^{M-1} \mathbb{P}_{\boldsymbol{\theta}^{(m)}} \left(Z_{n+1} = j, \mathbf{Z}_{n-k+1}^n = \boldsymbol{i}, Z_{n-k} \neq i | \mathbf{y} \right) \\ & + \sum_i \sum_\alpha \log R_{i;\alpha} \sum_{n=0}^{M} \mathbb{1}_{\{Y_n = \alpha\}} \mathbb{P}_{\boldsymbol{\theta}^{(m)}} \left(Z_n = i \mid \mathbf{y} \right) \\ & + \mathbb{E}_{\boldsymbol{\theta}^{(m)}} \left(\log \bar{H}_{Z_M}(M - S_{N(M)}) \mid \mathbf{y} \right). \end{aligned} \tag{4}$$

Barbu and Limnios (2008) used an approached MLE by considering an approached likelihood function $\tilde{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$ that results from $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$ by neglecting the last term of the righthand member of equation (4). This last term involves the survival function at the last visited hidden state and for ergodic systems, for a large M, its contribution to the likelihood function is small. Therefore, for HSMMs that the survival functions of the sojourn times in the different hidden states decrease rapidly and the observation length is big, an approached MLE can be satisfactory. For the maximization of $\tilde{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$ and the corresponding approached MLE of the SM kernel and the emission probabilities see pp. 157-158, Barbu and Limnios (2008). Nevertheless, it would be wishful to derive the exact MLE by maximizing directly equation (4). For this reason notice that

$$\mathbb{E}_{\boldsymbol{\theta}^{(m)}} \big(\log \bar{H}_{Z_M}(M - S_{N(M)}) \mid \mathbf{y} \big) = \sum_i \sum_{u=1}^{\widetilde{n}_i - 1} \log \bar{H}_i(u) \, \mathbb{P}_{\boldsymbol{\theta}^{(m)}} (\mathbf{Z}_{M-u}^M = \boldsymbol{i}, Z_{M-u-1} \neq i \mid \mathbf{y}).$$

Since this conditional expectation depends on survival functions, and each survival function by (3) depends only on $\boldsymbol{q}$, we have

$$Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) = Q_1(\boldsymbol{q} \mid \boldsymbol{\theta}^{(m)}) + Q_2(\boldsymbol{R} \mid \boldsymbol{\theta}^{(m)}),$$

where

$$\begin{aligned}
Q_1(\boldsymbol{q} \mid \boldsymbol{\theta}^{(m)}) &= \sum_{\substack{i,j,k \\ i\neq j}} \log q_{ij}(k) \sum_{n=k-1}^{M-1} \mathbb{P}_{\boldsymbol{\theta}^{(m)}}\left(Z_{n+1}=j, \mathbf{Z}_{n-k+1}^{n}=\boldsymbol{i}, Z_{n-k}\neq i \mid \mathbf{y}\right) \\
&\quad + \sum_{i} \sum_{u=1}^{\widetilde{n}_i-1} \log \bar{H}_i(u)\, \mathbb{P}_{\boldsymbol{\theta}^{(m)}}(\mathbf{Z}_{M-u}^{M}=\boldsymbol{i}, Z_{M-u-1}\neq i \mid \mathbf{y}), \\
Q_2(\boldsymbol{R} \mid \boldsymbol{\theta}^{(m)}) &= \sum_{i}\sum_{\alpha} \log R_{i;\alpha} \sum_{n=0}^{M} \mathbb{1}_{\{Y_n=\alpha\}} \mathbb{P}_{\boldsymbol{\theta}^{(m)}}\left(Z_n=i \mid \mathbf{y}\right).
\end{aligned}$$

Thus, starting from an arbitrary initial value $\boldsymbol{\theta}^{(0)}$, at each EM step, the conditional probabilities needed above are calculated by a Forward–Backward algorithm which will be presented in the rest of this section and the updated value $\boldsymbol{\theta}^{(m+1)}$ is obtained by maximizing the Q–function with respect to $\boldsymbol{\theta}$.

In the sequel, we define some quantities upon which will be based the recursive relations for the EM algorithm and its stochastic version. In particular, for the forward procedure we will need for each $n \in \mathbb{N}$ and $i \in E$:

$$\begin{aligned}
P_0 &:= \mathbb{P}_{\boldsymbol{\theta}}(Y_0=y_0), \\
P_n &:= \mathbb{P}_{\boldsymbol{\theta}}(Y_n=y_n \mid \mathbf{y}_0^{n-1}),\ n\neq 0, \\
B_n(i) &:= \mathbb{P}_{\boldsymbol{\theta}}(Z_n=i, Z_{n-1}\neq i, Y_n=y_n \mid \mathbf{y}_0^{n-1}). \qquad (5)
\end{aligned}$$

For the backward procedure we will need additionally for each $n \in \mathbb{N}$, $u \in \mathbb{N}^*$ and $i \in E$ the following posterior probabilities:

$$\begin{aligned}
L_n(i) &:= \mathbb{P}_{\boldsymbol{\theta}}(Z_n=i \mid \mathbf{y}), \\
L_{1;n}(i) &:= \mathbb{P}_{\boldsymbol{\theta}}(Z_n=i, Z_{n-1}\neq i \mid \mathbf{y}), \\
L_{1;n}(i,u) &:= \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u}^{n}=\boldsymbol{i}, Z_{n-u-1}\neq i \mid \mathbf{y}), \qquad (6) \\
L_{2;n}(i) &:= \mathbb{P}_{\boldsymbol{\theta}}(Z_n=i, Z_{n-1}=i \mid \mathbf{y}),\ n\neq 0, \qquad (7)
\end{aligned}$$

and for $i \neq j$ the auxiliary quantity

$$G_n(i,j,u) := \mathbb{P}_{\boldsymbol{\theta}}(Z_{n+1}=j, \mathbf{Z}_{n-u}^{n}=\boldsymbol{i}, Z_{n-u-1}\neq i \mid \mathbf{y}). \qquad (8)$$

Moreover, we denote for each $i \in E$ and $\alpha \in A$, the initial probabilities of the couple $(\mathbf{Z}, \mathbf{Y})$

$$\mu(i,\alpha) := \mathbb{P}_{\boldsymbol{\theta}}(Z_0=i, Y_0=\alpha),$$

which will be identified with the term $B_0(i)$.

In order to simplify the notation for the products of the form $\prod_{p=n-k+1}^{n} R_{i;y_p}$ that will appear in the sequel we define the term

$$T_{n,k}(i) := \prod_{p=n-k+1}^{n} R_{i;y_p},$$

and also we use the standard notation $x \wedge y = \min\{x,y\}$. For empty sums and products we assume that they are equal to zero and one respectively.

In the rest of this section we will present a detailed description of the EM algorithm for the HSMM case.

Estimation step

Forward step

- For $n = 0$, we have $\quad B_0(i) = \mu(i, y_0), \quad P_0 = \sum_i B_0(i).$

- For $n = 1, \dots, M$, we have

$$B_n(i) = \sum_{j \neq i} \sum_{t=1}^{\widetilde{n}_{ji} \wedge n} \frac{q_{ji}(t) B_{n-t}(j) R_{i,y_n} T_{n-1,t-1}(j)}{\prod_{p=n-t}^{n-1} P_p}, \tag{9}$$

$$P_n = \sum_i \sum_{u=0}^{(\widetilde{n}_i - 1) \wedge n} \frac{\bar{H}_i(u) B_{n-u}(i) T_{n,u}(i)}{\prod_{p=n-u}^{n-1} P_p}. \tag{10}$$

Backward step

- For $n = M$, we have $\quad L_{1;M}(i) = B_M(i)/P_M, \quad L_{1;M}(i, \widetilde{n}_i) = 0.$

  □ For $u = 1, \dots, \widetilde{n}_i - 1$,

$$L_{1;M}(i,u) = \frac{\bar{H}_i(u) B_{M-u}(i)\, T_{M,u}(i)}{\prod_{p=M-u}^{M} P_p}, \tag{11}$$

$$L_{2;M}(i) = \sum_{u=1}^{\widetilde{n}_i - 1} L_{1;M}(i,u), \tag{12}$$

$$L_M(i) = L_{1;M}(i) + L_{2;M}(i). \tag{13}$$

- For $n = M-1, M-2, \dots, 0$, we have

  □ For $u = 0, \dots, \widetilde{n}_{ij} - 1$ and $u \leq n$,

$$G_n(i,j,u) = \frac{q_{ij}(u+1) B_{n-u}(i) L_{1;n+1}(j) R_{j,y_{n+1}} T_{n,u}(i)}{B_{n+1}(j) \prod_{p=n-u}^{n} P_p}. \tag{14}$$

  □ For $u = 1, \dots, \widetilde{n}_i - 1$ and $u \leq n$,

$$L_{1;n}(i,u) = L_{1;n+1}(i, u+1) + \sum_{\substack{j:\, j \neq i \\ \tilde{n}_{ij} \geq u+1}} G_n(i,j,u), \tag{15}$$

$$L_{1;n}(i, \widetilde{n}_i) = 0, \tag{16}$$

$$L_{1;n}(i) = L_{1;M}(i, M-n) + \sum_{j \neq i} \sum_{v=1}^{\widetilde{n}_{ij} \wedge (M-n)} G_{n+v-1}(i,j,v-1), \tag{17}$$

$$L_{2;n}(i) = \sum_{u=1}^{(\widetilde{n}_i - 1) \wedge (n-1)} L_{1;n}(i,u), \tag{18}$$

$$L_n(i) = L_{1;n}(i) + L_{2;n}(i). \tag{19}$$

Maximization step

The iterative procedure of the estimation of the parameter is given as follows:

$$q_{ij}^{(m+1)}(k) = \left[\prod_{u=1}^{k-1}\left(1+\frac{L_{1;M}^{(m)}(i,u)}{\sum_{n=0}^{M-1}L_{1;n}^{(m)}(i,u)}\right)\right]\frac{\sum_{n=0}^{M-1}G_n^{(m)}(i,j,k-1)}{\sum_{n=0}^{M-1}L_{1;n}^{(m)}(i)}, \tag{20}$$

$$R_{i;\alpha}^{(m+1)} = \frac{\sum_{n=0}^{M}\delta_{y_n\alpha}L_n^{(m)}(i)}{\sum_{n=0}^{M}L_n^{(m)}(i)}. \tag{21}$$

The complexity of the proposed Forward–Backward algorithm is $O(M^2s^2)$ in time and $O(Ms)$ in space in the worst case. It is significantly faster than the one which is proposed by Barbu and Limnios (2008), since its complexity in time is $O(M^3s^2)$.

Remark: (SM-M1 case) In many applications it is more natural to assume that the observed sequence $\mathbf{Y}$ conditioned on $\mathbf{Z}$ forms a sequence of Markovian dependent r.v.. It is straightforward to see that the corresponding Q-function that we denote by Q_d similarly to (4) equals

$$\begin{aligned}Q_d(\boldsymbol{\theta}\mid\boldsymbol{\theta}^{(m)}) &= \sum_{\substack{i,j,k\\ i\neq j}}\log q_{ij}(k)\sum_{n=k-1}^{M-1}\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(Z_{n+1}=j,\mathbf{Z}_{n-k+1}^{n}=\boldsymbol{i},Z_{n-k}\neq i|\mathbf{y})\\ &\quad+\sum_i\sum_{\alpha,\beta}\log R_{i;\alpha,\beta}\sum_{n=1}^{M}\mathbb{1}_{\{Y_{n-1}=\alpha,Y_n=\beta\}}\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(Z_n=i\mid\mathbf{y})\\ &\quad+\mathbb{E}_{\boldsymbol{\theta}^{(m)}}\left(\log\bar{H}_{Z_M}(M-S_{N(M)})\mid\mathbf{y}\right).\end{aligned}$$

The iterative procedure of the estimation of the parameter by maximizing the function Q_d, given as above, is the same for the semi-Markov kernel for both cases (see relation (20)), while for the emission probabilities gives :

$$R_{i;\alpha,\beta}^{(m+1)} = \frac{\sum_{n=1}^{M}\mathbb{1}_{\{Y_{n-1}=\alpha,Y_n=\beta\}}\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(Z_n=i\mid\mathbf{y})}{\sum_{n=1}^{M}\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(Z_n=i\mid\mathbf{y})}.$$

This enables us to construct the EM algorithm for the conditional Markovian case as a simple extension of the conditionally independent case (compare with (31)) and therefore we will not enter into details.

4 Proofs for the Forward–Backward algorithm

We will prove the expression (9) referring to the term $B_n(i)$ which is given by (5). For $i,j\in E,\ i\neq j,\ u\in\mathbb{N},\ v\in\mathbb{N}^*$, we define

$$B_n(i,j,u,v):=\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+v}=j,\mathbf{Z}_{n-u}^{n+v-1}=\boldsymbol{i},Z_{n-u-1}\neq i,y_{n-u}\mid\mathbf{y}_0^{n-u-1}).$$

Note that

$$\begin{aligned}B_n(i,j,u,v) &= \mathbb{P}_{\boldsymbol{\theta}}(Z_{n-u}=i,Z_{n-u-1}\neq i,y_{n-u}\mid\mathbf{y}_0^{n-u-1})\\ &\quad\times\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+v}=j,\mathbf{Z}_{n-u+1}^{n+v-1}=\boldsymbol{i}\mid Z_{n-u}=i,Z_{n-u-1}\neq i,\mathbf{y}_0^{n-u})\\ &= B_{n-u}(i)q_{ij}(v+u),\end{aligned} \tag{22}$$

since the probability $\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+v} = j, \mathbf{Z}_{n-u+1}^{n+v-1} = \boldsymbol{i} \mid Z_{n-u} = i, Z_{n-u-1} \neq i, \mathbf{y}_0^{n-u})$ does not depend on the vector $\mathbf{y}_0^{n-u}$.

For $n = 1, \ldots, M$, and $i \in E$, we have

$$\begin{aligned}
B_n(i) &= \mathbb{P}_{\boldsymbol{\theta}}(Z_n = i, Z_{n-1} \neq i, y_n \mid \mathbf{y}_0^{n-1}) \\
&= \sum_{j \neq i} \sum_{t=1}^{\widetilde{n}_{ji} \wedge n} \mathbb{P}_{\boldsymbol{\theta}}(Z_n = i, \mathbf{Z}_{n-t}^{n-1} = \boldsymbol{j}, Z_{n-t-1} \neq j, y_n \mid \mathbf{y}_0^{n-1}) \\
& \sum_{j \neq i} \sum_{t=1}^{\widetilde{n}_{ji} \wedge n} \frac{\mathbb{P}_{\boldsymbol{\theta}}(Z_n = i, \mathbf{Z}_{n-t}^{n-1} = \boldsymbol{j}, Z_{n-t-1} \neq j, y_{n-t}, \mathbf{y}_{n-t+1}^{n} \mid \mathbf{y}_0^{n-t-1})}{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n-t}^{n-1} = \mathbf{y}_{n-t}^{n-1} \mid \mathbf{y}_0^{n-t-1})} \\
&= \sum_{j \neq i} \sum_{t=1}^{\widetilde{n}_{ji} \wedge n} \frac{B_{n-t}(j, i, 0, t) R_{i, y_n} T_{n-1, t-1}(j)}{\prod_{p=n-t}^{n-1} P_p}. \qquad (23)
\end{aligned}$$

By relations (22) and (23), we get (9).

Now we proceed to analyze the term P_n and for $n = 1, \ldots, M$, we have

$$\begin{aligned}
P_n &= \mathbb{P}_{\boldsymbol{\theta}}(Y_n = y_n \mid \mathbf{y}_0^{n-1}) \\
&= \sum_{i} \sum_{u=0}^{(\widetilde{n}_i - 1) \wedge n} \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u}^{n} = \boldsymbol{i}, Z_{n-u-1} \neq i, y_n \mid \mathbf{y}_0^{n-1}) \\
&= \sum_{i} \sum_{u=0}^{(\widetilde{n}_i - 1) \wedge n} \frac{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u}^{n} = \boldsymbol{i}, Z_{n-u-1} \neq i, y_{n-u}, \mathbf{y}_{n-u+1}^{n} \mid \mathbf{y}_0^{n-u-1})}{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n-u}^{n-1} = \mathbf{y}_{n-u}^{n-1} \mid \mathbf{y}_0^{n-u-1})}. \qquad (24)
\end{aligned}$$

If we denote by $A_n(i, u)$ the numerator of equation (24), we have

$$\begin{aligned}
A_n(i, u) &= \mathbb{P}_{\boldsymbol{\theta}}(Z_{n-u} = i, Z_{n-u-1} \neq i, y_{n-u} \mid \mathbf{y}_0^{n-u-1}) \\
& \quad \times \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u+1}^{n} = \boldsymbol{i} \mid Z_{n-u} = i, Z_{n-u-1} \neq i) \\
& \quad \times \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n-u+1}^{n} = \mathbf{y}_{n-u+1}^{n} \mid \mathbf{Z}_{n-u+1}^{n} = \boldsymbol{i}) \\
&= B_{n-u}(i) \bar{H}_i(u) T_{n,u}(i). \qquad (25)
\end{aligned}$$

By (24) and (25) we deduce the expression (10).

In the sequel, we will give the proofs that correspond to the recursive relations that appear in the backward step. For the validity of (11) note that for $u = 1, \ldots, \widetilde{n}_i - 1$,

$$L_{1;M}(i, u) = \frac{A_M(i, u)}{\prod_{p=M-u}^{M} P_p}.$$

Equations (12) and (13) are special cases of (18) and (19) respectively for $n = M$. Equations (18) and (19) can be verified directly.

Now we proceed to the proof for the expression of $G_n(i, j, u)$ that is given by (14). For $u = 0, \ldots, \widetilde{n}_{ij} - 1$, and $u \leq n$, by (8) we have

$$G_n(i, j, u) = \frac{\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+1} = j, \mathbf{Z}_{n-u}^{n} = \boldsymbol{i}, Z_{n-u-1} \neq i, \mathbf{y})}{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_0^{M} = \mathbf{y})}. \qquad (26)$$

If we denote by N_1 and N_2 the numerator and the denominator respectively of the righthand member of (26), then

$$\begin{aligned} N_1 &= \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_0^{n-u-1} = \mathbf{y}_0^{n-u-1})\mathbb{P}_{\boldsymbol{\theta}}(Z_{n-u} = i, Z_{n-u-1} \neq i, y_{n-u} \mid \mathbf{y}_0^{n-u-1}) \\ &\times\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+1} = j, \mathbf{Z}_{n-u+1}^{n} = \boldsymbol{i} \mid Z_{n-u} = i, Z_{n-u-1} \neq i) \\ &\times\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n-u+1}^{n+1} = \mathbf{y}_{n-u+1}^{n+1} \mid Z_{n+1} = j, Z_{n-u+1}^{n} = \boldsymbol{i}) \\ &\times\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n+2}^{M} = \mathbf{y}_{n+2}^{M} \mid Z_{n+1} = j, Z_n \neq j), \end{aligned} \tag{27}$$

and

$$N_2 = \frac{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_0^n = \mathbf{y}_0^n)\,\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+1} = j, Z_n \neq j, y_{n+1} \mid \mathbf{y}_0^n)\,\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n+2}^M = \mathbf{y}_{n+2}^M \mid Z_{n+1} = j, Z_n \neq j)}{\mathbb{P}_{\boldsymbol{\theta}}(Z_{n+1} = j, Z_n \neq j \mid \mathbf{y})}. \tag{28}$$

Since by (26) the term $G_n(i, j, u) = N_1/N_2$, by combining (27) and (28) we get (14).

In order to deduce equation (15) we remark by (6) and (8) that

$$L_{1;n}(i, u) = \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u}^{n+1} = \boldsymbol{i}, Z_{n-u-1} \neq i \mid \mathbf{y}) + \sum_{j \neq i} G_n(i, j, u), \tag{29}$$

and the first term of the righthand member of the above equality equals $L_{1;n+1}(i, u+1)$.

Finally, we infer relation (17) by using that

$$\begin{aligned} L_{1;n}(i) &= \sum_{j \neq i} \sum_{v=1}^{\widetilde{n}_{ij} \wedge (M-n)} \mathbb{P}_{\boldsymbol{\theta}}(Z_{n+v} = j, Z_n^{n+v-1} = \boldsymbol{i}, Z_{n-1} \neq i \mid \mathbf{y}) \\ &+\mathbb{P}_{\boldsymbol{\theta}}(Z_n^M = \boldsymbol{i}, Z_{n-1} \neq i \mid \mathbf{y}). \end{aligned} \tag{30}$$

At the last part of this section we will justify equations (21) and (20) that refer to the maximization step. The maximization problem of the Q-function can now be decomposed into two separate maximization problems of the Q_1-function and of the Q_2-function in order to obtain the MLE $\boldsymbol{q}^{(m+1)}$ and $\boldsymbol{R}^{(m+1)}$ of $\boldsymbol{q}$ and $\boldsymbol{R}$ respectively. The maximization of Q_2 with respect to $\boldsymbol{R}$ and under the corresponding constraints given by (1) yields easily that

$$R_{i;\alpha}^{(m+1)} = \frac{\sum_{n=0}^{M} \mathbb{1}_{\{Y_n = \alpha\}} \mathbb{P}_{\theta^{(m)}}(Z_n = i \mid \mathbf{y})}{\sum_{n=0}^{M} \mathbb{P}_{\theta^{(m)}}(Z_n = i \mid \mathbf{y})}. \tag{31}$$

This justifies (21). The maximization of the Q_1-function is a more demanding task since the survival functions that appear have to be taken into account as functions of the SM kernel. By using for example the Lagrange method (Lagrange multipliers) it can be verified that the MLE is indeed given by (20).

5 A stochastic EM algorithm for nonparametric HSMMs

Although the EM algorithm is a popular tool, it suffers from some shortcomings such as slow convergence and/or convergence in sub-optimal solutions. An alternative class of algorithms that were proposed to overcome the above problems and also the calculation of intractable, high–dimensional integrals at the E–step, is the class of stochastic EM algorithms. There are many stochastic versions of the EM algorithm but they are all based on the idea to replace the calculation of the expectation in the E–step by a simulation step.

The simplest stochastic version of EM algorithm is the Stochastic EM (SEM), which was proposed by Celeux and Diebolt (1985). The SEM algorithm estimates the Q-function by

$$\hat{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) = \log f(\mathbf{Z}_0^M(\omega), \mathbf{y} \mid \boldsymbol{\theta}),$$

where $\mathbf{Z}_0^M$ is simulated by $\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(\cdot \mid \mathbf{y})$.

Monte Carlo EM (MCEM) [Wei and Tanner (1990)] is a natural generalization of the SEM. In each simulation step m of the MCEM algorithm, the Q-function is estimated by averaging n_m SEM estimates drawn independently, i.e.,

$$\breve{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) = \frac{1}{n_m} \sum_{i=1}^{n_m} \log f(\mathbf{Z}_0^M(\omega_i), \mathbf{y} \mid \boldsymbol{\theta}). \tag{32}$$

So, by the strong law of large numbers $\breve{Q}$ will be a reasonable estimate of Q if the sample size is large enough. Moreover, it is important to mention that MCEM converges only if the Monte Carlo sample size increases in each iteration [see Booth et al. (2001), Chan and Ledolter (1995)]. Thus, the sample size must be determined at each iteration. The problem of selecting a proper sample size at each iteration is data–dependent. Actually, a specific choice of the sequence of n_m could be proper for one problem but really inefficient for another problem. Thus, the need of data–driven sample size rules arises in order to implement MCEM in an automatic way. Several authors have proposed automatic methods in order to specify the proper sample size in each iteration (see Booth and Hobert (1999), Levine and Casella (2001), Levine and Fan (2004), Caffo et al. (2005)), nevertheless it remains a hard task.

Another stochastic version of the EM algorithm is the Stochastic Approximation EM (SAEM) algorithm [Delyon et al. (1999)]. It could be thought of as a generalization of MCEM. In the SAEM the Q-function is estimated recursively by weighting the estimated function of the previous step and a Monte Carlo approximation in the current step. Consequently, the SAEM algorithm uses all the simulated data, contrary to the MCEM that drops the simulated values during the previous iterations and is based only on the current data. More specifically, the Q-function is estimated by

$$\breve{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) = \gamma_m \, \breve{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) + (1 - \gamma_m) \, \breve{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m-1)}),$$

where $\breve{Q}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$ is given by (32).

The choice of the sample size n_m of the simulated data does not affect the convergence of the algorithm, but an appropriate choice of n_m could lead to a faster convergence. In general, as n_m increases, the correlation between successive estimates of the parameters reduces. Thus, a large n_m decreases the number of iterations that the algorithm requires in order to achieve convergence.

The weights γ_m are typically chosen to form a positive decreasing sequence across the iterations and they have crucial importance for the rate of convergence. Polyak and Juditsky (1992) proved that for step size $\gamma_m \propto m^{-\alpha}$ with $1/2 < \alpha \leq 1$, SAEM converges at an optimal rate. Nonetheless, by choosing a large step size ($\alpha \approx 1/2$), the convergence is faster but the Monte Carlo error increases. On the other hand, if a small step size is chosen ($\alpha \approx 1$) the Monte Carlo error decreases but this is also the case for the rate of convergence. A more challenging way for the selection of the sequence of γ_m is proposed by Jank (2006b).

The main advantage of SAEM is that it converges with a constant, and usually small, value of sample size, contrary to MCEM that requires a new value of n_m in each iteration. Thus, the only decision that we have to make in order to implement a SAEM algorithm is the choice of the step size γ_m. For a recent extension of the SAEM algorithm, referred to as PX–SAEM, in order to increase further its performance by a parameter expansion method see Lavielle and Meza (2007)

A main feature of all stochastic versions of the EM algorithm is that they do not converge pointwise. They generate a Markov chain whose stationary distribution is concentrated around the MLE. A reasonable estimator could be the ergodic mean after a burn–in period. Moreover, contrary to the EM algorithm the increase in log-likelihood is not guaranteed at each iteration due to the Monte Carlo error that is introduced in the simulation step. Nevertheless, under some regularity conditions [see for example Chan and Ledolter (1995)], the stochastic versions of the EM algorithm still converge to the MLE.

One of the most challenging topics in the implementation of the stochastic versions of the EM algorithm is to find a proper stopping rule. The stopping rule of the EM algorithm could not be applied in that case due to the stochastic nature of the algorithm, since small differences in the absolute difference of two successive values of the Q–function and/or the parameter estimates can come by chance. Booth and Hobert (1999) proposed to apply the deterministic stopping criterion of the EM algorithm for a predefined number of times in order to reduce the probability of a premature stop of the algorithm. For more sophisticated stopping rules see Gu and Zhu (2001), Caffo et al. (2005) and Jank (2006b). A detailed review on the stochastic versions of the EM algorithm is presented by Jank (2005, 2006a).

In the rest of this section we present a stochastic version of the EM algorithm for finding the MLE in nonparametric HSMMs.

As we mentioned above, using any stochastic version of the EM algorithm, at each iteration m, firstly we have to simulate $\mathbf{Z}_0^M$ from the conditional distribution $\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(\cdot \mid \mathbf{y})$ and then we maximize the log–likelihood of the complete data with respect to $\boldsymbol{\theta}$.

The simulation starts by sampling the last visited state Z_M and the last jump time $S_{N(M)}$. So, for any $i \in E$ and $0 \leq u \leq \widetilde{n}_i - 1$ we have

$$\mathbb{P}_{\boldsymbol{\theta}}(Z_M = i, S_{N(M)} = M - u \mid \mathbf{y}) = \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{M-u}^M = \boldsymbol{i}, Z_{M-u-1} \neq i \mid \mathbf{y}) = L_{1;M}(i,u),$$

where $L_{1;M}(i,u)$ is given by (11).

For the intermediate steps, the recursive relation that gives us the previous visited state and the corresponding jump time (J_{l-1}, S_{l-1}) conditioned on the sequence $\left(\mathbf{J}_l^{N(M)}, \mathbf{S}_l^{N(M)}, \mathbf{Y}_0^M\right)$ can be obtained by noting that

$$\begin{aligned}
&\mathbb{P}_{\boldsymbol{\theta}}\left(J_{l-1} = i, S_l - S_{l-1} = u \mid \mathbf{J}_{l+1}^{N(M)}, \mathbf{S}_{l+1}^{N(M)}, S_l = n, J_l = j, \mathbf{y}\right) \\
&= \mathbb{P}_{\boldsymbol{\theta}}\left(\mathbf{Z}_{n-u}^{n-1} = \boldsymbol{i}, Z_{n-u-1} \neq i \mid S_l = n, Z_n = j, \mathbf{Z}_{n+1}^M, \mathbf{y}\right) \\
&= \frac{\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n-u}^{n-1} = \boldsymbol{i}, Z_{n-u-1} \neq i, S_l = n, Z_n = j, \mathbf{Z}_{n+1}^M, \mathbf{y})}{\mathbb{P}_{\boldsymbol{\theta}}\left(S_l = n, Z_n = j, \mathbf{Z}_{n+1}^M, \mathbf{y}\right)}. \qquad (33)
\end{aligned}$$

The numerator N_1 of equation (33) can be written as:

$$\begin{aligned} N_1 &= \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_0^{n-u-1} = \mathbf{y}_0^{n-u-1})\mathbb{P}_{\boldsymbol{\theta}}(Z_{n-u} = i, Z_{n-u-1} \neq i, y_{n-u} \mid \mathbf{y}_0^{n-u-1}) \\ &\quad \times \mathbb{P}_{\boldsymbol{\theta}}(Z_n = j, \mathbf{Z}_{n-u+1}^{n-1} = \boldsymbol{i} \mid Z_{n-u} = i, Z_{n-u-1} \neq i) \\ &\quad \times \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_{n-u+1}^{n} = \mathbf{y}_{n-u+1}^{n} \mid Z_n = j, \mathbf{Z}_{n-u+1}^{n-1} = \boldsymbol{i}) \\ &\quad \times \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n+1}^{M}, \mathbf{y}_{n+1}^{M} \mid Z_n = j, Z_{n-1} \neq j). \end{aligned} \tag{34}$$

The denominator N_2 of equation (33) can be written as:

$$N_2 = \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Y}_0^{n-1} = \mathbf{y}_0^{n-1})\ \mathbb{P}_{\boldsymbol{\theta}}(Z_n = j, Z_{n-1} \neq j, y_n \mid \mathbf{y}_0^{n-1})\ \mathbb{P}_{\boldsymbol{\theta}}(\mathbf{Z}_{n+1}^{M}, \mathbf{y}_{n+1}^{M} \mid Z_n = j, Z_{n-1} \neq j). \tag{35}$$

Finally, by equations (33)–(35) and the quantities defined already in Section 3 we get

$$\mathbb{P}_{\boldsymbol{\theta}}(J_{l-1} = i, S_l - S_{l-1} = u \mid \mathbf{J}_{l+1}^{N(M)}, \mathbf{S}_{l+1}^{N(M)}, S_l = n, J_l = j, \mathbf{y}) = \frac{q_{ij}(u) B_{n-u}(i) R_{j;y_n} T_{n-1,u-1}(i)}{B_n(j) \prod_{p=n-u}^{n-1} P_p}. \tag{36}$$

Remark: In the special case where the state space of $\mathbf{Z}$ has only 2 states, after we sample the last state Z_M we have a deterministic change in the states of the embedded MC $\mathbf{J}$ and all we have to sample are the sojourn times in each state.

A similar approach in order to sample from $\mathbb{P}_{\boldsymbol{\theta}^{(m)}}(\cdot \mid \mathbf{y})$ is proposed by Guédon (2007), but in his algorithm the author suggests to draw sequentially the current state and then conditionally on that and on the observations, the state occupancy is drawn.

The main advantage of the stochastic version of the EM algorithm is that in order to implement it we need only the forward procedure, that means significantly less CPU time per iteration. This gives a computational advantage, especially in real data problems with huge trajectories.

6 Simulation studies

In order to illustrate the performance of the proposed algorithms, they were applied on several examples. In this section we present some selected results. The algorithms were implemented in `Fortran 95` on a PC equipped with an Intel Core 2 Duo E6400 processor at 2.3 GHz and 1 GB of memory. All the variables were of double precision and the operating system was `Ubuntu Linux`. Figures were created using `Mathematica`.

Since the implementation of the SAEM seems to be the most straightforward, we used this variation in our examples.

The stopping criterion that is used is based on the absolute difference of two successive values of the log–likelihood function as it is used by Bulla et al. (2008) and for the SAEM algorithm we applied the same criterion three successive times [Booth and Hobert (1999)]. Moreover, in that case, we used the ergodic mean for the parameter estimation after a burn–in period. More specifically, we discard the first 75% of the estimated values and the rest simulated sequence is averaged.

Case 1: Let us consider the hidden SMC $(\mathbf{Z}, \mathbf{Y})$, where the state space of both $\mathbf{Z}$ and $\mathbf{Y}$ is the set $\{0, 1\}$. We denote by $\mathcal{W}(q, b)$ the discrete–time Weibull distribution, i.e., $X \sim \mathcal{W}(q, b)$ if

$$\mathbb{P}(X = n) = q^{(n-1)^b} - q^{n^b},\ n \in \mathbb{N}^*.$$

We use one truncated trajectory $\mathbf{Y}_0^M$, with $M = 50000$, that was generated by using the SM kernel that consists of $\boldsymbol{q}_{01}(\cdot) := \mathcal{W}(0.7, 0.9)$ and $\boldsymbol{q}_{10}(\cdot) := \mathcal{W}(0.5, 0.7)$ and the vectors of emission probabilities $\boldsymbol{R}_0 := (0.8, 0.2)$ and $\boldsymbol{R}_1 := (0.2, 0.8)$. Three different initial values $\boldsymbol{\theta}^{(0)} := \left(\boldsymbol{q}_{01}^{(0)}, \boldsymbol{q}_{10}^{(0)}, \boldsymbol{R}_0^{(0)}, \boldsymbol{R}_1^{(0)}\right)$ have been used for both EM and SAEM algorithms:

$$\begin{aligned}
(\mathbf{1}_\alpha):\quad & \boldsymbol{q}_{01}^{(0)} = \left(0.3, 0.2, 0.1, \tfrac{0.4}{\tilde{n}_{01}-3}, \ldots, \tfrac{0.4}{\tilde{n}_{01}-3}\right),\ \boldsymbol{q}_{10}^{(0)} = \left(0.5, 0.2, 0.1, \tfrac{0.2}{\tilde{n}_{10}-3}, \ldots, \tfrac{0.2}{\tilde{n}_{10}-3}\right), \\
& \boldsymbol{R}_0^{(0)} = (0.8, 0.2),\ \boldsymbol{R}_1^{(0)} = (0.2, 0.8), \\
(\mathbf{1}_\beta):\quad & \boldsymbol{q}_{01}^{(0)} = \tfrac{1}{\tilde{n}_{01}} \mathbf{1}_{\tilde{n}_{01}},\ \boldsymbol{q}_{10}^{(0)} = \tfrac{1}{\tilde{n}_{10}} \mathbf{1}_{\tilde{n}_{10}},\ \boldsymbol{R}_0^{(0)} = (0.8, 0.2),\ \boldsymbol{R}_1^{(0)} = (0.2, 0.8), \\
(\mathbf{1}_\gamma):\quad & \boldsymbol{q}_{01}^{(0)} = \tfrac{1}{\tilde{n}_{01}} \mathbf{1}_{\tilde{n}_{01}},\ \boldsymbol{q}_{10}^{(0)} = \tfrac{1}{\tilde{n}_{10}} \mathbf{1}_{\tilde{n}_{10}},\ \boldsymbol{R}_0^{(0)} = (0.6, 0.4),\ \boldsymbol{R}_1^{(0)} = (0.4, 0.6),
\end{aligned}$$

where $\mathbf{1}_s,\ s = \tilde{n}_{01},\ \tilde{n}_{10}$, represents the s-dimensional vector of ones and as support $(\tilde{n}_{01},\ \tilde{n}_{10})$ we used $(15, 10)$ and $(20, 15)$.

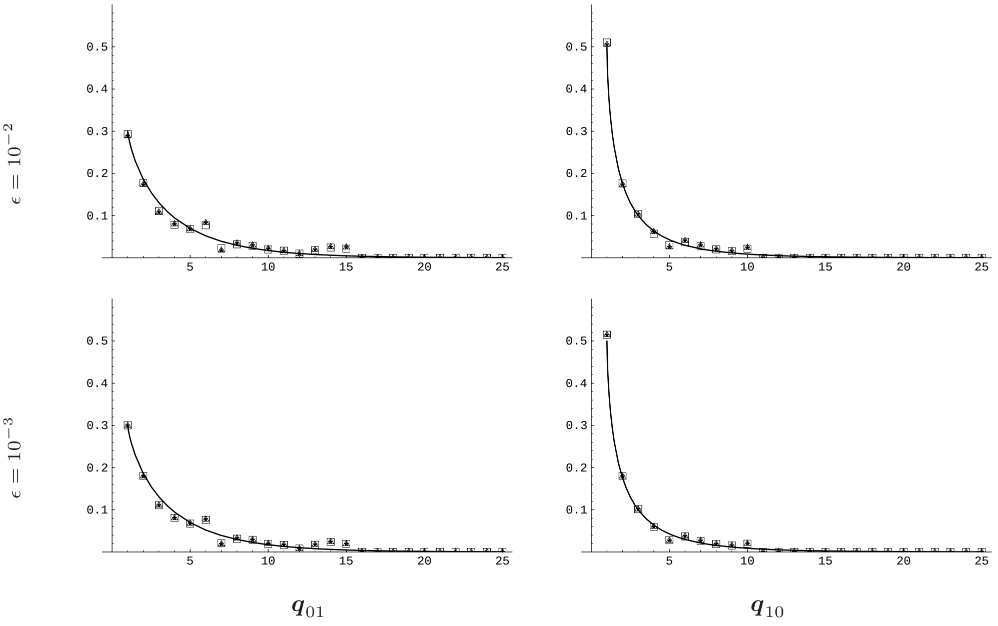

Figure 1: Left: The true values (solid line) and the estimated values using EM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using EM for $q_{10}(k)$ are presented. The estimates, that are obtained by using $(\mathbf{1}_\alpha)$ as initial values, are depicted by boxes, using $(\mathbf{1}_\beta)$ by triangles and using $(\mathbf{1}_\gamma)$ by stars ($M = 50000$, support ($\tilde{n}_{01} = 15,\ \tilde{n}_{10} = 10$)).

As we can see in Figure 1, with chosen support ($\tilde{n}_{01} = 15,\ \tilde{n}_{10} = 10$), the estimates are really close to the true values of almost all model parameters for the two accuracy levels $\epsilon = 10^{-2}$ and $\epsilon = 10^{-3}$. Moreover, we implemented the above example by using a different support ($\tilde{n}_{01} = 20,\ \tilde{n}_{10} = 15$). In that case we have very good estimates for all the aforementioned accuracy levels (see Figures 2). In Figures 3, we can see a comparison of the estimates that are obtained by using different supports. It is clear that by using a larger support we have better estimates in the tail of the distribution. On the other hand, if the support is further increased, there is a risk that the observed trajectory will not have a sufficient length in order to obtain good estimates. Moreover, we mention that, by using as stopping criterion $\epsilon = 10^{-3}$ we have slightly better estimates in the tail of the distribution than the ones obtained by using as stopping criterion $\epsilon = 10^{-2}$, but on the other hand we need much more EM steps in order the stopping criterion to be satisfied (see Table 1). Thus, in order to balance between the accuracy and the computational cost, we have to choose a proper

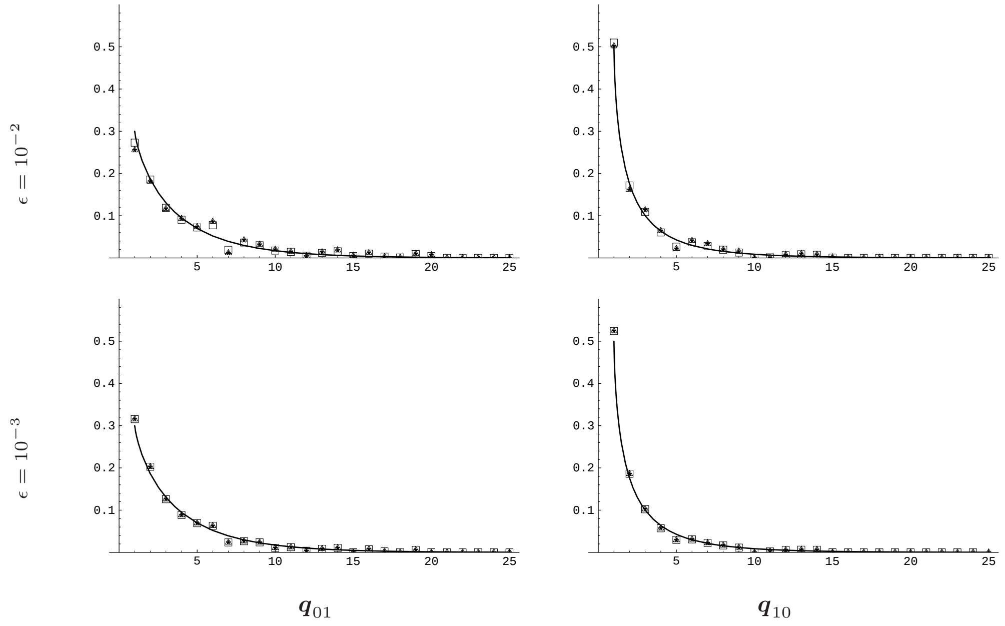

Figure 2: Left: The true values (solid line) and the estimated values using EM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using EM for $q_{10}(k)$ are presented. The estimates, that are obtained by using $(\mathbf{1}_\alpha)$ as initial values are depicted by boxes, using $(\mathbf{1}_\beta)$ by triangles and using $(\mathbf{1}_\gamma)$ by stars ($M = 50000$, support ($\tilde{n}_{01} = 20$, $\tilde{n}_{10} = 15$)).

	(I)			(II)		
	$(\mathbf{1}_\alpha)$	$(\mathbf{1}_\beta)$	$(\mathbf{1}_\gamma)$	$(\mathbf{1}_\alpha)$	$(\mathbf{1}_\beta)$	$(\mathbf{1}_\gamma)$
$\epsilon = 10^{-2}$	107	210	224	217	439	446
$\epsilon = 10^{-3}$	200	371	387	720	1071	1076

Table 1: Numbers of EM steps in order the stopping criterion to be satisfied [$M = 50000$, (I) for support ($\tilde{n}_{01} = 15$, $\tilde{n}_{10} = 10$) and (II) for ($\tilde{n}_{01} = 20$, $\tilde{n}_{10} = 15$)].

ϵ and $\tilde{n}_{ij}$'s.. In the rest of this section, we use as accuracy level $\epsilon = 10^{-2}$ because by using $\epsilon = 10^{-3}$ we do not have any great improvement in the estimates and the computational cost is extremely large. Another key point is to choose proper $\tilde{n}_{ij}$'s, in such a way that they are large enough in order to cover the most significant part of the original support and on the other hand, small enough compared to the length of the observed sequence. For the rest of this section we use as support $\tilde{n}_{01} = 15$ and $\tilde{n}_{10} = 10$.

We also implemented SAEM for the above data sets. We run 10 times SAEM algorithm for each initial value. The estimates are similar in each run and very close to the true values as Figure 4 depicts. It is important to mention here that, an iteration of SAEM needs not only less CPU time than an iteration of EM algorithm (18.87 and 31.65 seconds in average per iteration correspondingly), but also converges faster in almost all cases as we can see in Table 2.

Thus, we conclude that SAEM converges faster than EM. Moreover, we can further increase the convergence rate of SAEM by a proper choice of the step size γ_m.

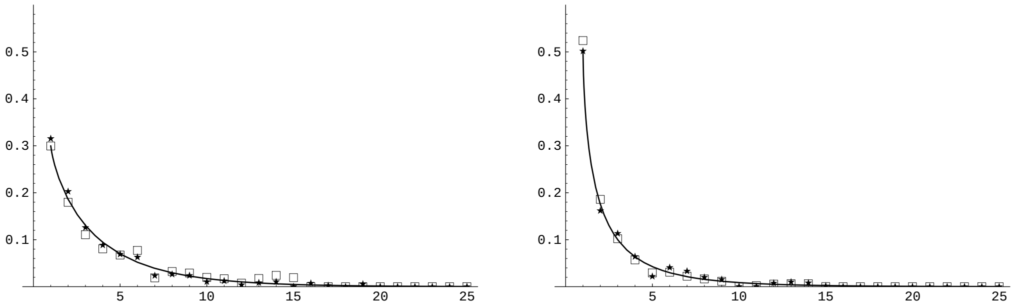

Figure 3: Left: The true values (solid line) and the estimated values using EM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using EM for $q_{10}(k)$ are presented. The estimates are obtained by using $(\mathbf{1}_\gamma)$ as initial values. The boxes depict the estimates that are obtained by using as support ($\tilde{n}_{01} = 15$, $\tilde{n}_{10} = 10$) and the triangles depict the estimates that are obtained by using as support ($\tilde{n}_{01} = 20$, $\tilde{n}_{10} = 15$) ($\epsilon = 10^{-3}$, $M = 50000$).

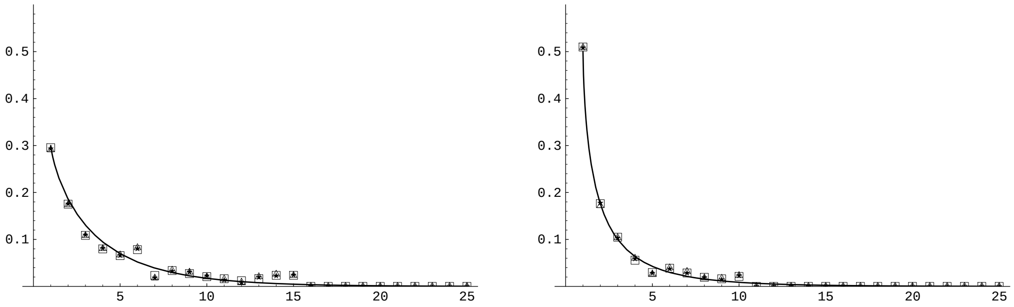

Figure 4: Left: The true values (solid line) and the estimated values using SAEM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using SAEM for $q_{10}(k)$ are presented. The estimates, that are obtained by using $(\mathbf{1}_\alpha)$ as initial values, are depicted by boxes, using $(\mathbf{1}_\beta)$ by triangles and using $(\mathbf{1}_\gamma)$ by stars ($\epsilon = 10^{-2}$, $M = 50000$).

	min	max	mean
$(\mathbf{1}_\delta)$	89	160	119.6
$(\mathbf{1}_\epsilon)$	150	347	242.0
$(\mathbf{1}_\zeta)$	178	299	222.6

Table 2: Numbers of SAEM steps in order the stopping criterion to be satisfied ($\epsilon = 10^{-2}, M = 50000$), support ($\tilde{n}_{01} = 15$, $\tilde{n}_{10} = 10$).

Case 2: Let us consider the HSMM $(\mathbf{Z}, \mathbf{Y})$, where $\mathbf{Z}$ has state space $\{0, 1\}$ and $\mathbf{Y}$ has $\{0, 1, 2, 3\}$. Moreover, let us assume that we have an observed sequence $\mathbf{Y}_0^M$ with $M = 70000$ that has been simulated with a SM kernel given by $\boldsymbol{q}_{01}(\cdot) := \mathcal{W}(0.7, 0.9)$ and $\boldsymbol{q}_{10}(\cdot) := \mathcal{W}(0.5, 0.7)$ (discrete–time random variables) and emission probabilities given by $\boldsymbol{R}_0 := (0.4, 0.3, 0.2, 0.1)$ and $\boldsymbol{R}_1 := (0.1, 0.2, 0.3, 0.4)$. We use as initial values for EM and SAEM the following vectors:

$$\begin{aligned}
(\mathbf{2}_\alpha):\ & \boldsymbol{q}_{01}^{(0)} = \left(0.3, 0.2, 0.1, \tfrac{0.4}{12}, \ldots, \tfrac{0.4}{12}\right),\ \boldsymbol{q}_{10}^{(0)} = \left(0.5, 0.2, 0.1, \tfrac{0.2}{7}, \ldots, \tfrac{0.2}{7}\right), \\
& \boldsymbol{R}_0^{(0)} = (0.4, 0.3, 0.2, 0.1),\ \boldsymbol{R}_1^{(0)} = (0.1, 0.2, 0.3, 0.4), \\
(\mathbf{2}_\beta):\ & \boldsymbol{q}_{01}^{(0)} = \tfrac{1}{15}\mathbf{1}_{15},\ \boldsymbol{q}_{10}^{(0)} = \tfrac{1}{10}\mathbf{1}_{10}, \\
& \boldsymbol{R}_0^{(0)} = (0.4, 0.3, 0.2, 0.1),\ \boldsymbol{R}_1^{(0)} = (0.1, 0.2, 0.3, 0.4), \\
(\mathbf{2}_\gamma):\ & \boldsymbol{q}_{01}^{(0)} = \tfrac{1}{15}\mathbf{1}_{15},\ \boldsymbol{q}_{10}^{(0)} = \tfrac{1}{10}\mathbf{1}_{10}, \\
& \boldsymbol{R}_0^{(0)} = (0.35, 0.3, 0.25, 0.1), \boldsymbol{R}_1^{(0)} = (0.1, 0.25, 0.3, 0.35).
\end{aligned}$$

In Figures 5–6 the estimated values that where obtained from EM and SAEM algorithms respectively, are depicted. Moreover, in Tables 3–4, we can see the corresponding number of iterations that were needed in each case.

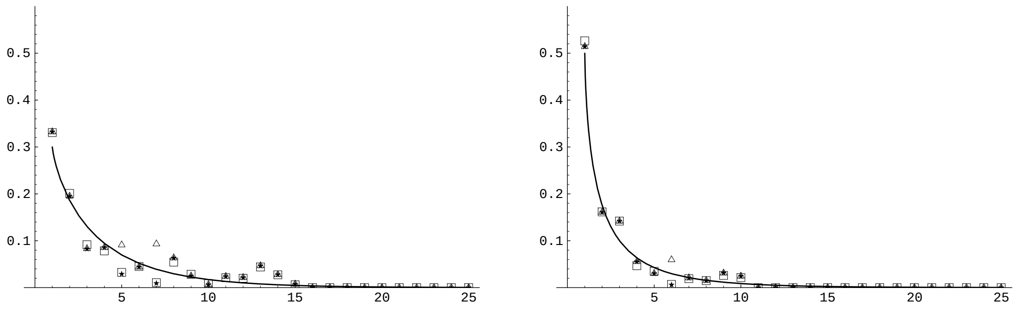

Figure 5: Left: The true values (solid line) and the estimated values using EM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using EM for $q_{10}(k)$ are presented. The estimates, that are obtained by using $(\mathbf{2}_\alpha)$ as initial values, are depicted by boxes, using $(\mathbf{2}_\beta)$ by triangles and using $(\mathbf{2}_\gamma)$ by stars ($\epsilon = 10^{-2}$, $M = 70000$).

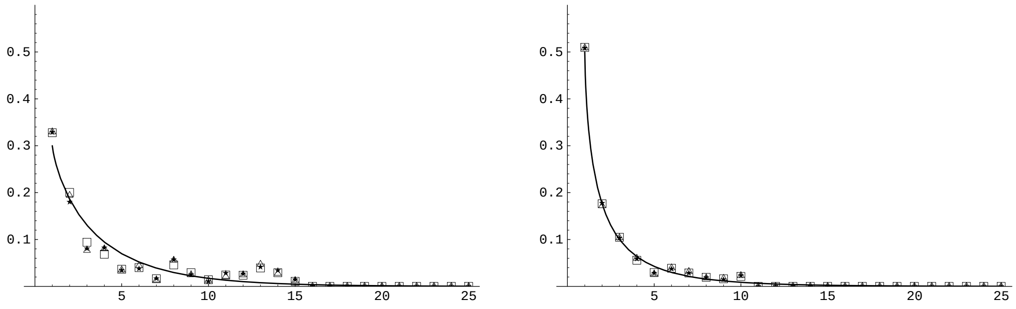

Figure 6: Left: The true values (solid line) and the estimated values using SAEM for $q_{01}(k)$ are presented. Right: The true values (solid line) and the estimated values using SAEM for $q_{10}(k)$ are presented. The estimates, that are obtained by using $(\mathbf{2}_\alpha)$ as initial values, are depicted by boxes, using $(\mathbf{2}_\beta)$ by triangles and using $(\mathbf{2}_\gamma)$ by stars ($\epsilon = 10^{-2}$, $M = 70000$).

	$(\mathbf{2}_\alpha)$	$(\mathbf{2}_\beta)$	$(\mathbf{2}_\gamma)$
$\epsilon = 10^{-2}$	304	525	540
$\epsilon = 10^{-3}$	705	953	969

Table 3: Numbers of EM steps in order the stopping criterion to be satisfied ($M = 70000$), support ($\tilde{n}_{01} = 15$, $\tilde{n}_{10} = 10$).

	min	max	mean
$(\mathbf{2}_\alpha)$	57	271	175.7
$(\mathbf{2}_\beta)$	243	482	359.1
$(\mathbf{2}_\gamma)$	261	427	356.8

Table 4: Numbers of SAEM steps in order the stopping criterion to be satisfied ($\epsilon = 10^{-2}, M = 70000$), support ($\tilde{n}_{01} = 15$, $\tilde{n}_{10} = 10$).

7 Conclusion

In this paper we proposed an EM and stochastic versions of EM algorithm for finding the MLE for nonparametric HSMMs. Our method is directly applicable to the cases that the support of the improper distributions of the semi–Markov kernel is finite or when the error of approximation by an appropriate finite support is small. This is not the case for heavy tailed distributions. Our experimental results are encouraging and promising for further investigation. We have taken good estimates in feasible time by using EM and SAEM algorithms. The proposed algorithms can find straightforward applications in biostatistics, operation research, reliability, queuing theory and several other fields. Concluding, we would like to mention some interesting problems that have arisen during this study. A main problem is to estimate appropriate barriers for the support of the improper distributions when they are unbounded. Moreover, the finding of proper weights γ_m for the SAEM algorithm in order to achieve faster convergence and also proper stopping rules are issues of great importance.

References

Barbu, V., Limnios, N., 2006. Maximum likelihood estimation for hidden semi-Markov models. C.R. Acad. Sci. Paris 342, 201–205.

Barbu, V., Limnios, N., 2008. Semi-Markov Chains and Hidden Semi-Markov Models toward Applications. Springer.

Baum, L., Petrie, T., Soules, G., Weiss, N., 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. The Annals of Mathematical Statistics 41 (1), 164–171.

Baum, L. E., Petrie, T., 1966. Statistical inference for probabilistic functions of finite state Markov chains. The Annals of Mathematical Statistics 37, 1554–1563.

Bhar, R., Hamori, S., 2004. Hidden Markov Models: Applications to Financial Economics. MA: Kluwer Academic Publishers, Boston.

Booth, J., Hobert, J., 1999. Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. Journal of the Royal Statistical Society, Series B 61, 265–285.

Booth, J., Hobert, J., Jank, W., 2001. A survey of Monte Carlo algorithms for maximizing the likelihood of a two-stage hierarchical model. Statistical Modelling 1, 333–349.

Boyles, R., 1983. On the convergence of EM algorithm. Journal of the Royal Statistical Society, Series B 45, 47–50.

Bulla, J., 2006. Application of hidden Markov models and hidden semi-Markov models to financial time series. Thesis, Georg-August-Universitat Gottingen.

Bulla, J., Bulla, I., 2006. Stylized facts of financial time series and hidden semi-Markov models. Computational Statistics & Data Analysis 51 (4), 2192–2209.

Bulla, J., Bulla, I., Nenadić, O., 2008. hsmm–An R package for analyzing hidden semi-Markov models. Computational Statistics & Data Analysis.

Caffo, B., Jank, W., Jones, G. ., 2005. Ascent-based Monte Carlo EM. Journal of the Royal Statistical Society, Series B 67, 235–252.

Cappé, O., Moulines, E., Rydén, T., 2005. Inference in Hidden Markov Models. Springer, New York.

Celeux, G., Diebolt, J., 1985. The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. Computational Statistics Quarterly 2, 73–82.

Chan, K., Ledolter, J., 1995. Monte Carlo EM Estimation of the Time Series Models involving counts. Journal of American Statistical Association 90, 242–252.

Delyon, B., Lavielle, V., Moulines, E., 1999. Convergence of a stochastic approximation version of the EM algorithm. Annals of Statistics 27, 94–128.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum Likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B 39, 1–38.

Durbin, R., Eddy, S., Krogh, A., Mitchison, G., 1998. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University press, Cambridge.

Ephraim, Y., Merhav, N., 2002. Hidden Markov Processes. IEEE Trans. Inf. Theory 48 (6), 1518–1569.

Ferguson, J., 1980. Variable duration models for speech. In: Proc. of the symposium on the Application of Hidden Markov Models to Text and Speech. Princeton, New Jersey, pp. 143–179.

Gu, M., Zhu, H., 2001. Maximum likelihood estimation for spatial models by Markov chain Monte Carlo stochastic approximation. Journal of the Royal Statistical Society, Series B 63, 339–355.

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

Guédon, Y., 2005. Hidden hybrid Markov/semi-Markov chains. Computational Statistics & Data Analysis 49 (3), 663–688.

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., Cocozza-Thivent, C., 1990. Explicit state occupancy modelling by hidden semi-Markov models: application of Derin's scheme. Computer Speech and Language 4, 167–192.

Jank, W., 2005. Stochastic Variants of EM: Monte Carlo, Quasi–Monte Carlo and More. In: Proc. of the American Statistical Association.

Jank, W., 2006a. The EM algorithm, Its Stochastic Implementation and Global Optimization: Some Challenges and Opportunities for OR. In: Alt, F., Fu, M., Golden, B. (Eds.), Topics in Modeling, Optimization and Decision Technologies: Honoring Saul Gass' Contributions to Operation Research. Springer-Verlag, pp. 367–392.

Jank, W., 2006b. Implementing and Diagnosing the Stochastic Approximation EM algorithm. Journal of Computational and Graphical Statistics 15 (4), 803–829.

Krogh, A., Brown, M., Mian, I. S., Sjflander, K., Haussler, D., 1994a. Hidden Markov models in computational biology: Applications to protein modeling. Journal of Molecular Biology 235, 1501–1531.

Krogh, A., Mian, I., Haussler, D., 1994b. A Hidden Markov model that finds genes in E. coli DNA. Nucleic Acids Research 22, 4768–4778.

Lavielle, M., Meza, C., 2007. A parameter expansion version of the SAEM algorithm. Statistics and Computing 17 (2), 121–130.

Levine, R., Casella, G., 2001. Implementations of the Monte Carlo EM algorithm. Journal of Computational and Graphical Statistics 10, 422–439.

Levine, R., Fan, J., 2004. An automated (Markov chain) Monte Carlo EM algorithm. Journal of Statistical Computation and Simulation 74, 349–359.

Levinson, S., 1986. Continuously variable duration hidden Markov models for automatic speech recognition. Computer Speech and Language 1, 29–45.

Li, J., Gray, R., 2000. Image Segmentation and Compression using Hidden Markov models. Springer, New York.

McLachlan, G. J., Krishnan, T., 2008. The EM Algorithm and Extensions. John Wiley & Sons Inc.

Polyak, B., Juditsky, A., 1992. Acceleration of Stochastic Approximation by Averaging. SIAM Journal on Control and Optimization 30, 838.

Pyke, R., 1961. Markov renewal processes: definitions and preliminary properties. The Annals of Mathematical Statistics 32, 1231–1242.

Rabiner, L. R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE 77, 257–284.

Rabiner, L. R., Juang, B. H., 1993. Fundamentals of Speech Recognition. Prentice Hall, Englewood Cliffs, NJ.

Sansom, J., 1998. A Hidden Markov model for Rainfall using breakpoint data. Journal of Climate 11 (1), 42–53.

Sansom, J., Thomson, P., 2001. Fitting hidden semi-Markov models to breakpoint rainfall data. Journal of Applied Probability 38A, 142–157.

Trevezas, S., Limnios, N., 2009. Maximum likelihood estimation for general hidden semi-Markov processes with backward recurrence time dependence. POMI 363, 105–125, *Special issue in honor of I. Ibragimov*.

Wei, G., Tanner, M., 1990. A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. Journal of the American Statistical Association 85, 699–704.

Wu, C., 1983. On the convergence properties of EM algorithm. The Annals of Statistics 11, 95–103.