16th IFAC Symposium on System Identification
The International Federation of Automatic Control
Brussels, Belgium. July 11-13, 2012

# An Online Expectation-Maximisation Algorithm for Nonnegative Matrix Factorisation Models

Sinan Yıldırım [*], A. Taylan Cemgil [**], Sumeetpal S. Singh [***]

[*] *Statistical Laboratory, DPMMS, University of Cambridge, UK*
[**] *Department of Computer Engineering, Boğaziçi University, Turkey*
[***] *Department of Engineering, University of Cambridge, UK*

**Abstract:** In this paper we formulate the nonnegative matrix factorisation (NMF) problem as a maximum likelihood estimation problem for hidden Markov models and propose online expectation-maximisation (EM) algorithms to estimate the NMF and the other unknown static parameters. We also propose a sequential Monte Carlo approximation of our online EM algorithm. We show the performance of the proposed method with two numerical examples.

## 1. INTRODUCTION

With the advancement of sensor and storage technologies, and with the cost of data acquisition dropping significantly, we are able to collect and record vast amounts of raw data. Arguably, the grand challenge facing computation in the 21st century is the effective handling of such large data sets to extract meaningful information for scientific, financial, political or technological purposes [Donoho, 2000]. Unfortunately, classical batch processing methods are unable to deal with very large data sets due to memory restrictions and slow computational time.

One key approach for the analysis of large datasets is based on the matrix and tensor factorisation paradigm. Given an observed dataset $Y$, where $Y$ is a matrix of a certain dimension and each element of it corresponds to an observed data point, the matrix factorisation problem is the computation of matrix factors $B$ and $X$ such that $Y$ is approximated by the matrix product $BX$, i.e.,

$$Y \approx BX$$

(Later we will make our notation and inferential goals more precise.) Indeed, many standard statistical methods such as clustering, independent components analysis (ICA), nonnegative matrix factorisation (NMF), latent semantic indexing (LSI), collaborative filtering can be expressed and understood as matrix factorisation problems [Lee and Seung, 1999, Singh and Gordon, 2008, Koren et al., 2009].

Matrix factorisation models also have well understood probabilistic/statistical interpretations as probabilistic generative models and many standard algorithms mentioned above can also be derived as maximum likelihood or maximum a-posteriori parameter estimation procedures [Fevotte and Cemgil, 2009, Salakhutdinov and Mnih, 2008, Cemgil, 2009]. The advantage of this interpretation is that it enables one to incorporate domain specific prior knowledge in a principled and consistent way. This can be achieved by building hierarchical statistical models to fit the specifics of the application at hand. Moreover, the probabilistic/statistical approach also provides a natural framework for sequential processing which is desirable for developing online algorithms that pass over each data point only once. While the development of effective online algorithms for matrix factorisation are of interest on their own, the algorithmic ideas can be generalised to more structured models such as tensor factorisations (e.g. see [Kolda and Bader, 2009]).

In this paper our primary interest is estimation of $B$ (rather than $B$ and $X$), which often is the main objective in NMF problems. We formulate the NMF problem as a maximum likelihood estimation (MLE) problem for hidden Markov models (HMMs). The advantage of doing so is that the asymptotic properties of MLE for HMM's has been studied in the past by many authors and these results may be adapted to the NMF framework. We propose a sequential Monte Carlo (SMC) based online EM algorithm [Cappé, 2009, Del Moral et al., 2009] for the NMF problem. SMC introduces a layer of bias which decreases as the number of particles in the SMC approximation is increased.

In the literature, several online algorithms have been proposed for online computation of matrix factorisations. Mairal et al. [2010] propose an online optimisation algorithm, based on stochastic approximations, which scales up gracefully to large data sets with millions of training samples. A proof of convergence is presented for the Gaussian case. There are similar formulations applied to other matrix factorisation formulations, notably NMF [Lefevre et al., 2011] and Latent Dirichlet Allocation [Hoffman et al., 2010], as well as alternative views for NMF which are based on incremental subspace learning [Bucak and Gunsel, 2009]. Although the empirical results of these methods suggest good performance, their asymptotic properties have not been established.

### *1.1 Notation*

Let $A$ be a $M \times N$ matrix. The $(m, n)$'th element of $A$ is $A(m, n)$. If $M$ (or $N$) is 1, then $A(i) = A(1, i)$ (or $A(i, 1)$). The $m$'th row of $A$ is $A(m, \cdot)$. If $A$ and $B$ are both $M \times N$ matrices, $C = A \odot B$ denotes element-by-element multiplication, i.e., $C(m, n) = A(m, n)B(m, n)$; $\frac{A}{B}$ (or $A/B$) means element-by-element division, in a similar

978-3-902823-06-9/12/$20.00 © 2012 IFAC

10.3182/20120711-3-BE-2027.00312

way. $\mathbf{1}_{M\times N}$ ($\mathbf{0}_{M\times N}$) is a $M \times N$ matrix of 1's (0's), where $\mathbf{1}_{M\times 1}$ is abbreviated to $\mathbf{1}_M$. $\mathbb{N} = \{0, 1, 2, \ldots\}$ and $\mathbb{R}_+ = [0, \infty)$ are the sets of nonnegative integers and real numbers. Random variables will be defined by using capital letters, such as $X, Y, Z$, etc., and their realisations will be corresponding small case letters ($x, y, z$, etc.). The indicator function $I_\alpha(x) = 1$ if $x = \alpha$, otherwise it is 0; also, for a set $A$, $I_A(x) = 1$ if $x \in A$, otherwise it is 0.

## 2. THE STATISTICAL MODEL FOR NMF

Consider the following HMM comprised of the latent processes $\{X_t, Z_t\}_{t\geq 1}$ and the observation process $\{Y_t\}_{t\geq 1}$. The process $\{X_t \in \mathbb{R}_+^K\}_{t\geq 1}$ is a Markov process of $K \times 1$ nonnegative vectors with an initial density $\mu_\psi$ and the transition density $f_\psi$ for $t = 2, 3, \ldots$

$$X_1 \sim \mu_\psi(x),\ X_t|\,(X_{t-1} = x_{t-1}) \sim f_\psi(x_t|x_{t-1}), \quad (1)$$

where $\psi \in \Psi$ is a finite dimensional parameter which parametrizes the law of the Markov process. $Z_t \in \mathbb{N}^{M\times K}$ is a $M \times K$ matrix of nonnegative integers, and its elements are independent conditioned on $X_t$ as follows:

$$Z_t|\,(X_t = x_t) \sim \prod_{m=1}^{M}\prod_{k=1}^{K} \mathcal{PO}(z_t(m,k); B(m,k)x_t(k))$$

where $B \in \mathbb{R}_+^{M\times K}$ is an $M \times K$ nonnegative matrix. Here $\mathcal{PO}(v; \lambda)$ denotes the Poisson distribution on $\mathbb{N}$ with intensity parameter $\lambda \geq 0$

$$\mathcal{PO}(v; \lambda) = \exp\left(v \log \lambda - \lambda - \log v!\right),$$

The $M \times 1$ observation vector $Y_t$ is conditioned on $Z_t$ in a deterministic way

$$Y_t(m) = \sum_{k=1}^{K} Z_t(m,k), \quad m = 1, \ldots, M.$$

This results in the conditional density of $Y_t$ given $X_t = x_t$, denoted by $g_B$, being a multivariate Poisson density

$$Y_t|\,(X_t = x_t) \sim g_B(y_t|x_t) = \prod_{m=1}^{M} \mathcal{PO}\left(y_t(m); B(m,\cdot)x_t\right)(2)$$

Hence the likelihood of $y_t$ given $x_t$ can analytically be evaluated. Moreover, the conditional posterior distribution $\pi_B(z_t|y_t, x_t)$ of $Z_t$ given $y_t$ and $x_t$ has a factorized closed form expression:

$$\begin{aligned}Z_t|\,(Y_t = y_t, X_t = x_t) &\sim \pi_B(z_t|y_t, x_t)\\ &= \prod_{m=1}^{M} \mathcal{M}\left(z_t(m,\cdot); y_t(m), \rho_{t,m}\right)(3)\end{aligned}$$

where $\rho_{t,m}(k) = B(m,k)x_t(k)/B(m,\cdot)x_t$ and $\mathcal{M}$ denotes a multinomial distribution defined by

$$\mathcal{M}(v; \alpha, \rho) = I_\alpha\left(\sum_{k=1}^{K} v_k\right) \alpha! \prod_{k=1}^{K} \frac{\rho_k^{v_k}}{v_k!},$$

where $v = [v_1 \ldots v_K]$ is a realisation of the vector valued random variable $V = [V_1 \ldots V_K]$, $\rho = (\rho_1, \ldots, \rho_K)$, and $\sum_{k=1}^{K} \rho_k = 1$. It is a standard result that the marginal mean of the $k$'th component is $\mathbb{E}_{\alpha,\rho}[V_k] = \alpha\rho_k$.

Let $\theta = (\psi, B) \in \Theta = \Psi \times \mathbb{R}_+^{M\times K}$ denote all the parameters of the HMM. We can write the joint density of $(X_{1:t}, Z_{1:t}, Y_{1:t})$ given $\theta$ as

$$\begin{aligned}p_\theta(x_{1:t}, z_{1:t},\ y_{1:t}) &= \mu_\psi(x_1)g_B(y_1|x_1)\pi_B(z_1|y_1, x_1)\\ &\times \prod_{i=2}^{t} f_\psi(x_i|x_{i-1})g_B(y_i|x_i)\pi_B(z_i|x_i, y_i). \quad (4)\end{aligned}$$

From (4), we observe that the joint density of $(X_{1:t}, Y_{1:t})$

$$p_\theta(x_{1:t}, y_{1:t}) = \mu_\psi(x_1)g_B(y_1|x_1)\prod_{i=2}^{t} f_\psi(x_i|x_{i-1})g_B(y_i|x_i)$$

defines the law of another HMM $\{X_t, Y_t\}_{t\geq 1}$ comprised of the latent process $\{X_t\}_{t\geq 1}$, with initial and transitional densities $\mu_\psi$ and $f_\psi$, and the observation process $\{Y_t\}_{t\geq 1}$ with the observation density $g_B$. Finally, the likelihood of data is given by

$$p_\theta(y_{1:T}) = \mathbb{E}_\psi\left[\prod_{t=1}^{T} g_B(y_t|X_t)\right]. \quad (5)$$

In this paper, we treat $\theta$ as unknown and seek for the MLE solution $\theta^*$ for it, which satisfies

$$\theta^* = \arg\max_{\theta\in\Theta} p_\theta(y_{1:T}). \quad (6)$$

### *2.1 Relation to the classical NMF*

In the classical NMF formulation [Lee and Seung, 1999, 2000], given a $M \times T$ nonnegative matrix $Y = [y_1 \ldots y_T]$, we want to factorize it to $M \times K$ and $K \times T$ nonnegative matrices $B$ and $X = [X_1 \ldots X_T]$ such that the difference between $Y$ and $BX$ is minimised according to a divergence

$$(B^*, X^*) = \arg\min_{B,X} D(Y||BX). \quad (7)$$

One particular choice for $D$ is the generalised Kullback-Leibler (KL) divergence which is written as

$$D(Y||U) = \sum_{m=1}^{M}\sum_{t=1}^{T} Y(m,t)\log\frac{Y(m,t)}{U(m,t)} - Y(m,t) + U(m,t)$$

Noticing the similarity between the generalised KL divergence and the Poisson distribution, [Lee and Seung, 1999] showed that the minimisation problem can be formulated in a MLE sense. More explicitly, the solution to

$$(B^*, X^*) = \arg\max_{B,X} \ell(y_1, \ldots, y_T|B, X),$$

$$\ell(y_1, \ldots, y_T|B, X) = \prod_{t=1}^{T} g_B\left(y_t|X_t\right) \quad (8)$$

is the same as the solution to (7). In our formulation of the NMF problem, $X = [X_1 \ldots X_T]$ is not a static parameter but it is a random matrix whose columns constitute a Markov process. Therefore, the formulation for MLE in our case changes to maximising the expected value of the likelihood in (8) over the parameter $\theta = (B, \psi)$ with respect to (w.r.t.) the law of $X$

$$(B^*, \psi^*) = \arg\max_{(B,\psi)\in\Theta} \mathbb{E}_\psi\left[\ell(y_1, \ldots, y_T|B, X)\right]. \quad (9)$$

It is obvious that (6) and (9) are equivalent. We will see in Section 3 that the introduction of the additional process $\{Z_t\}_{t\geq 1}$ is necessary to perform MLE using the EM algorithm (see Lee and Seung [2000] for its first use for the problem stated in (7)).

## 3. EM ALGORITHMS FOR NMF

Our objective is to estimate the unknown $\theta$ given $Y_{1:T} = y_{1:T}$. The EM algorithm can be used to find the MLE for $\theta$. We first introduce the batch EM algorithm and then explain how an online EM version can be obtained.

### *3.1 Batch EM*

With the EM algorithm, given the observation sequence $y_{1:T}$ we increase the likelihood $p_\theta(y_{1:T})$ in (5) iteratively until we reach a maximal point on the surface of the likelihood. The algorithm is as follows:

Choose $\theta^{(0)}$ for initialisation. At iteration $j = 0, 1, \dots$

- **E-step:** Calculate the intermediate function which is the expectation of the log joint distribution of $(X_{1:T}, Z_{1:T}, Y_{1:T})$ with respect to the law of $(X_{1:T}, Z_{1:T})$ given $Y_{1:T} = y_{1:T}$.
  $Q(\theta^{(j)};\theta) = \mathbb{E}_{\theta^{(j)}}\left[\log p_\theta(X_{1:T}, Z_{1:T}, Y_{1:T}) \middle| Y_{1:T} = y_{1:T}\right]$
- **M-step:** The new estimate is the maximiser of the intermediate function
  $$\theta^{(j+1)} = \arg\max_\theta Q(\theta^{(j)};\theta)$$

With a slight modification of the update rules found in Cemgil [2009, Section 2], one can show that for NMF models the update rule for $B$ reduces to calculating the expectations

$$\widehat{S}_{1,T} = \mathbb{E}_{\theta^{(j)}}\left[\sum_{t=1}^{T} X_t \middle| Y_{1:T} = y_{1:T}\right],$$
$$\widehat{S}_{2,T} = \mathbb{E}_{\theta^{(j)}}\left[\sum_{t=1}^{T} Z_t \middle| Y_{1:T} = y_{1:T}\right]$$

and updating the parameter estimate for $B$ as

$$B^{(j+1)} = \widehat{S}_{2,T} / \left(\mathbf{1}_M \left[\widehat{S}_{1,T}\right]^T\right).$$

Moreover, if the transition density $f_\psi$ belongs to an exponential family, the update rule for $\psi$ becomes calculating the expectation of a $J \times 1$ vector valued function

$$\widehat{S}_{3,T} = \mathbb{E}_{\theta^{(j)}}\left[\sum_{t=1}^{T} s_{3,t}(X_{t-1}, X_t) \middle| Y_{1:T} = y_{1:T}\right]$$

and updating the estimate for $\psi$ using a maximisation rule

$$\Lambda : \mathbb{R}^J \to \Psi, \quad \psi^{(j+1)} = \Lambda\left(\widehat{S}_{3,T}\right).$$

Note that $s_{3,t}$ and $\Lambda$ depend on the NMF model, particularly to the probability laws in (1) defining the Markov chain for $\{X_t\}_{t\geq 1}$. Therefore, we have to find the mean estimates of the following sufficient statistics at time $t$.

$$S_{1,t}(x_{1:t}) = \sum_{i=1}^{t} x_i, \quad S_{2,t}(z_{1:t}) = \sum_{i=1}^{t} z_i,$$
$$S_{3,t}(x_{1:t}) = \sum_{i=1}^{t} s_{3,t}(x_{t-1}, x_t). \quad (10)$$

Writing the sufficient statistics in additive forms as in (10) enables us to use a forward recursion to find the expectations of the sufficient statistics in an online manner. This leads to an online version of the EM algorithm as we shall see in the following section.

### *3.2 Online EM*

To explain the methodology in a general sense, assume that we want to calculate the expectations $\widehat{S}_t = \mathbb{E}_\theta\left[S_t(X_{1:t}, Z_{1:t}) \middle| Y_{1:t} = y_{1:t}\right]$ of sufficient statistics of the additive form

$$S_t(x_{1:t}, z_{1:t}) = \sum_{i=1}^{t} s_i(x_{i-1}, z_{i-1}, x_i, z_i) \quad (11)$$

w.r.t. the posterior density $p_\theta(x_{1:t}, z_{1:t}|y_{1:t})$ for a given parameter value $B$. Letting $u_t = (x_t, z_t)$ for simplicity, we define the intermediate function

$$T_t(u_t) = \int S_t(u_{1:t}) p_\theta(u_{1:t-1}|y_{1:t-1}, u_t) du_{1:t-1}.$$

One can show that we have the forward recursion [Del Moral et al., 2009, Cappé, 2011]

$$T_t(u_t) = \int (T_{t-1}(u_{t-1}) + s_t(u_{t-1}, u_t)) \times p_\theta(u_{t-1}|y_{1:t-1}, u_t) du_{t-1} \quad (12)$$

with the convention $T_0(u) = 0$. Hence, $T_t$ can be computed online, so are the estimates

$$\widehat{S}_t = \int T_t(u_t) p_\theta(u_t|y_{1:t}) du_t.$$

We can decompose the backward transition density $p_\theta(u_{t-1}|y_{1:t-1}, u_t)$ and the filtering density $p_\theta(u_t|y_{1:t})$ as

$$p_\theta(x_{t-1}, z_{t-1}|y_{1:t-1}, x_t, z_t) = \pi_B(z_{t-1}|x_{t-1}, y_{t-1}) \times p_\theta(x_{t-1}|x_t, y_{1:t-1}), \quad (13)$$
$$p_\theta(x_t, z_t|y_{1:t}) = \pi_B(z_t|x_t, y_t) p_\theta(x_t|y_{1:t}) \quad (14)$$

where $\pi_B$ is defined in (3). From (10) we know that the required sufficient statistics are additive in the required form; therefore, the recursion in (12) is possible for the NMF model. The recursion for $S_{3,t}$ depends on the choice of the transition density $f_\psi$; however the recursions for $S_{1,t}$ and $S_{2,t}$ are the same for any model regardless of the choice of $f_\psi$. For this reason, we shall have a detailed look at (12) for the first two sufficient statistics $S_{1,t}$ and $S_{2,t}$.

For $S_{1,t}$, notice from (13) that, $p_\theta(x_{t-1}, z_{t-1}|y_{1:t-1}, x_t, z_t)$ does not depend on $z_t$. Moreover, the sufficient statistic $S_{1,t}$ is not a function of $z_{1:t}$. Therefore, $z_{t-1}$ in (12) integrates out, and $T_{1,t}$ is a function of $x_t$ only. Hence we will write it as $T_{1,t}(x_t)$. To sum up, we have the recursion

$$T_{1,t}(x_t) = x_t + \int T_{1,t-1}(x_{t-1}) p_\theta(x_{t-1}|x_t, y_{1:t-1}) dx_{t-1}.$$

For $S_{2,t}$, we claim that $T_{2,t}(x_t, z_t) = z_t + C_t(x_t)$ where $C_t(x_t)$ is a nonnegative $M \times K$ matrix valued function depending on $x_t$ but not $z_t$, and the recursion for $C_t(x_t)$ is expressed as

$$C_t(x_t) = \int \left( C_{t-1}(x_{t-1}) + \frac{B \odot \left(y_{t-1} x_{t-1}^T\right)}{(B x_{t-1})\, \mathbf{1}_K^T} \right) \times p_\theta(x_{t-1}|x_t, y_{1:t-1})\, dx_{t-1}$$

This claim can be verified by induction. Start with $t = 1$. Since $T_{2,0} = \mathbf{0}_{M\times K}$, we immediately see that $T_{2,t}(x_1, z_1) = z_1 = z_1 + C_1(x_1)$ where $C_1(x_1) = \mathbf{0}_{M\times K}$. For general $t > 1$, assume that $T_{2,t-1}(x_{t-1}, z_{t-1}) = z_{t-1} + C_{t-1}(x_{t-1})$. Using (13),

$$T_{2,t}(x_t, z_t) = z_t + \int \left(z_{t-1} + C_{t-1}(x_{t-1})\right) \pi_B(z_{t-1}|x_{t-1}, y_{t-1}) \times p_\theta(x_{t-1}|x_t, y_{1:t-1})\, dx_{t-1} dz_{t-1}$$

Now, observe that the $(m,k)$'th element of the integral $\int z_{t-1}\pi_B(z_{t-1}|x_{t-1}, y_{t-1})dz_{t-1}$ is $\frac{B(m,k)y_{t-1}(m)x_{t-1}(k)}{B(m,\cdot)x_{t-1}}$. So, we can write the integral as

$$\int z_{t-1}\pi_B(z_{t-1}|x_{t-1}, y_{t-1})dz_{t-1} = \frac{B \odot \left(y_{t-1} x_{t-1}^T\right)}{(B x_{t-1})\, \mathbf{1}_K^T}$$

So we are done. Using a similar derivation and substituting (14) into (13), we can show that

$$\widehat{S}_{2,t} = \int \left( C_t(x_t) + \frac{B \odot \left(y_t x_t^T\right)}{(B x_t)\, \mathbf{1}_K^T} \right) p_\theta(x_t|y_{1:t}) dx_t.$$

The online EM algorithm is a variation over the batch EM where the parameter is re-estimated each time a new observation is received. In this approach running averages of the sufficient statistics are computed [Elliott et al., 2002, Mongillo and Deneve, 2008, Cappé, 2009, 2011], [Kantas et al., 2009, Section 3.2.]. Specifically, let $\gamma = \{\gamma_t\}_{t\geq 1}$, called the step-size sequence, be a positive decreasing sequence satisfying $\sum_{t\geq 1}\gamma_t = \infty$ and $\sum_{t\geq 1}\gamma_t^2 < \infty$. A common choice is $\gamma_t = t^{-a}$ for $0.5 < a \leq 1$. Let $\theta_1$ be the initial guess of $\theta^*$ before having made any observations and at time $t$, let $\theta_{1:t}$ be the sequence of parameter estimates of the online EM algorithm computed sequentially based on $y_{1:t-1}$. Letting $u_t = (x_t, z_t)$ again to show for the general case, when $y_t$ is received, online EM computes

$$T_{\gamma,t}(u_t) = \int \left(\left(1-\gamma_t\right) T_{\gamma,t-1}(u_{t-1}) + \gamma_t s_t(u_{t-1}, u_t)\right) \times p_{\theta_{1:t}}(u_{t-1}|y_{1:t-1}, u_t) du_{t-1}, \tag{15}$$

$$\mathcal{S}_t = \int T_{\gamma,t}(u_t) p_{\theta_{1:t}}(u_t|y_{1:t}) du_t \tag{16}$$

and then applies the maximisation rule using the estimates $\mathcal{S}_t$. The subscript $\theta_{1:t}$ on the densities $p_{\theta_{1:t}}(u_{t-1}|y_{1:t-1}, u_t)$ and $p_{\theta_{1:t}}(u_t|y_{1:t})$ indicates that these laws are being computed sequentially using the parameter $\theta_k$ at time $k$, $k \leq t$. (See Algorithm 1 for details.) In practice, the maximisation step is not executed until a burn-in time $t_b$ for added stability of the estimators as discussed in Cappé [2009].

The online EM algorithm can be implemented exactly for a linear Gaussian state-space model [Elliott et al., 2002] and for finite state-space HMM's. [Mongillo and Deneve, 2008, Cappé, 2011]. An exact implementation is not possible for NMF models in general, therefore we now investigate SMC implementations of the online EM algorithm.

### *3.3 SMC implementation of the online EM algorithm*

Recall that $\{X_t, Y_t\}_{t\geq 1}$ is also a HMM with the initial and transition densities $\mu_\psi$ and $f_\psi$ in (1), and the observation density $g_B$ in (2). Since the conditional density $\pi_B(z_t|x_t, y_t)$ has a close form expression, it is sufficient to have a particle approximation to only $p_\theta(x_{1:t}|y_{1:t})$. This approximation can be performed in an online manner using a SMC approach. Suppose that we have the particle approximation to $p_\theta(x_{1:t}|y_{1:t})$ at time $t$ with $N$ particles

$$p_\theta^N(dx_{1:t}|y_{1:t}) = \sum_{i=1}^N w_t^{(i)} \delta_{x_{1:t}^{(i)}}(dx_{1:t}), \quad \sum_{i=1}^N w_t^{(i)} = 1, \tag{17}$$

where $x_{1:t}^{(i)} = (x_1^{(i)}, \ldots, x_t^{(i)})$ is the $n$'th path particle with weight $w_t^{(i)}$ and $\delta_x$ is the dirac measure concentrated at $x$. The particle approximation of the filter at time $t$ can be obtained from $p_\theta^N(dx_{1:t}|y_{1:t})$ by marginalization

$$p_\theta^N(dx_t|y_{1:t}) = \sum_{i=1}^N w_t^{(i)} \delta_{x_t^{(i)}}(dx_t).$$

At time $t+1$, for each $n$ we draw $x_{t+1}^{(i)}$ from a proposal density $q_\theta(x_{t+1}|x_t^{(i)})$ with a possible implicit dependency on $y_{t+1}$. We then update the weights according to the recursive rule:

$$w_{t+1}^{(i)} \propto \frac{w_t^{(i)} f_\psi(x_{t+1}^{(i)}|x_t^{(i)}) g_B(y_{t+1}|x_{t+1}^{(i)})}{q_\theta(x_{t+1}^{(i)}|x_t^{(i)})}.$$

To avoid weight degeneracy, at each time one can resample from (17) to obtain a new collection of particles $x_t^{(i)}$ with weights $w_t^{(i)} = 1/N$, and then proceed to the time $t+1$. Alternatively, this resampling operation can be done according to a criterion which measures the weight degeneracy [Doucet et al., 2000]. The SMC online EM algorithm for NMF models executing (15) and (16) based on the SMC approximation of $p_\theta(x_{1:t}|y_{1:t})$ in (17) is presented Algorithm 1.

*Algorithm 1.* **SMC online EM algorithm for NMF models**

- **E-step:** If t = 1, initialise $\theta_1$; sample $\widetilde{x}_1^{(i)} \sim q_{\theta_1}(\cdot)$, and set $w_1^{(i)} = \frac{\mu_{\psi_1}(\widetilde{x}_1^{(i)}) g_{B_1}(y_1|\widetilde{x}_1^{(i)})}{q_{\theta_1}(\widetilde{x}_1^{(i)})}$, $\widetilde{T}_{1,1}^{(i)} = \widetilde{x}_1^{(i)}$, $\widetilde{C}_1^{(i)} = 0$, $\widetilde{T}_{3,1}^{(i)} = s_{3,1}(\widetilde{x}_1^{(i)})$, $i = 1, \ldots, N$. If $t > 1$,
    - For $i = 1, \ldots, N$, sample $\widetilde{x}_t^{(i)} \sim q_{\theta_t}(\cdot|x_{t-1}^{(i)})$ and compute
    $$\widetilde{T}_{1,t}^{(i)} = (1-\gamma_t)T_{1,t-1}^{(i)} + \gamma_t \widetilde{x}_t^{(i)},$$
    $$\widetilde{T}_{3,t}^{(i)} = (1-\gamma_t)T_{3,t-1}^{(i)} + \gamma_t s_{3,t}(x_{t-1}^{(i)}, \widetilde{x}_t^{(i)})$$
    $$\widetilde{C}_t^{(i)} = (1-\gamma_t)C_{t-1}^{(i)} + (1-\gamma_t)\gamma_{t-1}\frac{B_t \odot \left(y_{t-1}x_{t-1}^{(i)T}\right)}{\left(B_t x_{t-1}^{(i)}\right)\mathbf{1}_K^T},$$
    $$\widetilde{w}_t^{(i)} \propto \frac{w_{t-1}^{(i)} f_{\psi_t}(\widetilde{x}_t^{(i)}|x_{t-1}^{(i)}) g_{B_t}(y_t|\widetilde{x}_t^{(i)})}{q_{\theta_t}(\widetilde{x}_t^{(i)}|x_{t-1}^{(i)})}.$$
    - Resample from particles $\{(\widetilde{x}_t, \widetilde{T}_{1,t}, \widetilde{C}_t, \widetilde{T}_{3,t})^{(i)}\}$ for $i = 1, \ldots, N$ according to the weights $\{\widetilde{w}_t^{(i)}\}_{i=1,\ldots,N}$ to get $\{(x_t, T_{1,t}, C_t, T_{3,t})^{(i)}\}$ for $i = 1, \ldots, N$ each with weight $w_t^{(i)} = 1/N$.

- **M-step:** If $t < t_b$, set $B_{t+1} = B_t$. Else, calculate using the particles before resampling

$$\mathcal{S}_{1,t} = \sum_{i=1}^{N} \widetilde{T}_t^{1(i)} \widetilde{w}_t^{(i)},$$

$$\mathcal{S}_{2,t} = \sum_{i=1}^{N} \left( \widetilde{C}_t^{(i)} + \gamma_t \frac{B_t \odot \left( y_t \widetilde{x}_t^{(i)T} \right)}{\left( B_t \widetilde{x}_t^{(i)} \right) \mathbf{1}_K^T} \right) \widetilde{w}_t^{(i)}$$

$$\mathcal{S}_{3,t} = \sum_{i=1}^{N} \widetilde{T}_t^{3(i)} \widetilde{w}_t^{(i)},$$

update the parameter $\theta_{t+1} = (B_{t+1}, \psi_{t+1})$, $B_{t+1} = \frac{\mathcal{S}_{2,t}}{\mathbf{1}_M [\mathcal{S}_{1,t}]^T}$, $\psi_{t+1} = \Lambda(\mathcal{S}_{3,t})$.

Algorithm 1 is a special application of the SMC online EM algorithm proposed in Cappé [2009] for a general state-space HMM, and it only requires $\mathcal{O}(N)$ computations per time step. Alternatively, one can implement an $\mathcal{O}(N^2)$ SMC approximation to the online EM algorithm, see Del Moral et al. [2009] for its merits and demerits over the current $\mathcal{O}(N)$ implementation. The $\mathcal{O}(N^2)$ is made possible by plugging the following SMC approximation to $p_\theta(x_{t-1}|x_t, y_{1:t-1})$ into (12)

$$p_\theta^N(dx_{t-1}|x_t, y_{1:t-1}) = \frac{p_\theta^N(dx_{t-1}|y_{1:t-1}) f_\psi(x_t|x_{t-1})}{\int p_\theta^N(dx_{t-1}|y_{1:t-1}) f_\psi(x_t|x_{t-1})}.$$

## 4. NUMERICAL EXAMPLES

### *4.1 Multiple basis selection model*

In this simple basis selection model, $X_t \in \{0,1\}^K$ determines which columns of $B$ are selected to contribute to the intensity of the Poisson distribution for observations. For $k = 1, \ldots, K$,

$$X_1(k) \sim \mu(\cdot), \quad \text{Prob}(X_t(k) = i | X_{t-1}(k) = j) = P(j, i),$$

where $\mu_0$ is a distribution over $\mathcal{X}$ and $P$ is such that $P(1,1) = p$ and $P(2,2) = q$. Estimation of $\psi = (p,q)$ can be done by calculating

$$\widehat{S}_{3,T} = \mathbb{E}_\theta \left[ \sum_{i=1}^{T} s_{3,i}(X_{i-1}, X_i) \middle| Y_{1:T} = y_{1:T} \right],$$

$$s_{3,t}(x_t, x_{t-1}) = \sum_{k=1}^{K} \begin{bmatrix} I_{(0,0)}(x_{t-1}(k), x_t(k)) \\ I_0(x_t(k)) \\ I_{(1,1)}(x_{t-1}(k), x_t(k)) \\ I_1(x_t(k)) \end{bmatrix}$$

and applying the maximisation rule $(p^{(j+1)}, q^{(j+1)}) = \Lambda(\widehat{S}_{3,t}^{(j)})$ where $\Lambda(\cdot)$ for this model is defined as

$$\Lambda(\widehat{S}_{3,t}) = (\widehat{S}_{3,t}(1)/\widehat{S}_{3,t}(2), \widehat{S}_{3,t}(3)/\widehat{S}_{3,t}(4)).$$

Figure 4.1 shows the estimation results of the exact implementation of online EM (with $\gamma_t = t^{-0.8}$ and $t_b = 100$) for the $8 \times 5$ matrix $B$ (assuming $(p,q)$ known) given the $8 \times 100000$ matrix $Y$ which is simulated $p = 0.8571, q = 0.6926$.

### *4.2 A relaxation of the multiple basis selection model*

In this model, the process $\{X_t \in (0,1)\}_{t \geq 1}$ is not a discrete one, but it is a Markov process on the unit interval $(0,1)$.

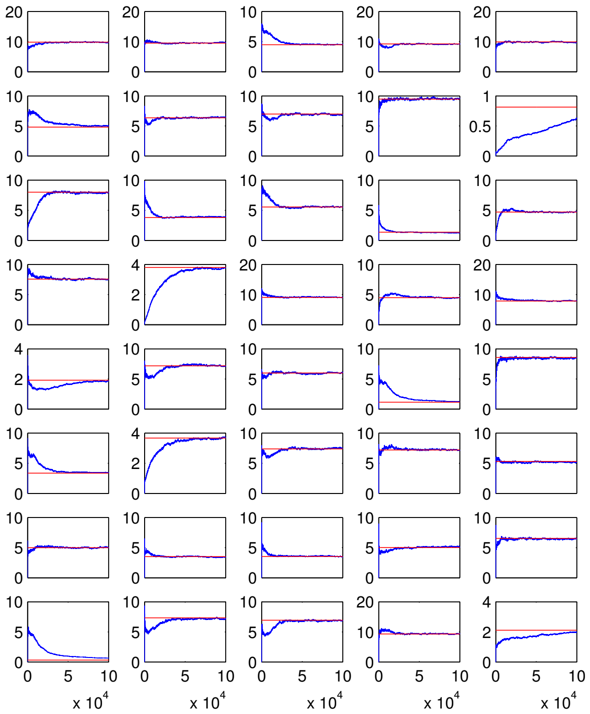

Fig. 1. Online estimation of $B$ in the NMF model in Section 4.1 using exact implementation of online EM for NMF. The $(i,j)$'th subfigure shows the estimation result for the $B(i,j)$ (horizontal lines).

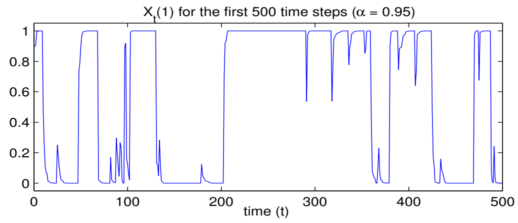

Fig. 2. A realisation of $\{X_t(1)\}_{t \geq 1}$ for $\alpha = 0.95$.

The law of the Markov chain for $\{X_t\}_{t \geq 1}$ is as follows: for $k = 1, \ldots, K$, $X_1(k) \sim \mathcal{U}(0,1)$, and

$$X_{t+1}(k) | (X_t(k) = x) \sim \rho(x)\mathcal{U}(0,x) + (1 - \rho(x))\mathcal{U}(x,1),$$

$$\rho(x) = \begin{cases} \alpha, & \text{if } x \leq 0.5 \\ 1 - \alpha, & \text{if } x > 0.5. \end{cases}$$

When $\alpha$ is close to 1, the process will spend most of its time around 0 and 1 with a strong correlation. (Figure 4.2 shows a realisation of $\{X_t(1)\}_{t \geq 1}$ for 500 time steps when $\alpha = 0.95$.) For estimation of $\alpha$, one needs to calculate

$$\widehat{S}_{3,T} = \mathbb{E}_\theta \left[ \sum_{i=1}^{T} s_{3,i}(X_{i-1}, X_i) \middle| Y_{1:T} = y_{1:T} \right],$$

$$s_{3,t}(x_{t-1}, x_t) = \begin{bmatrix} I_{A_{x_{t-1}(k)}}(x_{t-1}(k), x_t(k)) \\ I_{(0,1)\times(0,1)/A_{x_{t-1}(k)}}(x_{t-1}(k), x_t(k)) \end{bmatrix}$$

where, for $u \in (0,1)$, we define the set

$$A_u = ((0, 0.5] \times (0, u]) \cup ((0.5, 1) \times (u, 1)).$$

The maximisation step for $\alpha$ is characterised as

$$\Lambda(\widehat{S}_{3,t}) = \widehat{S}_{3,t}(1) / \left( \widehat{S}_{3,t}(1) + \widehat{S}_{3,t}(2) \right).$$

We generated a $8 \times 50000$ observation matrix $Y$ by using a $8 \times 5$ matrix $B$ and $\alpha = 0.95$. We used the SMC EM

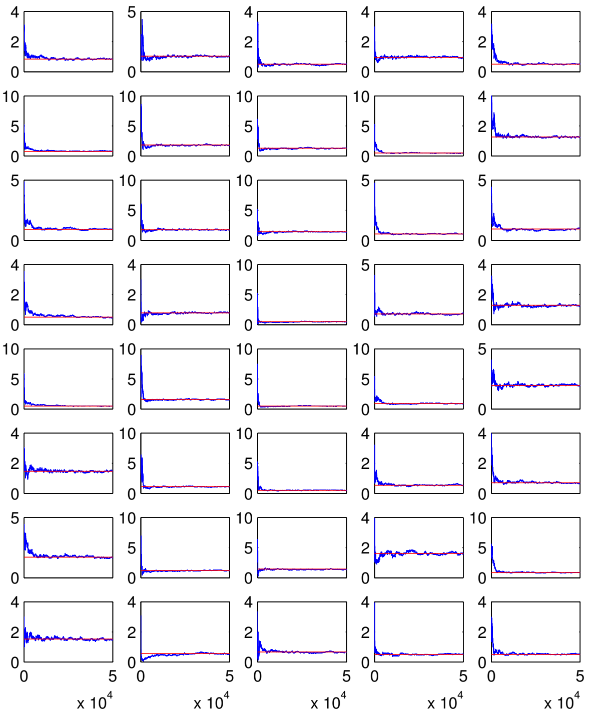

Fig. 3. Online estimation of $B$ in the NMF model in Section 4.2 using Algorithm 1. The $(i, j)$'th subfigure shows the estimation result for $B(i, j)$ (horizontal lines).

algorithm described in Algorithm 1 to estimate $B$ (assuming $\alpha$ known), with $N = 1000$ particles, $q_\theta(x_t|x_{t-1}) = f_\varphi(x_t|x_{t-1})$, $\gamma_t = t^{-0.8}$, and $t_b = 100$. Figure 4.2 shows the estimation results.

## 5. DISCUSSION

In this paper, we presented and online EM algorithm for NMF models with Poisson observations. We demonstrated an exact implementation and the SMC implementation of the online EM method on two separate NMF models. However, the method is applicable to any NMF model where the columns of the matrix $X$ can be represented as a stationary Markov process, e.g. the log-Gaussian process.

The results in Section 4 do not reflect on the generality of the method, i.e., only $B$ is estimated but the parameter $\varphi$ is assumed to be known, although we formulated the estimation rules for all of the parameters in $\theta$. Also, we perform experiments where the dimension of the $B$ matrix may be too small for realistic scenarios. Note that in Algorithm 1 we used the bootstrap particle filter, which is the simplest SMC implementation. The SMC implementation may be improved devising sophisticated particle filters, (e.g. those involving better proposal densities that learn from the current observation, SMC samplers, etc.), and we believe that only with that improvement the method can handle more complete problems with higher dimensions.

## REFERENCES

S. S. Bucak and B Gunsel. Incremental subspace learning via non-negative matrix factorization. *Pattern Recognition*, 42:788–797, 2009.

O Cappé. Online sequential Monte Carlo EM algorithm. In *Proc. IEEE Workshop on Statistical Signal Processing*, 2009.

O Cappé. Online EM algorithm for hidden Markov models. *Journal of Computational and Graphical Statistics*, 20 (3):728–749, 2011.

A. T. Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009:1–17, 2009.

P. Del Moral, A. Doucet, and S.S Singh. Forward smoothing using sequential Monte Carlo. Technical Report 638, Cambridge University, Engineering Department, 2009.

D. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. ”Math Challenges of the 21st Century”, 2000.

A. Doucet, S.J. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.

Robert J. Elliott, Jason J. Ford, and John B. Moore. On-line almost-sure parameter estimation for partially observed discrete-time linear systems with known noise characteristics. *International Journal of Adaptive Control and Signal Processing*, 16:435–453, 2002. doi: 10.1002/acs.703.

C. Fevotte and A. T. Cemgil. Nonnegative matrix factorisations as probabilistic inference in composite models. In *Proc. 17th European Signal Processing Conference (EUSIPCO'09)*, Glasgow, 2009.

Matthew Hoffman, David Blei, and Francis Bach. Online learning for latent dirichlet allocation. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 856–864, 2010.

N. Kantas, A. Doucet, S. S. Singh, and J. M. Maciejowski. An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. In *Proceedings IFAC System Identification (SysId) Meeting.*, 2009.

T.G. Kolda and B.W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. 42(8):30–37, 2009.

D. D. Lee and H. S. Seung. Learning the parts of objects with nonnegative matrix factorization. *Nature*, 401:788–791, 1999.

D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

A. Lefevre, F. Bach, and C. Fevotte. Online algorithms for nonnegative matrix factorization with the itakura-saito divergence. In *(WASPAA) IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 313–316, 2011.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online Learning for Matrix Factorization and Sparse Coding. February 2010. URL `http://arxiv.org/abs/0908.0050`.

G. Mongillo and S. Deneve. Online learning with hidden Markov models. *Neural Computation*, 20(7):1706–1716, 2008.

R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20, 2008.

A. P. Singh and G. J. Gordon. A unified view of matrix factorization models. In *ECML PKDD'08, Part II*, number 5212, pages 358–373. Springer, 2008.