Algorithme EM en ligne simulé pour la factorisation non-négative probabiliste

Olivier CAPPÉ, Cédric FÉVOTTE, David ROHDE

LTCI, Télécom ParisTech & CNRS
46, Rue Barrault, 75634 Paris Cédex 13, France
{cappe,fevotte,rohde}@telecom-paristech.fr

Résumé – On considère dans cette contribution l'estimation en ligne (ou adaptative) des facteurs communs dans les modèles probabilistes de factorisation non-négative. A cette fin, on introduit l'algorithme EM en ligne simulé qui étend l'approche proposée par Cappé et Moulines (2009) à des cas ou l'étape E de l'algorithme EM n'est pas explicite. L'algorithme résultant estime les facteurs au sens du maximum de vraisemblance (ou du MAP) en ayant recours à des simulations conditionnelles de type MCMC (Monte Carlo par Chaîne de Markov) au niveau de chaque observation. On détaille l'application de cette approche pour l'estimation de factorisations non-négatives probabilistes utilisant le critère dit d'Itakura-Saito en l'illustrant par des résultats obtenus en analyse non-supervisée de signaux audio.

Abstract – This paper describes a general purpose algorithm for online estimation in probabilistic latent factor models. The algorithm called *simulated online EM* is an extension of the online EM algorithm of Cappé & Moulines (2009) based on the use of MCMC simulations to approximate individual E-steps. The algorithm is applied to perform unsupervised analysis of audio signals using a probabilistic version of the Itakura-Saito non-negative matrix factorization model.

1 Introduction

On considère dans cette contribution la famille de modèles probabilistes à données latentes de la forme :

$$Y_n|H_n \sim g_{\sum_{k=1}^K \theta_k H_{n,k}}\,, \tag{1}$$

où $\{g_\lambda\}_{\lambda\in\Lambda}$ correspond à une famille exponentielle de distributions, H_n est un vecteur aléatoire latent de poids réels positifs associé à l'observation Y_n disponible au temps n et $\{\theta_k\}_{1\leq k\leq K}$ sont des facteurs communs, considérés comme des paramètres déterministes que l'on cherche à estimer. Par rapport aux approches plus classiques de factorisation non-négative de matrices, il est important de souligner que (1) constitue un modèle de chaque observation Y_n et non de l'ensemble des observations (s'il y a bien factorisation non-négative, il n'y a donc pas vraiment de matrice dans cette approche...) De plus H_n est considérée comme une variable aléatoire latente et non comme un paramètre, ce qui rend le modèle mieux conditionné mais également plus difficile à estimer. Les principaux exemples de modèles de ce type sont le modèle *LDA (Latent Dirichlet Allocation)* et ses variantes qui sont devenues au cours des dernières années l'approche dominante pour l'analyse non-supervisée de corpus de documents textuels [1, 2]. L'autre exemple important dont nous reparlerons plus loin est celui des approches probabilistes de la factorisation non-négative adaptées à l'analyse de spectres de signaux audio [6]. On trouvera d'autres exemples de modèles et d'applications dans [2, 8, 11].

Dans ces modèles, l'estimation des paramètres requiert des techniques avancées reposant soir sur l'utilisation de simulations de type Monte Carlo par chaîne de Markov (MCMC) ou l'utilisation d'approximations variationnelles. On s'intéresse ici à l'estimation en ligne ou adaptative de θ dans laquelle chaque observation Y_n est considérée tour à tour et sert à remettre à jour une estimation $\theta_n = (\theta_{n,1}, \ldots, \theta_{n,K})$ du paramètre inconnu. Ce type d'approches suscite actuellement un fort intérêt car il permet de s'attaquer, avec une complexité d'implémentation réaliste, à des problèmes dans lesquels le nombre d'observations disponibles est très grand, voire potentiellement infini (avec la nécessité de s'adapter à un flux constant de nouvelles observations). A ce jour, les seules approches d'estimation proposées dans ce cadre impliquent l'utilisation d'approximations variationnelles [9, 12]. Le travail présenté ici constitue le premier exemple de technique d'estimation en ligne qui repose directement sur le principe de l'algorithme EM en ligne proposé par [4] combiné avec l'utilisation de techniques de simulation MCMC appliquées *au niveau de chaque observation* pour approcher les calculs d'espérances conditionnelles difficiles dans ce type de modèles. L'intérêt de cette approche est de conduire à des algorithmes de complexité comparable à celle des algorithmes variationnels en ligne mais avec une analyse plus simple et des garanties plus fortes du fait de l'absence d'approximation variationnelle.

2 Algorithme EM en ligne simulé

Dans cette section, on présente le principe de l'approche proposée avec des notations plus générales où l'ensemble des don-

nées latentes est noté X_n (on verra ci-dessous que dans le cas des modèles factoriels les données latentes à considérer ne sont en général pas directement H_n). Pour se rattacher à l'approche EM en ligne, il est important que la vraisemblance complète (loi jointe de X_n et Y_n) appartienne à une famille exponentielle dans le sens où

$$p_\theta(x,y) \propto \exp\left(\phi'(\theta)s(x,y) - A(\theta)\right), \tag{2}$$

$\phi(\cdot)$ étant la fonction qui associe à θ les paramètres dits naturels, $s(x,y)$ est la statistique exhaustive, et $A(\cdot)$ est le logarithme de la constante de normalisation (le prime désignant la transposée). La vraisemblance des observations est définie, comme d'habitude dans ce type de modèles, par marginalisation : $f_\theta(y) = \int p_\theta(x,y)dx$.

Dans ce cadre, l'algorithme EM en ligne de [4] consiste en

$$\begin{aligned} S_n &= (1-\gamma_n)S_{n-1} + \gamma_n \mathbb{E}_{\theta_{n-1}}\left[s(X_n,Y_n)|Y_n\right], \\ \theta_n &= \bar{\theta}(S_n), \end{aligned} \tag{3}$$

où γ_n est une séquence de poids décroissants qui est typiquement choisie de la forme $\gamma_n = n^{-\alpha}$ avec $1/2 < \alpha < 1$. Dans l'équation (3), $\bar{\theta}$ est la fonction (supposée explicite) qui définit l'estimateur du maximum de vraisemblance associé à une valeur donnée de la statistique exhaustive :

$$\bar{\theta}(S) = \arg\max_\theta \{\phi'(\theta)s(x,y) - A(\theta)\}.$$

Par analogie avec (3), l'algorithme EM en ligne simulé que nous proposons est décrit ci-dessous.

Algorithme EM en ligne simulé

Pour $n = 1, 2, \ldots,$

Etant donné les valeurs courantes de la statistique S_{n-1} et du paramètre estimé $\theta_{n-1} = \bar{\theta}(S_{n-1})$, et au vu de la nouvelle observation Y_n, effectuer la mise à jour suivante :

$$\begin{aligned} &\tilde{X}_n^1, \ldots, \tilde{X}_n^m \overset{\text{indép.}}{\sim} p_{\theta_{n-1}}(x_n|Y_n), \\ &S_n = (1-\gamma_n)S_{n-1} + \gamma_n \frac{1}{m}\sum_{i=1}^m s(\tilde{X}_n^i, Y_n), \\ &\theta_n = \bar{\theta}(S_n). \end{aligned}$$

Dans l'algorithme ci-dessus $\tilde{X}_n^1, \ldots, \tilde{X}_n^m$ sont des simulations conditionnellement indépendantes sous la loi conditionnelle $p_{\theta_{n-1}}(x_n|Y_n)$ de la donnée latente X_n. Cependant dans les modèles factoriels, il est normalement possible de marginaliser partiellement par rapport à une sous composante $\tilde{H}_n^i$ de $\tilde{X}_n^i$ (voir le cas IS-NMF dans la section 3 ci-dessous). Dans ce cas, il est préférable de remplacer $s(\tilde{X}_n^i, Y_n)$ par son estimation "Rao-Blackwellisée" :

$$\mathbb{E}_{\theta_{n-1}}\left[s(X_n,Y_n)\,\middle|\,Y_n, \tilde{H}_n^i\right], \tag{4}$$

qui conduit à une estimation de plus faible variance de S_n.

L'analyse de cet algorithme sous les hypothèses de [4] montre qu'il est très proche de l'algorithme EM en ligne "exact" dans lequel on calcule explicitement les espérances conditionnelles. Asymptotiquement, le seul prix à payer pour l'utilisation de simulations est un accroissement de variance que l'ont peut chiffrer dans le cas de simulations supposées indépendantes et qui est peu significatif même pour des valeurs modérées de m (voir la proposition 1 de [10]). Il est n'est en particulier nullement nécessaire que m augmente au fur et à mesure des itérations pour garantir la convergence de l'algorithme.

Un point important à souligner est que l'algorithme proposé est en fait très différent des versions Monte Carlo usuelles de l'algorithme EM [13], y compris de l'algorithme SAEM (Stochastic Approximation EM) [5] qui respose également sur l'approximation stochastique. En effet, ces algorithmes sont destinés à approcher le comportement de l'algorithme EM dit en bloc ou *batch* dans lequel chaque itération implique de disposer de l'ensemble des observations $Y_1, \ldots, Y_N$ supposées en nombre fini. Ainsi, là où l'algorithme EM en ligne simulé ne nécessite de simuler que la donnée latente X_n correspondant à l'observation courante Y_n, chaque itération de l'algorithme SAEM requiert la simulation de l'ensemble des données latentes $X_1, \ldots, X_N$ associées aux observations $Y_1, \ldots, Y_N$. L'analyse de ces algorithmes est également différente car là où l'algorithme SAEM permet de déterminer les maximums (éventuellement locaux) de la vraisemblance des observations $Y_1, \ldots, Y_N$, l'algorithme EM en ligne simulé converge, lorsque n est grand, vers les minimums (éventuellement locaux) de la divergence de Kullback $D(\pi|f_\theta)$, où π désigne la loi des observations (supposées i.i.d.) [4]. En pratique pour réaliser les simulations conditionnelles de X_n sachant Y_n, il sera le plus souvent nécessaire de recourir à quelques (nombre noté L dans la suite) itérations d'un algorithme MCMC ce qui introduit un léger biais ainsi qu'une dépendance entre les $\tilde{X}_n^i$.

3 Application au modèle IS-NMF

On considère ici la version probabiliste du modèle IS-NMF (pour Itakura-Saito NMF) décrite dans [6, 7]. Celle-ci peut être décrite par les représentations graphiques dirigées de la figure 1.

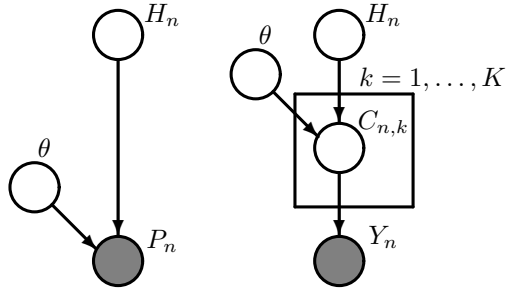

FIGURE 1 – Représentation graphique du modèle IS-NMF considéré : complétion minimale à gauche et modèle complet (famille exponentielle) à droite (où $P_n = |Y_n|^2$).

Dans la première représentation (à gauche), le module au

carré du spectre observé au temps n, P_n, suit une loi exponentielle de moyenne $\sum_{k=1}^{K} \theta_{fk} H_{n,k}$ à la fréquence f. Ainsi

$$\log p_\theta(P_n|H_n) = -\sum_{f=1}^{F} \log\left(\sum_{k=1}^{K} \theta_{fk} H_{n,k}\right) + \frac{P_{n,f}}{\sum_{k=1}^{K} \theta_{fk} H_{n,k}}, \quad (5)$$

où F désigne le nombre de fréquences et K est toujours le nombre de composantes. Le modèle est complété par la donnée de l'a priori $p(H_n)$ qui est choisi dans la famille conjuguée, c'est à dire, inverse gamma de paramètre (α, β). Le défaut de cette représentation directe est de ne pas appartenir à une famille exponentielle analogue à (2).

Pour utiliser l'approche détaillée dans la section précédente, il nous faut donc considérer la seconde représentation de la figure 1 qui implique plus de variables latentes. Ici, on considère de façon équivalente que l'observation Y_n est le spectre complexe (partie réelle et partie imaginaire) et l'on fait explicitement intervenir les K composantes $C_{n,k}$ complexes telles que $Y_{n,f} = \sum_{k=1}^{F} C_{n,k,f}$. Dans cette représentation, on a

$$\log p_\theta(C_n|H_n) = \sum_{k=1}^{K}\sum_{f=1}^{F} \log \mathcal{N}_C\left(C_{n,k,f}|0, \theta_{fk} H_{n,k}\right), \quad (6)$$

où $\mathcal{N}_C(z|\mu, \upsilon)$ désigne la densité gaussienne complexe circulaire (telle que $\mathbb{E}[Z] = \mu$ et $\mathbb{E}[|Z|^2] = \upsilon$). Tous calculs faits et en négligeant les termes ne dépendant pas de θ, on obtient

$$\log p_\theta(C_n|H_n) = C^{te} - \sum_{k=1}^{K}\sum_{f=1}^{F} \log(\theta_{fk}) + \frac{|C_{n,k,f}|^2}{\theta_{fk} H_{n,k}}. \quad (7)$$

Cette seconde représentation est donc sous forme exponentielle avec une statistique exhaustive pour θ_{fk} égale à $|C_{n,k,f}|^2/H_{n,k}$. L'algorithme proposé s'écrit donc dans ce cas sous la forme :

$$S_{n,fk} = (1-\gamma_n) S_{n-1,fk} + \gamma_n \frac{1}{m} \sum_{i=1}^{m} \frac{|\tilde{C}^i_{n,k,f}|^2}{\tilde{H}^i_{n,k}}, \quad (8)$$

$$\theta_{n,fk} = \frac{S_{n,fk} + b_n}{1 + a_n}. \quad (9)$$

Dans l'équation (9), (a_n, b_n) sont des hyperparamètres de lissage : le choix $(a_n, b_n) = (0, 0)$ correspond au maximum de vraisemblance tandis que le cas général correspond à une estimateur MAP (maximum a posteriori) avec un a priori inverse gamma sur θ. En pratique, l'utilisation du lissage est importante pour éviter que $\theta_{n,fk}$ ne prenne des valeurs nulles lorsque nombre d'observation est faibles. Dans la suite, on prend $b_n = b/n$ et $a_n = 1/n$ selon les recommandations de [3].

Pour simuler $\tilde{C}_{n,k,f}$ et $\tilde{H}_{n,k}$ conditionnellement à Y_n on utilise l'échantillonneur de Gibbs décrit dans [7] qui revient à tirer $(\tilde{C}_{n,k,f})_{1\leq k\leq K}$ conjointement sous une loi gaussienne multivariée complexe (avec une matrice de covariance de rang $K-1$) tandis que $\tilde{H}_{n,k}$ est tiré sous la loi inverse gamma de paramètres $\alpha + F$ et $\beta + \sum_{f=1}^{F} |\tilde{C}_{n,k,f}|^2/\theta_{fk}$. En pratique, on itère l'algorithme de Gibbs pendant L itérations et $(\tilde{C}^i_{n,k,f}, \tilde{H}^i_{n,k})_{1\leq i\leq m}$ correspondent aux m dernières valeurs simulées dans la chaîne MCMC.

Comme indiqué en (4), on a ici tout intérêt à utiliser une mise à jour Rao-Blackwellisé à la place de (9) en remplaçant $|\tilde{C}^i_{n,k,f}|^2/H^i_{n,k}$ par $E_{\theta_{n-1}}\left[|C_{n,k,f}|^2 \middle| \tilde{H}^i_n, Y_n\right]/\tilde{H}^i_{n,k}$. Conditionnellement à H_n, $C_{n,k,f}$ suit une loi gaussienne complexe, d'où l'expression

$$E_{\theta_{n-1}}\left[|C_{n,k,f}|^2 \middle| H_n, Y_n\right] = \left(\frac{\theta_{fk} H_{n,k}}{\sum_{j=1}^{K} \theta_{fj} H_{n,j}}\right)^2 |Y_{n,f}|^2 + \theta_{fk} H_{n,k}\left(1 - \frac{\theta_{fk} H_{n,k}}{\sum_{j=1}^{K} \theta_{fj} H_{n,j}}\right).$$

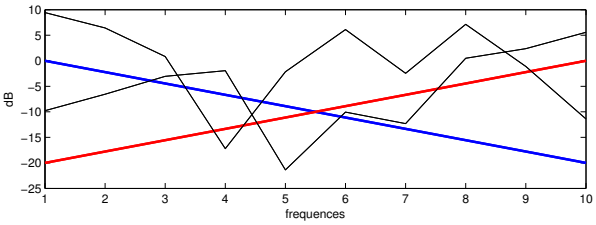

FIGURE 2 – Les deux profiles spectraux θ_{f1} et θ_{f2} et deux exmples d'observations $|Y_{n,f}|^2$.

Pour illustrer les résultats obtenus par cette approche, on considère tout d'abord un exemple jouet où le nombre de fréquences F est égal à 10, $K = 2$ et les deux profils spectraux correspondant aux colonnes de θ sont représentés en traits épais sur la figure 2. Sur la même figure, on a représenté également deux exemples d'observations obtenues selon le modèle génératif de la figure 1 en prenant des valeurs $\alpha = \beta = 1$ pour la loi a priori de $H_{n,1}$ et $H_{n,2}$. On constate la très grande variabilité due à la loi exponentielle (ce d'autant plus que la figure (2) utilise une échelle logarithmique), ce qui est une caractéristique bien connue du périodogramme de signaux stationnaires. Notons que dans ce modèle θ n'est identifiable que parce que les valeurs $H_{n,1}$ et $H_{n,2}$ des coefficients d'activations des deux spectres varient d'une observation à l'autre.

La figure 3 présente les résultats obtenus par l'algorithme EM en ligne simulé en utilisant $m = 1$, $L = 100$ itérations de l'algorithme de Gibbs pour chaque observation et un pas de l'algorithme de la forme $\gamma_n = n^{-0.6}$. On constate que l'algorithme estime les deux profils spectraux de façon consistante mais avec une surestimation notable dans les parties où l'un des profils spectraux est dominé par l'autre. Cette observation, générale dans ce modèle, est due au fait que conditionnellement à $H_{n,1}$ et $H_{n,2}$, l'observation à la fréquence f est de loi exponentielle de moyenne $H_{n,1}\theta_{f1} + H_{n,2}\theta_{f2}$. Lorsque l'une des deux valeurs est beaucoup plus faible que l'autre (par exemple pour $f = 1$, $\theta_{f2} = \theta_{f1}/100$), elle devient très difficile à estimer. Pour que les paramètres du modèle soient précisément estimés, il est indispensable dans ce modèle que la loi des coefficients d'activation $H_{n,k}$ soit très dispersée (loi à queue lourde ou qui charge la valeur 0 avec une probabilité non nulle).

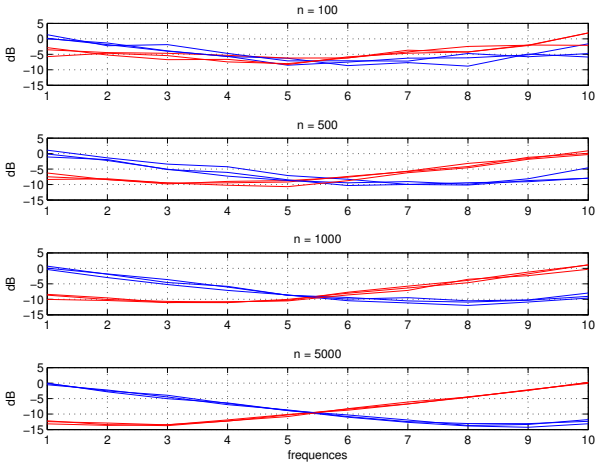

FIGURE 3 – Estimations de θ pour (de haut en bas) $n = 100, 500, 1000$ puis 5000 observations pour 3 initialisations distinctes.

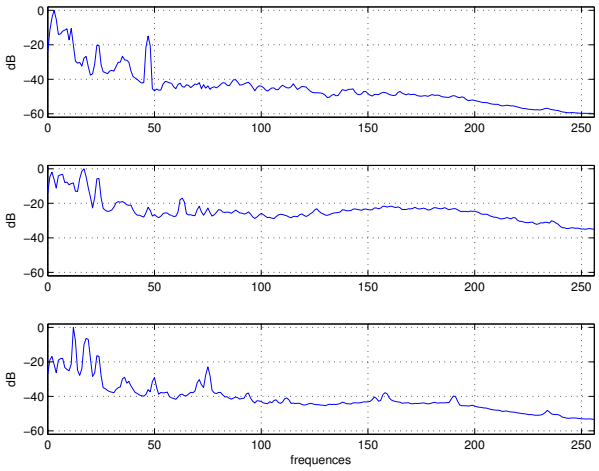

FIGURE 4 – Profiles spectraux $\theta_{f1}, \theta_{f2}, \theta_{f3}$ estimés (pour $n = 5603$).

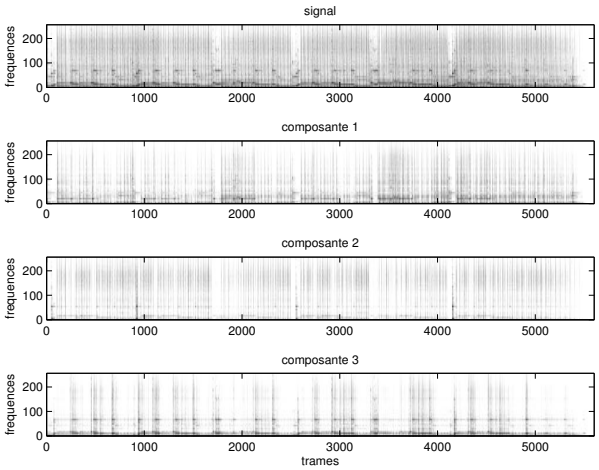

FIGURE 5 – Signal et composantes reconstruites.

Pour finir, les figures 4 et 5 résument les résultats obtenus sur un extrait audio de 1mn30s correspondant à 5603 spectres calculés sur $F = 256$ fréquences. L'algorithme est utilisé ici avec $L = 400$ et $m = 200$. Qualitativement la composante 1 correspond à un contenu basses fréquences, la deuxième a un contenu rythmique marqué (batterie) et la troisième isole pour l'essentiel un instrument soliste (vibraphone).

Références

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *JMLR*, 3 :993–1022, 2003.

[2] W. Buntine and A. Jakulin. Discrete component analysis. In *Subspace, Latent Structure and Feature Selection Techniques*. Springer-Verlag, 2006.

[3] O. Cappé. Online Expectation-Maximization. In K. Mengersen, M. Titterington, and C. P. Robert, editors, *Mixtures*. Wiley, 2011.

[4] O. Cappé and E. Moulines. On-line expectation-maximization algorithm for latent data models. *J. Roy. Statist. Soc. B*, 71(3) :593–613, 2009.

[5] B. Delyon, M. Lavielle, and E. Moulines. On a stochastic approximation version of the EM algorithm. *Ann. Statist.*, 27(1), 1999.

[6] C. Févotte, N. Bertin, and J.-L. Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence. with application to music analysis. *Neural Comput.*, 21(3) :793–830, Mar. 2009.

[7] C. Févotte and A. T. Cemgil. Nonnegative matrix factorisations as probabilistic inference in composite models. In *Proc. 17th European Signal Processing Conference (EUSIPCO'09)*, pages 1913–1917, 2009.

[8] K. A. Heller, S. Williamson, and Z. Ghahramani. Statistical models for partial membership. In *ICML 25*, pages 392–399, 2008.

[9] M. Hoffman, D. Blei, and F. Bach. Online learning for Latent Dirichlet Allocation. In *NIPS 23*, pages 856–864, 2010.

[10] D. Rohde and O. Cappé. Online maximum-likelihood estimation for latent factor models. In *IEEE Workshop on Statistical Signal Processing*, 2011.

[11] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *NIPS 20*, pages 1257–1264, 2008.

[12] I. Sato, K. Kurihara, and H. Nakagawa. Deterministic single-pass algorithm for LDA. In *NIPS 23*, pages 2074–2082, 2010.

[13] G. C. G. Wei and M. A. Tanner. A Monte-Carlo implementation of the EM algorithm and the poor man's Data Augmentation algorithms. *J. Am. Statist. Assoc.*, 85 :699–704, 1991.