Algorithmes de factorisation en matrices non-négatives fondée sur la β-divergence

Cédric FÉVOTTE[1], Jérôme IDIER[2]

[1]CNRS LTCI; Télécom ParisTech
37-39 ru Dareau, 75014 Paris, France

[2]CNRS IRCCyN; École Centrale de Nantes
1 rue de la Noë, 44000 Nantes, France

fevotte@telecom-paristech, jerome.idier@irccyn.ec-nantes.fr

Résumé – Cet article décrit des algorithmes pour la factorisation en matrices non-négatives dans le cas où la mesure de similarité appartient à la famille des β-divergences. Cette famille contient en particulier la distance euclidienne, la divergence de Kullback-Leibler généralisée et la divergence d'Itakura-Saito, des mesures de similarité classiques en traitement du signal et des images. Les algorithmes décrits reposent sur la construction (locale) d'une fonction majorante (globale) de la fonction objectif. Un premier type d'algorithme, dit de *majorisation-minimisation*, repose sur la minimisation itérative de cette fonction, donnant lieu à des mises à jour multiplicatives. Nous décrivons ensuite un nouveau type d'algorithme, dit de *majorisation-égalisation*, forme sur-relaxée du précédent qui produit en pratique une convergence plus rapide.

Abstract – This paper describes algorithms for nonnegative matrix factorization (NMF) with the β-divergence (β-NMF). The β-divergence is a family of cost functions parametrized by a single shape parameter β that takes the Euclidean distance, the Kullback-Leibler divergence and the Itakura-Saito divergence as special cases ($\beta = 2, 1, 0$ respectively). The proposed algorithms are based on a surrogate *auxiliary function* (an upper bound of the objective function constructed locally). We first describe a *majorization-minimization* (MM) algorithm that leads to multiplicative updates. Then we introduce the concept of *majorization-equalization* (ME) algorithm which produces updates that move along constant level sets of the auxiliary function and lead to larger steps than MM. Simulations illustrate the faster convergence of the ME approach.

1 Introduction

Étant donné une matrice $\mathbf{V}$ de dimension $F \times N$ à coefficients non-négatifs, i.e., positifs ou nuls, la factorisation en matrices non-négatives, dont nous utiliserons l'accronyme anglais NMF pour *nonnegative matrix factorization*, consiste à trouver une approximation

$$\mathbf{V} \approx \mathbf{WH}, \tag{1}$$

telle que les matrices $\mathbf{W}$ et $\mathbf{H}$ soient à coefficients non-négatifs et de dimensions $F \times K$ et $K \times N$, respectivement. Le «rang» K de la factorisation est souvent choisi tel que $F\,K + K\,N \ll F\,N$, produisant une réduction de dimension. Depuis son apparition dans un article de la revue *Nature* en 1999 [1] la NMF connaît une forte popularité dans les domaines de l'apprentissage et du signal/image ; elle a été appliquée à des problèmes divers tels que l'extraction de caractéristiques sémantiques de visages ou de texte [1], la transcription musicale [2], l'imagerie hyperspectrale [3], etc. La factorisation (1) est généralement obtenue par résolution du problème de minimisation suivant :

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{V}|\mathbf{WH}) \text{ sous contrainte } \mathbf{W} \geq 0, \mathbf{H} \geq 0,$$

où la notation $\mathbf{A} \geq 0$ exprime la non-négativité des coefficients de $\mathbf{A}$ (et non celle des valeurs propres) et où $D(\mathbf{V}|\mathbf{WH})$ est une mesure de similarité telle que

$$D(\mathbf{V}|\mathbf{WH}) = \sum_{f=1}^{F}\sum_{n=1}^{N} d([\mathbf{V}]_{fn}|[\mathbf{WH}]_{fn}).$$

La fonction $d(x|y)$ est une mesure de similarité entre scalaires (parfois appelée fonction de coût), i.e., une fonction de $\mathbb{R}_+ \times \mathbb{R}_+$ dans $\mathbb{R}_+$ avec un unique minimum égal à zéro en $x = y$. Une fonction de coût souvent considérée pour la NMF est la β-divergence [4, 5], une famille continue de divergences dont l'expression est donnée par

$$d_\beta(x|y) \stackrel{\text{def}}{=} \begin{cases} \frac{1}{\beta\,(\beta-1)}\left(x^\beta + (\beta-1)\,y^\beta - \beta\,x\,y^{\beta-1}\right) & \beta \in \mathbb{R}\backslash\{0,1\} \\ x\,\log\frac{x}{y} - x + y & \beta = 1 \\ \frac{x}{y} - \log\frac{x}{y} - 1 & \beta = 0. \end{cases} \tag{2}$$

La β-divergence prend comme cas particuliers la distance euclidienne, la divergence de Kullback-Leibler et la divergence d'Itakura-Saito ($\beta = 2, 1$ et 0, respectivement). Ces derniers cas sous-tendent des modèles d'observation Gaussien additif, Poisson et multiplicatif Gamma (voir [6]) et la β-divergence offre donc un continuum de modèles de bruit interpolant ces

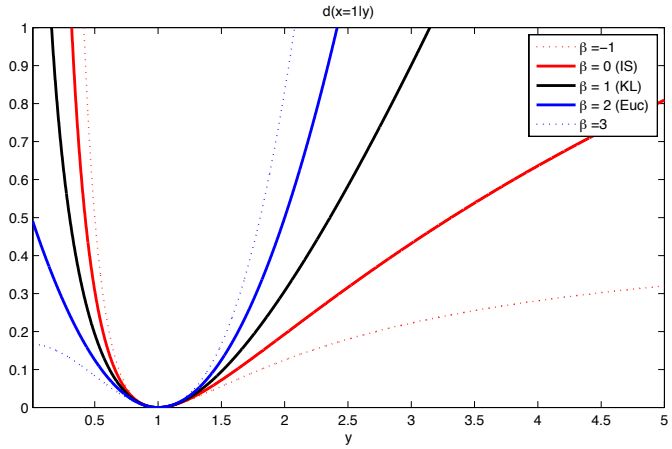

FIG. 1 – Gauche : β-divergence $d_\beta(x|y)$ pour $x = 1$. Considérée comme une fonction de y à x fixé, la β-divergence est convexe pour $1 \leq \beta \leq 2$. Pour $\beta < 0$, la divergence possède une asymptote horizontale de coordonnée $x^\beta/(\beta(\beta-1))$ lorsque $y \to \infty$. Pour $\beta > 1$, la divergence prend la valeur finie $x^\beta/(\beta(\beta-1))$ en $y = 0$, où la dérivée est également nulle pour $\beta > 2$.

cas particuliers, voir Figure 1. Le paramètre β offre donc un degré de liberté propre à la modélisation des données et sa valeur peut être ou bien fixée arbitrairement ou bien apprise sur un jeu d'apprentissage pour un contexte et une application donnés. Pour illustration, la Figure 2 rapporte des résultats d'interpolation obtenus par la NMF avec la β-divergence (nous utiliserons par la suite l'abbréviation β-NMF) pour différentes valeurs de β et K. La valeur de β peut être ajustée pour obtenir la qualité visuelle désirée. De manière similaire, la β-divergence a souvent été considérée en audio (voir références dans [7]), pour la décomposition du spectrogramme en composantes élémentaires, où la valeur de β peut être réglée de façon à optimiser des résultats de transcription ou de séparation de sources sur des données d'apprentissage.

Cet article présente trois types d'algorithmes pour la β-NMF. Un premier algorithme heuristique, connu de la littérature est présenté au paragraphe 2.1. Donc un deuxième temps nous décrivons deux algorithmes reposant sur la construction d'une fonction auxiliaire de la fonction objective originale : un algorithme de *majorisation-minimisation* est présenté au paragraphe 2.2 et un nouveau type d'algorithme, dit de *majorisation-égalisation*, est présenté au paragraphe 2.3. Cet article rapporte des résultats de la publication en revue à paraître [7] et font pour la première fois l'objet d'une publication en conférence. Les figures 2, 3 et 4 sont reproduites de [7], avec l'autorisation de MIT Press Journals.

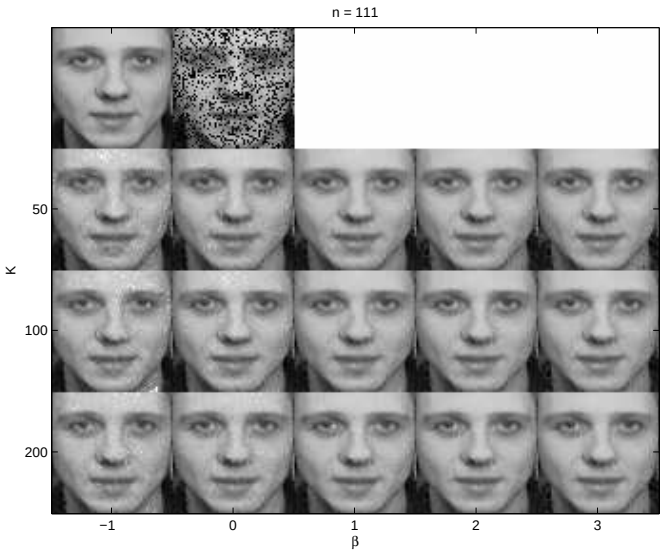

FIG. 2 – Résultat d'interpolation sur le jeu de données Olivetti, constitué de 400 images de visages (40 personnes, 10 prises de vue par personne), de dimensions 64×64 et codées en 8 bits. Les images forment les colonnes de la matrice $\mathbf{V}$, dont 25% des pixels ont été éliminés au hasard. Une β-NMF est appliquée sur la base des données restantes et les pixels manquants sont reconstruits en utilisant les valeurs données par la reconstruction $\mathbf{WH}$. Les reconstructions présentées sont obtenues pour $K = \{50, 100, 200\}$ et $\beta = \{-1, 0, 1, 2, 3\}$. Perceptiblement, les reconstructions obtenus avec $\beta = 3$ s'avèrent de meilleure qualité.

2 Algorithmes

2.1 Algorithme heuristique

Le premier algorithme pour la β-NMF est du à Cichocki et al. [8]. C'est un algorithme de descente de gradient à pas adaptatif, produisant des règles de mise à jour dites multiplicatives. Le même algorithme peut être construit en utilisant l'heuristique suivante, décrite dans [9]. Soit θ un coefficient de $\mathbf{W}$ ou $\mathbf{H}$. Avec la β-divergence la dérivée $\nabla_\theta D(\theta)$ du critère $D(\mathbf{V}|\mathbf{WH})$ par rapport à θ peut s'exprimer comme la différence de deux fonctions non-négatives tel que $\nabla_\theta D(\theta) = \nabla_\theta^+ D(\theta) - \nabla_\theta^- D(\theta)$. L'algorithme heuristique s'écrit alors comme

$$\theta \leftarrow \theta . \frac{\nabla_\theta^- D(\theta)}{\nabla_\theta^+ D(\theta)}, \tag{3}$$

assurant la non-négativité des itérés, étant donnée une initialisation à des valeurs positives. L'heuristique produit un algorithme de descente dans la mesure où θ est mis à jour vers la gauche (respectivement, droite) lorsque le gradient est positif (respectivement, négatif). Si la décroissance du critère sous cette règle de mise à jour a été observée en pratique, cette propriété n'a été que partiellement prouvée (seulement pour $\beta \in [1, 2]$, intervalle de convexité de la β-divergence par rapport à sa seconde variable) [10]. Nous produisons ci-après de

nouveaux algorithmes assurant la décroissance de la fonction objectif à chaque itération pour toute valeur de β.

2.2 Algorithme de majorisation-minimisation (MM)

L'algorithme MM présenté dans ce paragraphe repose sur la construction d'une fonction auxiliaire de la fonction objectif initiale. Précisons au préalable que nous nous plaçons dans un cadre d'optimisation itérative, dans lequel $\mathbf{W}$ est mise à jour conditionnellement à la valeur courante de $\mathbf{H}$, et réciproquement. On remarquera en outre que les mises à jour de $\mathbf{W}$ et $\mathbf{H}$ sont équivalentes, à une transposition près du problème ($\mathbf{V} \approx \mathbf{WH}$ est équivalent à $\mathbf{V}^T \approx \mathbf{H}^T\mathbf{W}^T$ et les rôles de $\mathbf{W}$ et $\mathbf{H}$ sont échangés). Enfin on remarquera que la fonction objectif (1) est séparable en les contributions (indépendantes) des lignes de $\mathbf{W}$ ou des colonnes de $\mathbf{H}$. Au final on pourra donc se concentrer sur la résolution du problème plus restreint de régression linéaire non-négative suivant :

$$\min_{\mathbf{h}\geq 0} C(\mathbf{h}) \stackrel{\text{def}}{=} D(\mathbf{v}|\mathbf{W}\mathbf{h}), \tag{4}$$

où $\mathbf{v} \in \mathbb{R}_+^F$, $\mathbf{W} \in \mathbb{R}_+^{F\times K}$ et $\mathbf{h} \in \mathbb{R}_+^K$.

Définition : fonction auxiliaire La fonction $G(\mathbf{h}|\tilde{\mathbf{h}})$, de $\mathbb{R}_+^K \times \mathbb{R}_+^K$ dans $\mathbb{R}_+$, est appelée fonction auxiliaire pour $C(\mathbf{h})$ si et seulement si

- $\forall \mathbf{h} \in \mathbb{R}_+^K, C(\mathbf{h}) = G(\mathbf{h}|\mathbf{h})$
- $\forall (\mathbf{h}, \tilde{\mathbf{h}}) \in \mathbb{R}_+^K \times \mathbb{R}_+^K, C(\mathbf{h}) \leq G(\mathbf{h}|\tilde{\mathbf{h}})$

En d'autres termes, la fonction auxiliaire $G(\mathbf{h}|\tilde{\mathbf{h}})$ est une fonction majorante de $C(\mathbf{h})$, tangente en $\mathbf{h} = \tilde{\mathbf{h}}$. La minimisation de $C(\mathbf{h})$ peut alors être remplacée par la minimisation itérative, potentiellement plus simple, de $G(\mathbf{h}|\tilde{\mathbf{h}})$. En effet, tout itéré $\mathbf{h}^{(i+1)}$ satisfaisant $G(\mathbf{h}^{(i+1)}|\mathbf{h}^{(i)}) \leq G(\mathbf{h}^{(i)}|\mathbf{h}^{(i)})$ produit un algorithme monotone (i.e., qui décroît la fonction objectif à chaque itération), car nous avons

$$C(\mathbf{h}^{(i+1)}) \leq G(\mathbf{h}^{(i+1)}|\mathbf{h}^{(i)}) \leq G(\mathbf{h}^{(i)}|\mathbf{h}^{(i)}) = C(\mathbf{h}^{(i)}). \tag{5}$$

L'itéré $\mathbf{h}^{(i+1)}$ est souvent choisi par

$$\mathbf{h}^{(i+1)} = \arg\min_{\mathbf{h}\geq 0} G(\mathbf{h}|\mathbf{h}^{(i)}), \tag{6}$$

définissant un algorithme MM (voir [11] pour une vue d'ensemble des algorithmes MM). Cependant, il est important de remarquer que tout itéré satisfaisant $G(\mathbf{h}^{(i+1)}|\mathbf{h}^{(i)}) \leq G(\mathbf{h}^{(i)}|\mathbf{h}^{(i)})$ produit un algorithme monotone, et nous considérerons une telle alternative à (6) au paragraphe 2.3.

Construction d'une fonction auxiliaire pour la β-NMF Il est facile de voir que la β-divergence peut s'écrire, pour toutes les valeurs de β, comme la somme d'une fonction convexe et d'une fonction concave. Aussi, une fonction auxiliaire à $C(\mathbf{h})$ peut être construite en majorant les parties convexes et concaves du critère séparément, la première par une inégalité de Jensen (avec égalité en $\tilde{\mathbf{h}}$) et la seconde par une approximation de Taylor au première ordre en $\tilde{\mathbf{h}}$ (utilisant la propriété qu'une fonction concave est majorée par sa tangente en tout point). La fonction auxiliaire résultante (convexe par construction) est donnée dans [7].

Mise à jour MM La minimisation (analytique) de la fonction auxiliaire construite produit la mise à jour suivante :

$$h_k^{\text{MM}} = \tilde{h}_k \left(\frac{\sum_f w_{fk}\, v_f\, \tilde{v}_f^{\beta-2}}{\sum_f w_{fk}\, \tilde{v}_f^{\beta-1}} \right)^{\gamma(\beta)} = \tilde{h}_k \left(\frac{\nabla_{h_k}^- C(\tilde{\mathbf{h}})}{\nabla_{h_k}^+ C(\tilde{\mathbf{h}})} \right)^{\gamma(\beta)} \tag{7}$$

où $\gamma(\beta) = 1/(2-\beta)$ si $\beta < 1$, $\gamma(\beta) = 1$ si $1 \leq \beta \leq 2$ et $\gamma(\beta) = 1/(\beta-1)$ si $\beta > 2$. L'algorithme MM produit donc des mises à jour multiplicatives, préservant la positivité de $\mathbf{h}$ étant donné des initialisations positives. L'algorithme MM diffère seulement de l'algorithme heuristique (3) par l'exposant $\gamma(\beta)$. Cet exposant vaut 1 pour l'intervalle de convexité de la β-divergence ($1 \leq \beta \leq 2$), intervalle de valeurs sur lequel l'algorithme MM et l'algorithme heuristique coïncident. En dehors de cet intervalle, l'algorithme MM produit un algorithme à décroissance garantie, ce qui n'est qu'expérimentalement vérifié pour l'algorithme heuristique.[1]

2.3 Algorithme de majorisation-égalisation (ME)

La mise à jour heuristique peut-être interprétée comme une version sur-relaxée de la mise à jour MM. En effet, pour toute valeur de β on peut montrer facilement

$$\forall k, \quad |h_k^{\text{H}} - \tilde{h}_k| \geq |h_k^{\text{MM}} - \tilde{h}_k|. \tag{8}$$

On observe empiriquement qu'«allonger» les pas accélère la convergence. Ce principe de sur-relaxation nous amène à proposer une stratégie de mise à jour par *majorisation-égalisation*, définie par

$$G(\mathbf{h}^{\text{ME}}|\tilde{\mathbf{h}}) = G(\tilde{\mathbf{h}}|\tilde{\mathbf{h}}). \tag{9}$$

Cette mise à jour consiste à chercher $\mathbf{h}$ sur l'«autre versant» de la fonction auxiliaire, qui est convexe par construction, voir l'illustration donnée par la figure 3. Il est à noter que l'équation (9) définit un ensemble de points plutôt qu'un point unique. La résolution de cette équation dans le cas général n'est pas possible, mais des solutions peuvent être obtenues analytiquement dans certains cas. En particulier, pour $\beta \in \{-1, 0, 1/2, 3/2, 2, 3\}$, une solution peut être obtenue par simple résolution d'un polynôme d'ordre 1 ou 2, voir détails dans [7]. Lorsque $0 \leq \beta \leq 2$, il peut être montré que la mise à jour ME produit des pas plus grands que la mise à jour heuristique, i.e., $\forall k, |h_k^{\text{ME}} - \tilde{h}_k| \geq |h_k^{\text{H}} - \tilde{h}_k| \geq |h_k^{\text{MM}} - \tilde{h}_k|$.

[1] Dans [7], nous donnons une preuve de monotonicité de l'algorithme heuristique pour $0 \leq \beta \leq 1$ basée sur la fonction auxiliaire construite.

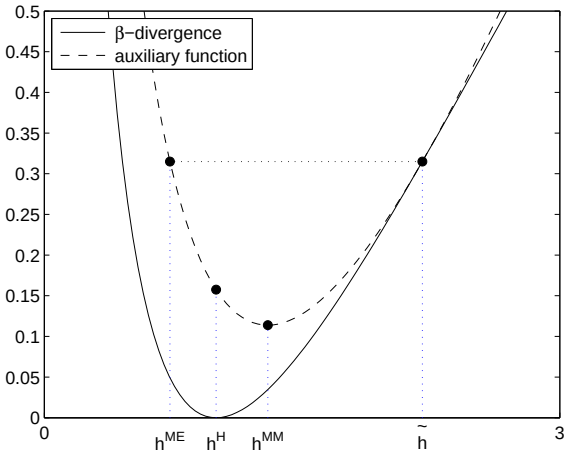

FIG. 3 – Illustration dans le cas scalaire (avec $\beta = 0.5$) des stratégies de mises à jour possible : heuristique h^{H}, majorisation-minimisation h^{MM} et majorisation-égalisation h^{ME}. Dans le cas scalaire, la mise à jour heuristique minimise la fonction objectif originale, mais cela n'est pas vrai en plus grande dimension.

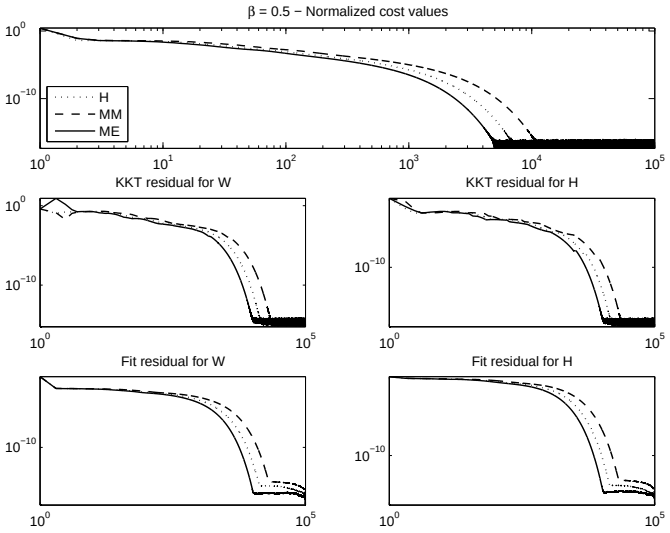

FIG. 4 – Diagnostic des algorithmes heuristique (H), ME et MM sur des données synthétiques, avec $\beta = 0.5$. Échelles logarithmiques en abscisse et ordonnée.

3 Simulations

Nous considérons un exemple de données exactement factorisable, i.e., tel que $\mathbf{V} = \mathbf{W}^{\star}\mathbf{H}^{\star}$, et où les facteurs sont générées comme les valeurs absolues d'un bruit gaussien, avec $F = 10$, $N = 25$ et $K = 5$. Une solution satisfaisant $D(\mathbf{V}|\mathbf{WH}) = 0$ peut donc être attendue dans ce cas. Nous avons exécuté les algorithmes heuristique, MM et ME avec ces données, en utilisant une initialisation aléatoire commune, dans le cas particulier (et arbitraire) où $\beta = 0.5$. La figure 4 présente 1) les valeurs de la fonction objectif au cours des itérations, 2) un résiduel montrant l'adéquation des itérés aux conditions de Karush-Kuhn-Tucker (KKT), 3) un résiduel montrant l'adéquation des itérés à la solution à convergence (i.e., au bout des 10^5 itérations). Cette simulation montre la convergence plus rapide de l'algorithme ME (pour une complexité équivalente aux deux autres). D'autres simulations, pour d'autres valeurs de β et sur des données réelles sont disponibles dans [7].

Références

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

[2] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'03)*, Oct. 2003.

[3] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics & Data Analysis*, 52(1) :155–173, Sep. 2007.

[4] A. Basu, I. R. Harris, N. L. Hjort, and M. C. Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3) :549–559, Sep. 1998.

[5] A. Cichocki and S. Amari. Families of Alpha- Beta- and Gamma- divergences : Flexible and robust measures of similarities. *Entropy*, 12(6) :1532–1568, June 2010.

[6] C. Févotte and A. T. Cemgil. Nonnegative matrix factorisations as probabilistic inference in composite models. In *Proc. 17th European Signal Processing Conference (EUSIPCO)*, pages 1913–1917, Glasgow, Scotland, Aug. 2009.

[7] C. Févotte and J. Idier. Algorithms for nonnegative matrix factorization with the beta-divergence. *Neural Computation*, in press.

[8] A. Cichocki, R. Zdunek, and S. Amari. Csiszar's divergences for non-negative matrix factorization : Family of new algorithms. In *Proc. 6th International Conference on Independent Component Analysis and Blind Signal Separation (ICA'06)*, pages 32–39, Charleston SC, USA, Mar. 2006.

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

[10] R. Kompass. A generalized divergence measure for nonnegative matrix factorization. *Neural Computation*, 19(3) :780–791, 2007.

[11] D. R. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 58 :30 – 37, 2004.