ALTERNATING DIRECTION METHOD OF MULTIPLIERS FOR NON-NEGATIVE MATRIX FACTORIZATION WITH THE BETA-DIVERGENCE

Dennis L. Sun

Stanford University
Department of Statistics
Stanford, CA, USA

Cédric Févotte

Laboratoire Lagrange
CNRS, OCA & University of Nice
Nice, France

ABSTRACT

Non-negative matrix factorization (NMF) is a popular method for learning interpretable features from non-negative data, such as counts or magnitudes. Different cost functions are used with NMF in different applications. We develop an algorithm, based on the alternating direction method of multipliers, that tackles NMF problems whose cost function is a beta-divergence, a broad class of divergence functions. We derive simple, closed-form updates for the most commonly used beta-divergences. We demonstrate experimentally that this algorithm has faster convergence and yields superior results to state-of-the-art algorithms for this problem.

Index Terms— non-negative matrix factorization, beta-divergence, alternating direction method of multipliers.

1. INTRODUCTION

In many applications, observations come in the form of vectors $\boldsymbol{v}_n$, and the data are assumed to be generated as linear combinations of relatively few underlying basis vectors or prototypes. The goal is to simultaneously learn the basis vectors $\{\boldsymbol{w}_k\}$ and activations $\{H_{kn}\}$ from the data so that $\boldsymbol{v}_n \approx \sum_{k=1}^{K} H_{kn}\boldsymbol{w}_k$. In matrix notation $V = \begin{bmatrix}\boldsymbol{v}_1 & \cdots & \boldsymbol{v}_N\end{bmatrix}$, the problem can be recast in terms of finding a low-rank factorization of V, i.e., matrices W and H such that:

$$V \approx WH. \tag{1}$$

There are many ways of factoring V into W and H, of which the most classical and well-known is principal components analysis (PCA). However, in many applications, the observations V are non-negative and it often makes sense to assume W and H to be non-negative as well. The problem of finding non-negative matrices W and H is known as non-negative matrix factorization (NMF) [1]. Examples include:

- hyperspectral imaging: $\boldsymbol{v}_n$ is vector of reflectance coefficients observed in pixel n, $\boldsymbol{w}_k$ corresponds to “end-member” spectra characteristic of the materials present in the observed scene, and h_{kn} are the mixing proportions or so-called “abundances” [2].
- topic modeling: $\boldsymbol{v}_n$ is a vector of word counts in document n, $\boldsymbol{w}_k$ corresponds to a “topic” (a distribution over words) and h_{kn} the representation of those topics in document n [3].
- audio signal processing: $\boldsymbol{v}_n$ is the magnitude or power spectrum at time n, $\boldsymbol{w}_k$ corresponds to an underlying spectral feature and h_{kn} the activation of that feature at time n [4].

The general form of the NMF problem is

$$\begin{aligned} &\text{minimize} && D(V|WH) \\ &\text{subject to} && W_{fk} \geq 0, H_{kn} \geq 0, \end{aligned} \tag{2}$$

where $D(V|\hat{V})$ represents some measure of divergence between V and its reconstruction $\hat{V}$.

We consider a general family of divergence functions known as the β-divergence D_β for $\beta \in \mathbb{R}$ [5, 6]. The divergence between two matrices is defined as the sum of the element-wise divergence, i.e., $D_\beta(V|\hat{V}) = \sum_{f,n} d_\beta(V_{fn}|\hat{V}_{fn})$, where d_β is defined for $\beta \in \mathbb{R}\backslash\{0,1\}$ by

$$d_\beta(x|y) = \frac{x^\beta}{\beta(\beta-1)} + \frac{y^\beta}{\beta} - \frac{xy^{\beta-1}}{\beta-1}. \tag{3}$$

This definition is extended to $\beta \in \{0, 1\}$ in the obvious way, by taking limits. The three divergence functions most commonly used with NMF are special cases of the β-divergence:

- $\beta = 2$ (Euclidean): $d(x|y) = \frac{1}{2}(x-y)^2$
- $\beta = 1$ (Kullback-Leibler): $d(x|y) = x \log \frac{x}{y} - x + y$
- $\beta = 0$ (Itakura-Saito): $d(x|y) = \frac{x}{y} - \log \frac{x}{y} - 1$.

Euclidean distance is perhaps the “obvious” way of measuring the divergence between two numbers, but in many applications involving non-negative data, it is not natural, namely because it underlies a real-valued Gaussian distribution for the data. In topic modeling where the data are counts, it is natural to minimize the Kullback-Leibler (KL) divergence, since it is equivalent to maximizing a Poisson log-likelihood [3]. In audio, the Itakura-Saito (IS) divergence is a natural choice because it underlies a suitable multiplicative exponential noise model of the spectrogram [7]. It is also scale-invariant, so it gives small time-frequency coefficients the same importance as larger ones, much like the human ear.

By considering the entire class of β-divergences, we develop algorithms not only for the much studied case of Euclidean distance, but also for other divergences that are used in specific applications.

2. EXISTING ALGORITHMS

Most existing work on algorithms for non-negative matrix factorization has focused on Euclidean NMF. To be explicit, the problem

D. Sun acknowledges support of the Ric Weiland Graduate Fellowship and Laboratoire Lagrange, which hosted him while this work was performed.

C. Févotte acknowledges support of CNRS MASTODONS project DISPLAY “Distributed processing for very large arrays in radioastronomy”.

under consideration is

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}||V - WH||_F^2 \\ &\text{subject to} \quad W_{fk} \geq 0, H_{kn} \geq 0. \end{aligned} \tag{4}$$

Since the problem is *bi-convex*, i.e., convex in H for W fixed and convex in W for H fixed, many algorithms adopt a alternating minimization approach. Paatero and Tapper used alternating least squares [8], calculating the least squares estimate and thresholding negative entries to zero. Lee and Seung derived multiplicative updates that alternately induce a descent in W and H [9]. Convergence to a stationary point is not guaranteed with either algorithm, and counterexamples can be found [10]. Lin proposed using projected gradient descent, an all-purpose algorithm with convergence guarantees [11]. State-of-the-art algorithms for (4) solve the subproblems for $W \geq 0$ and then $H \geq 0$ exactly using active-set methods [12]. There are also algorithms that attempt to update W and H simultaneously, such as HALS [13]. An up-to-date survey of algorithms for NMF can be found in [14, 15].

With the exception of projected gradient and multiplicative updates, the aforementioned algorithms rely on special properties of Euclidean distance and do not generalize readily to other members of the β-divergence family. Using a majorization-minimization approach, [6] showed that multiplicative updates could be derived for the entire family of β-divergences. Projected gradient descent for NMF with the β-divergence is straightforward. However, multiplicative updates have been more popular, owing to their:

1. **stability**: they do not require solving possibly ill-conditioned systems, as with Newton's method.
2. **ease of implementation**: each iteration requires just two lines of Matlab code, in contrast with projected gradient, which usually requires a more involved line search.
3. **linear complexity per iteration**: $O(FKN)$

However, they also suffer from problems, including:

1. **slow convergence**, especially tail convergence.
2. **asymptotic convergence to zeros:** since each update involves multiplying the matrix element-wise by a positive matrix, the iterates can only converge to zero values in the limit [2]. Stopping after any finite number of iterations will result in a dense matrix. Yet the sparsity pattern induced by the non-negativity constraint is often of interest in many applications [16].
3. **poor local optima?** Folk wisdom suggests that multiplicative updates are especially susceptible to become trapped in poor local optima [17].

The motivating question for our work is: is it possible to devise an algorithm that addresses each of these three shortcomings, while maintaining the advantages of multiplicative updates? That is, can we develop a faster algorithm that yields sparse solutions, but that is just as simple to implement without nested subroutines and complex bookkeeping?

3. ALTERNATING DIRECTION METHODS

Variable splitting is a powerful technique in optimization. The idea is to split the multiple occurrences of a single variable in a problem such as

$$\text{minimize} \sum_{i=1}^{p} f_i(x) \tag{5}$$

into multiple variables, with a constraint tying the variables together:

$$\text{minimize} \sum_{i=1}^{p} f_i(x_i) \text{ subject to } \sum_{i=1}^{p} A_i x_i = b. \tag{6}$$

The advantage is that $\sum_{i=1}^{p} f_i(x_i)$ can be optimized coordinate-wise, although there is a constraint coupling the problems together.

The alternating direction method of multipliers (ADMM) provides an elegant way of handling the constraint, while maintaining the separability of the objective. ADMM alternately optimizes the augmented Lagrangian

$$\begin{aligned} L_\rho(x_1, ..., x_p, u) = \sum_{i=1}^{p} f_i(x_i) + u^T \left(\sum_{i=1}^{p} A_i x_i - b \right) \\ + \frac{\rho}{2} || \sum_{i=1}^{p} A_i x_i - b||_2^2 \end{aligned} \tag{7}$$

with respect to each x_i individually, followed by dual ascent in u. Convergence is known for the special case $p = 2$ for f_1 and f_2 convex. A history and description of the method can be found in [18].

Although there are no guarantees when there are more than two blocks or when the function is non-convex, ADMM has also been considered for solving NMF problems. One approach is presented in [18], but it requires solving a quadratic program as a subproblem at each iteration, and thus is neither easy to implement nor fast. More similar to our approach is [19]. Both papers consider only the case of NMF with Euclidean distance. In the next section, we present an efficient ADMM algorithm that works for the entire β-divergence family, including Euclidean distance.

4. ALGORITHM

NMF, particularly with non-Euclidean divergences, is amenable to splitting for several reasons:

1. While $D_\beta(V|X)$ is simple to minimize with respect to X, $D_\beta(V|WH)$ is not simple to minimize with respect to W or H. Multiplicative updates implicitly tackle this problem, by majorizing D_β so that the W and H decouple [6]. In an ADMM context, a natural split would be to minimize $D_\beta(V|X)$ with the constraint $X = WH$.
2. The non-negativity constraints on W and H complicate the optimization over W and H. We can introduce new variables W_+ and H_+ to which the non-negativity constraints are applied, with the constraints $W = W_+$ and $H = H_+$.

In summary, the NMF problem (2) can be rewritten

$$\begin{aligned} \text{minimize} \quad & D_\beta(V|X) \\ \text{subject to} \quad & X = WH \\ & W = W_+, \ H = H_+ \\ & W_+ \geq 0, H_+ \geq 0. \end{aligned} \tag{8}$$

The notation above implies an augmented Lagrangian consisting of eight variables—five primal and three dual—although from the perspective of ADMM, this is only a three-block optimization: W, H, and (X, W_+, H_+). This is because the objective splits as a function of X, W_+, and H_+, so optimizing them separately is equivalent to optimizing them jointly:

$$
\begin{aligned}
L_\rho(X,&W,H,W_+,H_+,\alpha_X,\alpha_W,\alpha_H) = \\
&D_\beta(V|X) + \langle \alpha_X, X - WH\rangle + \frac{\rho}{2}||X - WH||_F^2 \\
&+ \langle \alpha_W, W - W_+\rangle + \frac{\rho}{2}||W - W_+||_F^2 \\
&+ \langle \alpha_H, H - H_+\rangle + \frac{\rho}{2}||H - H_+||_F^2. \qquad (9)
\end{aligned}
$$

The updates alternately optimize L_ρ with respect to each of the five primal variables, followed by gradient ascent in each of the three dual variables. This is summarized below.

Algorithm 1 ADMM for NMF with the β-divergence

```
inputs $V$
initialize $X, W, H, W_+, H_+, \alpha_X, \alpha_W, \alpha_H$
repeat
    $W^T \leftarrow (HH^T + I)\backslash(HX^T + W_+^T + \frac{1}{\rho}(H\alpha_X^T - \alpha_W^T))$
    $H \leftarrow (W^TW + I)\backslash(W^TX + H_+ + \frac{1}{\rho}(W^T\alpha_X - \alpha_H))$
    $X \leftarrow \operatorname{argmin}_{X \geq 0} D_\beta(V|X) + \langle \alpha_X, X\rangle + \frac{\rho}{2}||X - WH||_F^2$
    $W_+ \leftarrow \max(W + \frac{1}{\rho}\alpha_W, 0)$
    $H_+ \leftarrow \max(H + \frac{1}{\rho}\alpha_H, 0)$
    $\alpha_X \leftarrow \alpha_X + \rho(X - WH)$
    $\alpha_H \leftarrow \alpha_H + \rho(H - H_+)$
    $\alpha_W \leftarrow \alpha_W + \rho(W - W_+)$
until convergence return $W_+, H_+$
```

In the updates for W and H in Algorithm 1, we have used the Matlab notation $A\backslash b$ to denote the solution to the least squares problem $\operatorname{argmin}_x ||Ax - b||_2$. Since the matrices A in these updates are square and nonsingular, $A\backslash b = A^{-1}b$. Although least squares problems can be unstable in general, the addition of the identity matrix I in these cases stabilize the problem. The complexity of this operation is low, since A in this case is $K \times K$, where $K \ll F, N$.

The only update not provided in closed form above is the one for X, which we restate here for convenience:

$$X \leftarrow \operatorname*{argmin}_{X \geq 0} D_\beta(V|X) + \langle \alpha_X, X\rangle + \frac{\rho}{2}||X - WH||_F^2. \qquad (10)$$

Note that this is the only update that depends on β. As we shall see, (10) can be solved in closed form in the three most important cases $\beta = 0, 1, 2$. In general, (10) can be efficiently solved using Newton's method. We detail the closed-form updates for $\beta = 0, 1$ below. The updates for $\beta = 2$ can be derived similarly, although in the Euclidean case, the splitting of X and WH is unnecessary. The algorithm for $\beta = 2$ without this splitting can be found in [19].

Theorem 1. *For Kullback-Leibler divergence ($\beta = 1$),* (10) *is given by:*

$$X \leftarrow \frac{(\rho WH - \alpha_X - 1) + \sqrt{(\rho WH - \alpha_X - 1)^2 + 4\rho V}}{2\rho} \qquad (11)$$

where all operations are element-wise.

Proof. Substituting the expression for D_β, $\beta = 1$ into (10) and setting equal to zero, we obtain the condition

$$-\frac{V_{fn}}{X_{fn}} + 1 + (\alpha_X)_{fn} + \rho(X_{fn} - (WH)_{fn}) = 0.$$

Multiplying by X_{fn}, we obtain a quadratic equation; applying the quadratic formula, we obtain one positive and one negative root. The positive root is (11). □

Theorem 2. *For Itakura-Saito divergence ($\beta = 0$),* (10) *is given by the following series of updates:*

$$A \leftarrow \alpha_X/\rho - WH \qquad (12)$$

$$B_{fn} \leftarrow 1/(3\rho) - A_{fn}^2/9 \qquad (13)$$

$$C_{fn} \leftarrow -A_{fn}^3/27 + A_{fn}/(6\rho) + V_{fn}/2\rho \qquad (14)$$

$$D_{fn} \leftarrow B_{fn}^3 + C_{fn}^2 \qquad (15)$$

$$Y_{fn} \leftarrow \begin{cases} \left(C_{fn} + \sqrt{D_{fn}}\right)^{1/3} + \left(C_{fn} - \sqrt{D_{fn}}\right)^{1/3} & D_{fn} \geq 0 \\ 2\sqrt{-B_{fn}}\cos\left(\frac{1}{3}\cos^{-1}\frac{C_{fn}}{\sqrt{-B_{fn}^3}}\right) & D_{fn} < 0 \end{cases} \qquad (16)$$

$$X_{fn} \leftarrow Y_{fn} - A_{fn}/3 \qquad (17)$$

Proof. We substitute the expression for D_β, $\beta = 0$ into (10). Then, $X^* > 0$ is a minimizer if and only if the gradient vanishes at X^* and the Hessian is positive definite. Since the objective is separable in the entries of X, we can state this as

$$g_{fn}(X_{fn}^*) = 0 \qquad g'_{fn}(X_{fn}^*) > 0 \qquad \text{for all } f, n \qquad (18)$$

where g_{fn} denotes the derivative with respect to X_{fn}:

$$g_{fn}(x) = -\frac{V_{fn}}{x^2} + \frac{1}{x} + (\alpha_X)_{fn} + \rho(x - (WH)_{fn}).$$

Define $p_{fn}(x) = (x^2/\rho)g_{fn}(x)$ so that p_{fn} is a cubic polynomial. Then p_{fn} has the same roots as g_{fn}, and p'_{fn} has the same sign as g'_{fn} at the roots, so it is equivalent to check (18) for p_{fn}.

We can express p_{fn} explicitly as

$$p_{fn}(x) = x^3 + A_{fn}x^2 + (1/\rho)x - V_{fn}/\rho \qquad (19)$$

where A is defined in (12). We want a positive root $x_0 > 0$ of p_{fn} such that $p'_{fn} < 0$. At least one such root exists, since $p_{fn}(0) < 0$ and $p_{fn}(x) \to \infty$ as $x \to \infty$.

Next, we shift the cubic using the substitution $x = y - A_{fn}/3$ to obtain the depressed cubic

$$q_{fn}(y) = y^3 + 3B_{fn}y - 2C_{fn} \qquad (20)$$

where B, C are defined in (13), (14). The roots y_k of q_{fn} are related to the roots x_k of p_{fn} by $x_k = y_k - A_{fn}/3$. The discriminant D_{fn} of q_{fn} is then given by (15), and there are three cases, studied next.

$D_{fn} > 0$, one real root:

$$y_0 = \left(C_{fn} + \sqrt{D_{fn}}\right)^{1/3} + \left(C_{fn} - \sqrt{D_{fn}}\right)^{1/3}. \qquad (21)$$

Therefore, the corresponding root x_0 of p_{fn} must be positive and the minimizer of (10).

$D_{fn} = 0$, two distinct real roots:

y_0 as defined above and a double root $y_1 = y_2 = -y_0/2$. However, double roots correspond to point of inflections of q_{fn}, which means $g'(x_1) = 0$, so x_1 is not a minimizer of (10). Therefore, the relevant root is again y_0.

$D_{fn} < 0$, three distinct real roots:

$$y_k = 2\sqrt{-B_{fn}}\cos\left(\frac{1}{3}\cos^{-1}\frac{C_{fn}}{\sqrt{-B_{fn}^3}} - k\frac{2\pi}{3}\right)$$

for $k = 0, 1, 2$. If there are three roots, then p'_{fn} (and hence g'_{fn}) can only be positive at the largest and smallest roots. Since it is always true that $y_0 \geq y_1 \geq y_2$, it is sufficient to check only x_0 and x_2 (the latter only if $x_2 > 0$). For simplicity, in the implementation above, we have always taken x_0, which is guaranteed to be at least a local minima of (10).

In all three cases, to recover the corresponding root of p_{fn} from y_0, we re-apply the substitution: $x_0 = y_0 - A_{fn}/3$. We have omitted many details in this derivation which are standard [20]. We have instead focused on the simplifications that are possible because the problem only requires positive roots of p_{fn} at which $p'_{fn} > 0$. □

We note that these updates for optimizing (10) exactly in the case $\beta = 0$ could also be used with the ADMM algorithm in [21].

5. EXPERIMENTS

We first tested our algorithm on a moderately sized example, with $F = 200$, $N = 1000$, and $K = 100$. We generated ground truth matrices W_0 and H_0 and set $V = W_0 H_0$. Therefore, the true minimum value of the objective (2) is zero. We examine the performance of Algorithm 1 against the standard multiplicative updates [6]. In the ADMM algorithm, there is a tuning parameter ρ, which controls the convergence rate. A smaller value of ρ leads to larger step sizes, which can result in faster convergence but also instability. We considered four values of ρ and the two cases of the β-divergence examined above: $\beta = 1$ (Kullback-Leibler) and $\beta = 0$ (Itakura-Saito). The algorithms were implemented in Matlab.

The convergence of the different algorithms is shown in Figure 5 on a logarithmic scale. Runtime, rather than iteration number, is displayed on the x-axis to account for the higher per-iteration complexity of ADMM as compared with multiplicative updates. The experiments show that for values of ρ in the right range, ADMM can produce much faster convergence than multiplicative updates, achieving levels of accuracy that would take multiplicative updates several orders of magnitude longer to achieve. However, the convergence of ADMM can be slower than multiplicative updates if ρ is large, as represented by the red lines. When ρ is small, convergence can be unstable, as the magenta lines show, although they eventually reach a fast convergence rate. Note that ADMM does not guarantee monotone decrease in the objective, a downside that especially arises when ρ is small.

We also compared the two algorithms on an audio source separation task. We synthesized 10-second, 0 dB mixtures of speech from TIMIT and noise from AURORA. First, basis vectors were learned from held-out training speech and noise using Itakura-Saito NMF ($\beta = 0$) on the power spectrograms. Then, the activations of these basis vectors in the mixture were learned, again using NMF. This produces estimates of the speech and noise in the mixture [4].

The same initialization was used for both algorithms, and each algorithm was allowed to run for 20 seconds. ρ was set to 1 for ADMM. We then computed the signal-to-distortion (SDR) ratio of the recovered speech source [22]. The results are shown in Table 1. They demonstrate that ADMM achieves at least comparable, if not superior, separation performance to multiplicative updates, lending some support to the claim that the latter can be susceptible to poor local optima. Moreover, ADMM yielded sparse matrices (~80% of the estimated activations H for the mixture signal were zeros), whereas multiplicative updates produced no exact zeros.

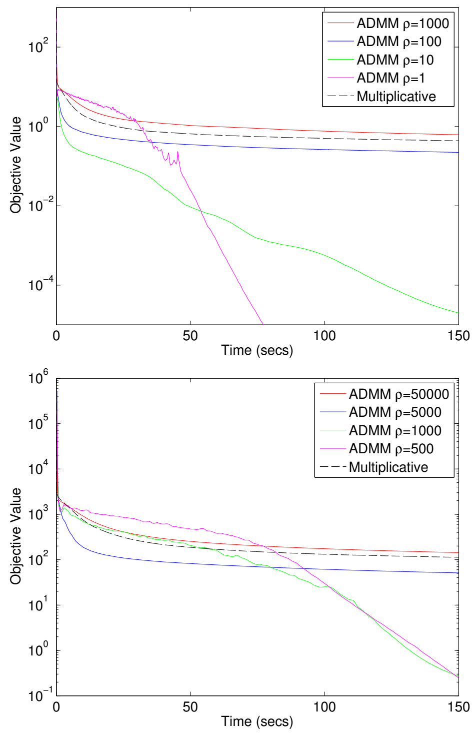

Fig. 1. The objective value (2) for $\beta = 1$ (top) and $\beta = 0$ (bottom) as a function of runtime. We compare Algorithm 1 for four settings of ρ with multiplicative updates on a synthetic example.

	car	street	subway	train
fpas0	**5.96** / 5.13	**6.71** / 6.08	**4.09** / 3.76	5.96 / **6.08**
fpkt0	5.60 / **5.70**	**5.92** / 5.78	**4.14** / 3.97	6.05 / **8.40**
mtdt0	3.83 / **4.17**	**5.35** / 3.05	**3.71** / 2.34	**5.48** / 5.43
mwew0	4.82 / **6.28**	**7.05** / 5.99	**4.56** / 2.78	5.30 / **7.96**

Table 1. SDR of the estimated speech using ADMM / multiplicative updates on combinations of 4 TIMIT speech examples (2 female, 2 male) and 4 AURORA noise examples. The higher SDR is **bold**.

6. CONCLUSION

We have demonstrated that the ADMM framework can be used to derive an algorithm for NMF with β-divergence that outperforms the state-of-the-art multiplicative updates used to solve these problems. ADMM has faster convergence and produces exact sparsity, and is as straightforward to implement, requiring only one additional tuning parameter ρ. However, we have also seen that the performance can also be sensitive to this parameter. Although there is some literature on choosing ρ automatically [23], we also envision applications where the practitioner can afford to monitor the convergence and tune ρ appropriately. As we have seen, ADMM can achieve a given level of accuracy orders of magnitude faster than multiplicative updates, so the tuning of ρ may be a small price to pay.

7. REFERENCES

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

[2] 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*, vol. 52, no. 1, pp. 155–173, Sep. 2007.

[3] Thomas Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd International Conference on Research and Development in Information Retrieval (SIGIR)*, 1999.

[4] 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.

[5] 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)*, Charleston SC, USA, Mar. 2006, pp. 32–39.

[6] Cédric Févotte and Jérôme Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[7] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, Mar. 2009.

[8] Pentti Paatero and Unto Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[9] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, vol. 13, pp. 556–562, 2001.

[10] Edward F Gonzalez and Yin Zhang, "Accelerating the lee-seung algorithm for non-negative matrix factorization," *Dept. Comput. & Appl. Math., Rice Univ., Houston, TX, Tech. Rep. TR-05-02*, 2005.

[11] Chih-Jen Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[12] Jingu Kim and Haesun Park, "Fast nonnegative matrix factorization: An active-set-like method and comparisons," *SIAM Journal on Scientific Computing*, vol. 33, no. 6, pp. 3261–3281, 2011.

[13] Andrzej Cichocki, Rafal Zdunek, and Shun-ichi Amari, "Hierarchical als algorithms for nonnegative matrix and 3d tensor factorization," in *Independent Component Analysis and Signal Separation*, pp. 169–176. Springer, 2007.

[14] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-ichi Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, Wiley, 2009.

[15] Jingu Kim, Yunlong He, and Haesun Park, "Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework," *Journal of Global Optimization*, pp. 1–35, 2013.

[16] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[17] A. Lefèvre, *Dictionary learning methods for single-channel source separation*, Ph.D. thesis, Télécom ParisTech, 2012.

[18] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[19] Yangyang Xu, Wotao Yin, Zaiwen Wen, and Yin Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Frontiers of Mathematics in China*, vol. 7, no. 2, pp. 365–384, 2012.

[20] Eric W. Weisstein, "Cubic formula. From MathWorld—A Wolfram Web Resource," 2013.

[21] Dennis L. Sun and Rahul Mazumder, "Non-negative matrix completion for bandwidth extension: a convex optimization approach.," in *IEEE Workshop on Machine Learning for Signal Processing*, 2013.

[22] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 4, pp. 1462–1469, 2006.

[23] Euhanna Ghadimi, André Teixeira, Iman Shames, and Mikael Johansson, "On the optimal step-size selection for the alternating direction method of multipliers?," *SIAM Journal on Optimization*, vol. 20, no. 3, pp. 1157–1170, 2009.