PRIMAL-DUAL ALGORITHMS FOR NON-NEGATIVE MATRIX FACTORIZATION WITH THE KULLBACK-LEIBLER DIVERGENCE

Felipe Yanez[*]

INSEAD, Fontainebleau, France

Francis Bach

INRIA/École normale supérieure, Paris, France

ABSTRACT

Non-negative matrix factorization (NMF) approximates a given matrix as a product of two non-negative matrix factors. Multiplicative algorithms deliver reliable results, but they show slow convergence for high-dimensional data and may be stuck away from local minima. Gradient descent methods have better behavior, but only apply to smooth losses. For non-smooth losses such as the Kullback-Leibler (KL) loss, surprisingly, these methods are lacking. In this paper, we propose a first-order primal-dual algorithm for non-negative decomposition problems (one of the two factors is fixed) with the KL distance. All required computations may be obtained in closed form and we provide an efficient heuristic way to select step-sizes. By using alternating optimization, our algorithm readily extends to NMF and, on synthetic or real world data, it is either faster than existing algorithms, or leads to improved local optima, or both.

***Index Terms*—** Non-negative matrix factorization, Primal-dual approaches, Optimization, Kullback-Leibler divergence.

1. INTRODUCTION

Non-negative matrix factorization (NMF) is a method that aims at finding part-based linear representations of non-negative data by factorizing it as the product of two low-rank non-negative matrices [1, 2]. Two well-known multiplicative updates algorithms (MUAs) for NMF were introduced in [3], minimizing either the least-squares or Kullback-Leibler (KL) loss. MUAs extend to other losses and have been reported in different applications, e.g., face recognition [4], and music analysis [5]. However, they have slow convergence rate in high-dimensional data and are susceptible to become trapped in poor local optima [6]. Gradient descent methods for NMF provide additional flexibility and fast convergence, but only apply to the minimization of the least-squares loss [6, 7]. The goal of this paper is to propose a strong and simple alternative to MUAs, by providing a similar first-order method for the KL distance, with updates as cheap as in MUAs. Our method builds on [8] which consider the alternating direction method of multipliers (ADMM). We instead rely on the Chambolle-Pock algorithm [9], which may be seen as a linearized version of ADMM, and thus we may reuse some of the tools developed in [8] while having an empirically faster algorithm. Because of the lack of smoothness of the KL loss (i.e., second-order derivatives not bounded), some techniques such as the forward-backward algorithm cannot be applied. We thus need methods adapted to non-smooth problems, where the algorithm in [9] is natural.

The main contributions of this paper are as follows:

- A primal-dual formulation for the convex KL decomposition problem (Section 3.1), and an extension to the non-convex problem of NMF (Section 3.3).
- Based on convergence proofs, a purely data-driven way to select all step-sizes (Section 3.2).
- A Matlab implementation (Algorithm 2), available at `http://www.di.ens.fr/~fbach/nmf_fpa.html`

2. PROBLEM FORMULATION

Let $\mathbf{V} \in \mathbb{R}_+^{n \times m}$ denote the given matrix formed by m non-negative column vectors of dimensionality n. Considering $r \leq \min(n, m)$, let $\mathbf{W} \in \mathbb{R}_+^{n \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times m}$ such that $\mathbf{V} \approx \mathbf{WH}$. Then, the NMF problem with the KL distance is

$$\underset{\mathbf{W},\mathbf{H} \geq 0}{\text{minimize}} \quad D(\mathbf{V}\|\mathbf{WH}), \tag{1}$$

with $D(\mathbf{V}\|\mathbf{WH}) = \sum_{i,j} -\mathbf{V}_{ij} \left\{ \log\left(\frac{(\mathbf{WH})_{ij}}{\mathbf{V}_{ij}} \right) + 1 \right\} + (\mathbf{WH})_{ij}$. Problem (1) is non-convex in both factors simultaneously, whereas convex in each factor separately. These non-negative decomposition (ND) problems are defined as follows:

$$\underset{\mathbf{W} \geq 0}{\text{minimize}}\ D(\mathbf{V}\|\mathbf{WH}) \quad \text{and} \quad \underset{\mathbf{H} \geq 0}{\text{minimize}}\ D(\mathbf{V}\|\mathbf{WH}). \tag{2}$$

2.1. Multiplicative updates algorithm (MUA)

The multiplicative updates may be derived from expectation-maximization (EM) for a certain probabilistic model [3, 10]. The complexity per iteration of this algorithm is $O(rmn)$.

$$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{\sum_{\mu=1}^m \mathbf{H}_{a\mu} \mathbf{V}_{i\mu} / (\mathbf{WH})_{i\mu}}{\sum_{\nu=1}^m \mathbf{H}_{a\nu}}, \text{ and}$$

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} \frac{\sum_{i=1}^n \mathbf{W}_{ia} \mathbf{V}_{i\mu} / (\mathbf{WH})_{i\mu}}{\sum_{k=1}^n \mathbf{W}_{ka}}.$$

[*]F.Y. performed the work while at INRIA/École normale supérieure.

This project was partially supported by the grant ERC SIERRA 239993.

2.2. Alternating direction method of multipliers (ADMM)

In [8], Problem (1) is reformulated as

$$\begin{aligned} \text{minimize} \quad & D(\mathbf{V}\|\mathbf{X}) \\ \text{subject to} \quad & \mathbf{X}=\mathbf{YZ},\ \mathbf{Y}=\mathbf{W},\ \mathbf{Z}=\mathbf{H} \\ & \mathbf{W}\geq 0,\ \mathbf{H}\geq 0. \end{aligned}$$

The updates for the primal variables $\mathbf{W}$, $\mathbf{H}$, $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ involve certain proximal operators for the KL loss which are the same as ours in Section 3.1:

$$\begin{aligned} \mathbf{Y}^\top &\leftarrow \left(\mathbf{Z}\mathbf{Z}^\top+\mathbf{I}\right)^{-1}\left(\mathbf{Z}\mathbf{X}^\top+\mathbf{W}^\top+\tfrac{1}{\rho}\left(\mathbf{Z}\alpha_{\mathbf{X}}^\top-\alpha_{\mathbf{Y}}^\top\right)\right) \\ \mathbf{Z} &\leftarrow \left(\mathbf{Y}^\top\mathbf{Y}+\mathbf{I}\right)^{-1}\left(\mathbf{Y}^\top\mathbf{X}+\mathbf{H}+\tfrac{1}{\rho}\left(\mathbf{Y}^\top\alpha_{\mathbf{X}}-\alpha_{\mathbf{Z}}\right)\right) \\ \mathbf{X} &\leftarrow \frac{(\rho\mathbf{Y}\mathbf{Z}-\alpha_{\mathbf{X}}-\mathbf{1})+\sqrt{(\rho\mathbf{Y}\mathbf{Z}-\alpha_{\mathbf{X}}-\mathbf{1})^2+4\rho\mathbf{V}}}{2\rho} \\ \mathbf{W} &\leftarrow \left(\mathbf{Y}+\tfrac{1}{\rho}\alpha_{\mathbf{Y}}\right)_+ \quad \text{and} \quad \mathbf{H} \leftarrow \left(\mathbf{Z}+\tfrac{1}{\rho}\alpha_{\mathbf{Z}}\right)_+. \end{aligned}$$

These primal updates require solving linear systems of size $r\times r$, but that the overall complexity remains $O(rmn)$ per iteration. Note that the parameter $\rho\in\mathbb{R}_+$ needs to be tuned. The dual variables $\alpha_{\mathbf{X}}$, $\alpha_{\mathbf{Y}}$ and $\alpha_{\mathbf{Z}}$ are updated as:

$$\begin{aligned} \alpha_{\mathbf{X}} &\leftarrow \alpha_{\mathbf{X}}+\rho\left(\mathbf{X}-\mathbf{Y}\mathbf{Z}\right) \\ \alpha_{\mathbf{Y}} &\leftarrow \alpha_{\mathbf{Y}}+\rho\left(\mathbf{Y}-\mathbf{W}\right) \\ \alpha_{\mathbf{Z}} &\leftarrow \alpha_{\mathbf{Z}}+\rho\left(\mathbf{Z}-\mathbf{H}\right). \end{aligned}$$

Our approach has the following differences: (i) we aim at solving alternatively *convex* problems with a few steps of primal-dual algorithms for convex problems, as opposed to aiming at solving directly the non-convex problem with an iterative approach, (ii) for the ND problems, we have certificates of optimality, which can be of use for online methods for which ND problems are repeatedly solved [11], and (iii) we use a different splitting method, namely as in [9], which does not require matrix inversions, and which allows us to compute all step-sizes in a data-driven way.

3. PROPOSED METHOD

We formulate the ND as a first-order primal-dual algorithm (Algorithm 1), extending it then to NMF (Algorithm 2).

3.1. Primal and dual computation

We consider a vector $a\in\mathbb{R}_+^p$ and a matrix $K\in\mathbb{R}_+^{p\times q}$ as known parameters, and $x\in\mathbb{R}_+^q$ as an unknown vector to be estimated such that $a\approx Kx$. We aim at minimizing the KL divergence between a and Kx. This is equivalent to a ND as defined in (2), considering a as a column of the given data, K as the fixed factor, and x as a column of the estimated factor. The ND problem with KL loss is thus

$$\underset{x\in\mathbb{R}_+^q}{\text{minimize}} \ -a^\top\left(\log(Kx\oslash a)+\mathbf{1}\right)+\mathbf{1}^\top Kx, \qquad (3)$$

where $\oslash$ represents the entry-wise division operator, and $\mathbf{1}$ is a vector where every element is equal to 1. Based on [9], the primal in (3) can be written as $\min_x F(Kx)+G(x)$ with $F(z)=-a^\top\left(\log(z\oslash a)+\mathbf{1}\right)$, and $G(x)=\mathbb{1}_{x\succeq 0}+\mathbf{1}^\top Kx$. The idicator function $\mathbb{1}_{x\succeq 0}$ gives 0 if all components of x are non-negative and $+\infty$ otherwise. The dual is then $\max_y -F^*(y)-G^*(-K^\top y)$ with $F^*(y)=-a^\top\log(-y)$ and $G^*(y)=\mathbb{1}_{y\preceq K^\top\mathbf{1}}$, i.e.,

$$\underset{K^\top(-y)\preceq K^\top\mathbf{1}}{\text{maximize}} \quad a^\top\log(-y). \qquad (4)$$

Algorithm 1: First-order primal-dual algorithm [9].

Select $K\in\mathbb{R}_+^{p\times q}$, $x\in\mathbb{R}_+^q$, σ, τ, $N>0$;
Set $\bar{x}=x_{old}=x$, $y=-a\oslash Kx$;
for N *iterations* **do**
  $y\leftarrow \mathbf{prox}_{\sigma F^*}(y+\sigma K\bar{x})$;
  $x\leftarrow \mathbf{prox}_{\tau G}(x-\tau K^\top y)$;
  $\bar{x}\leftarrow 2x-x_{old}$;
  $x_{old}\leftarrow x$;
end
return $x^\star=x$.

The step-sizes are σ and τ. The proximal operator definition is $\mathbf{prox}_{\tau F}(x)=\arg\min_y\left\{\|x-y\|^2/2\tau+F(y)\right\}$ [12, 13]. As shown in [14], using F^* and G we can obtain closed-form solution operators $\mathbf{prox}_{\sigma F^*}(y)=\frac{1}{2}\left(y-\sqrt{y\circ y+4\sigma a}\right)$, and $\mathbf{prox}_{\tau G}(x)=\left(x-\tau K^\top\mathbf{1}\right)_+$.

3.2. Automatic heuristic selection of step-sizes

Based on the convergence proofs [9, Theorem 1], we describe an automatic heuristic selection of step-sizes. These results state that (a) the step-sizes have to satisfy $\tau\sigma\|K\|^2<1$, where $\|K\|=\max\{\|Kx\|:\|x\|\leq 1\}$ is the largest singular value of K; and (b) the convergence rate is controlled by the quantity $C=\frac{\|y_0-y^\star\|^2}{2\sigma}+\frac{\|x_0-x^\star\|^2}{2\tau}$, where $(x^\star,y^\star)$ is an optimal primal/dual pair. If $(x^\star,y^\star)$ was known, we could thus consider $\min_{\sigma,\tau}\frac{\|y_0-y^\star\|^2}{2\sigma}+\frac{\|x_0-x^\star\|^2}{2\tau}$ with the constraint $\tau\sigma\|K\|^2=1$. Applying first order conditions we get

$$\sigma=\frac{\|y_0-y^\star\|}{\|x_0-x^\star\|}\frac{1}{\|K\|} \quad \text{and} \quad \tau=\frac{\|x_0-x^\star\|}{\|y_0-y^\star\|}\frac{1}{\|K\|}.$$

However, we do not know the optimal pair $(x^\star,y^\star)$ and we use heuristic replacements. That is, we consider the unknown constants α and β, and assume that $x^\star=\alpha\mathbf{1}$ and $y^\star=\beta\mathbf{1}$. This gives us $\|x_0-x^\star\|\approx|\alpha|\sqrt{q}$ and $\|y_0-y^\star\|\approx|\beta|\sqrt{p}$. Plugging $x^\star$ to Problem (3) we find $\alpha=\frac{\mathbf{1}^\top a}{\mathbf{1}^\top K\mathbf{1}}>0$. Using optimality conditions, $y^\star\circ Kx^\star=-a$, we obtain $\beta=-1$. The automatic heuristic selection of step-sizes is as follows:

$$\sigma=\sqrt{\frac{p}{q}}\frac{1}{\|K\|}\frac{\mathbf{1}^\top K\mathbf{1}}{\mathbf{1}^\top a} \quad \text{and} \quad \tau=\sqrt{\frac{q}{p}}\frac{1}{\|K\|}\frac{\mathbf{1}^\top a}{\mathbf{1}^\top K\mathbf{1}}.$$

Note the invariance by rescaling of a and K.

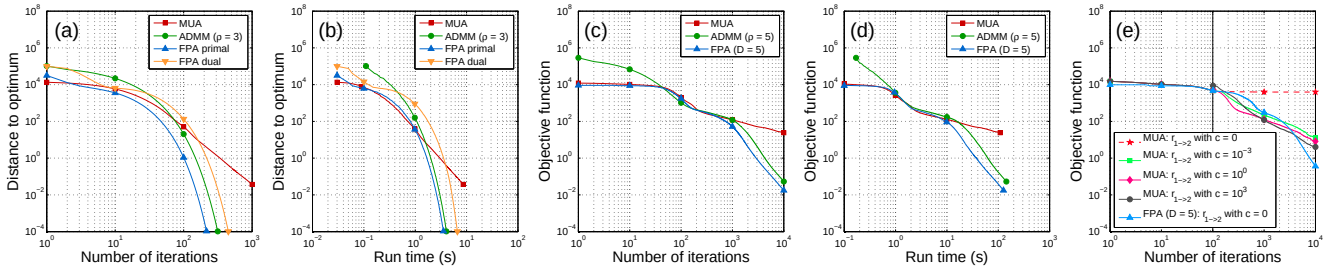

Fig. 1: Experiments on synthetic data. (a-b) ND problem (estimate $\mathbf{H}$ given $\mathbf{W}^\star$). Distance to optimum versus iteration number and run time, respectively. Distance to optimum is the absolute difference between the values of the objective function and optimal point. (c-d) NMF problem. Objective function versus iteration number and run time, respectively. Note that the dual function is not presented due to the non-convexity of the NMF problem. (e) NMF with warm restarts. Objective function at each iteration for various c, a parameter that controls the magnitude of the non-zero entries in one of the matrix factors.

3.3. Extension to NMF

Our alternating first-order primal-dual algorithm (FPA) for NMF can be found in Algorithm 2. Similar optimization approaches were previously reported in [15, 16]. For algorithmic efficiency, we work directly with the matrices, e.g., $a \in \mathbb{R}_+^{n\times m}$ instead of $\mathbb{R}_+^n$. A key algorithmic choice is the number of inner iterations D of the convex method, which we consider in Section 4. The running-time complexity is still $O(rnm)$ for each inner iterations. Note that computing the largest singular value of $\mathbf{H}$ or $\mathbf{W}$ (required for the heuristic selection of step-sizes everytime we switch from one convex problem to the other) is of order $O(r\max\{m,n\})$ and is thus negligible compared to the iteration cost. The stopping criteria is set for maximum number of iterations (access to data).

Algorithm 2: FPA for NMF with the KL loss.

Select $\mathbf{V} \in \mathbb{R}_+^{n\times m}$, $\mathbf{W} \in \mathbb{R}_+^{n\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times m}$, $N, D > 0$;
Set $\bar{\mathbf{W}} = \mathbf{W}_{old} = \mathbf{W}$, $\bar{\mathbf{H}} = \mathbf{H}_{old} = \mathbf{H}$, $\chi = -\mathbf{V} \oslash \mathbf{WH}$;
for N/D *iterations* **do**
  Set $\sigma = \sqrt{\frac{n}{r}}\frac{1}{\|\mathbf{W}\|}\frac{\mathbf{1}^\top\mathbf{W1}}{\mathbf{1}^\top\mathbf{V1}}$ and $\tau = \sqrt{\frac{r}{n}}\frac{1}{\|\mathbf{W}\|}\frac{\mathbf{1}^\top\mathbf{V1}}{\mathbf{1}^\top\mathbf{W1}}$;
  for D *iterations* **do**
    $\chi \leftarrow \chi + \sigma\mathbf{W}\bar{\mathbf{H}}$;
    $\chi \leftarrow \frac{1}{2}\left(\chi - \sqrt{\chi\circ\chi + 4\sigma\mathbf{V}}\right)$;
    $\mathbf{H} \leftarrow \left(\mathbf{H} - \tau\mathbf{W}^\top(\chi + \mathbf{1})\right)_+$;
    $\bar{\mathbf{H}} \leftarrow 2\mathbf{H} - \mathbf{H}_{old}$;
    $\mathbf{H}_{old} \leftarrow \mathbf{H}$;
  end
  Set $\sigma = \sqrt{\frac{m}{r}}\frac{1}{\|\mathbf{H}\|}\frac{\mathbf{1}^\top\mathbf{H1}}{\mathbf{1}^\top\mathbf{V1}}$ and $\tau = \sqrt{\frac{r}{m}}\frac{1}{\|\mathbf{H}\|}\frac{\mathbf{1}^\top\mathbf{V1}}{\mathbf{1}^\top\mathbf{H1}}$;
  for D *iterations* **do**
    $\chi \leftarrow \chi + \sigma\bar{\mathbf{W}}\mathbf{H}$;
    $\chi \leftarrow \frac{1}{2}\left(\chi - \sqrt{\chi\circ\chi + 4\sigma\mathbf{V}}\right)$;
    $\mathbf{W} \leftarrow \left(\mathbf{W} - \tau(\chi + \mathbf{1})\mathbf{H}^\top\right)_+$;
    $\bar{\mathbf{W}} \leftarrow 2\mathbf{W} - \mathbf{W}_{old}$;
    $\mathbf{W}_{old} \leftarrow \mathbf{W}$;
  end
end
return $\mathbf{W}^\star = \mathbf{W}$ *and* $\mathbf{H}^\star = \mathbf{H}$.

3.4. Extension to topic models

Probabilistic latent semantic analysis [17] or latent Dirichlet allocation [18], generative probabilistic models for collections of discrete data, have been extensively used in text analysis. Their formulations relate to ours in Problem (3), where we just need to include an additional constraint: $\mathbf{1}^\top x = 1$. If we modify G to $G(x) = \mathbb{1}\{\mathbf{1}^\top x = 1;\ x \succeq 0\} + 1^\top Kx$, we can use Algorithm 1 to find the latent topics. It is important to mention that herein $\mathbf{prox}_{\tau G}(x)$ does not have a closed solution, but can be efficiently solved with dedicated methods for orthogonal projections on the simplex [19].

4. EXPERIMENTAL RESULTS

The performance of our FPA (Algorithm 2), MUA [3], and ADMM [8] is tested on both synthetic and real world data.

4.1. Synthetic data

Consider $n = 200$, $m = 500$, and $r = 10$. The ground truth matrix factors $\mathbf{W}^\star$ and $\mathbf{H}^\star$ are randomly generated from the magnitude of a normal distribution. The given matrix $\mathbf{V}$ is set as the product of the optimal factors $\mathbf{W}^\star$ and $\mathbf{H}^\star$. The initial factors $\mathbf{W}_0$ and $\mathbf{H}_0$ are randomly generated from an uniform distribution. This setting is used in all synthetic experiments.

4.1.1. ND problem

We estimate $\mathbf{H}$ given $\mathbf{W}^\star$. The number of iterations for the three algorithms is set to $N = 1\,000$. The tuning parameter that controls the convergence rate of ADMM is set to $\rho = 3$ (small values imply larger step sizes, which may result in faster convergence but also instability). Figure 1 (a) illustrates the distance to optimum per iteration. FPA and ADMM reach the optimal point, whereas MUA gets stuck far away from it. Similar behavior can be observed when comparing the distance to optimum versus run time in Figure 1 (b), FPA and ADMM converge significantly faster than MUA.

4.1.2. NMF problem

We solve the non-convex Problem (1) with the three algorithms setting the number of iterations to $N = 10\,000$. The parameter D indicates the number of iterations to solve each ND problem. We set D to 5 iterations. To have a fair comparison between algorithms, for the FPA, the number of iterations means access to data, i.e., we use 5 iterations to solve each problem in (2), and repeat this $2\,000$ times. Figure 1 (c-d) illustrates the objective function versus iteration number and run time, respectively. The MUA and FPA objectives decrease dramatically in only seconds (few iterations), however, MUA presents evident slow tail convergence. Even though ADMM has the poorest initial performance, it reaches an improved local optima compared to MUA. The best local optima and fastest convergence is reported by FPA. Note the significant advantage towards our algorithm, together with the fact that step-sizes (σ and τ) are automatically selected.

4.1.3. NMF with warm restarts

Consider we solve the problem just described (Section 4.1.2) but with $r_1 = 5$ and $N_1 = 100$, and then solve the problem for $r_2 = 10$ and $N_2 = 9\,900$. Having solved the first optimization problem, the computational effort of solving the second one can be reduced if we use the optimal matrix factors of the first problem, $\mathbf{W}_1^\star$ and $\mathbf{H}_1^\star$, as an advanced starting point. Note that $r_2 > r_1$, therefore, we need to include $(r_2 - r_1)n$ and $(r_2 - r_1)m$ entries for each factor, respectively. If we add only zeros, we would be in a saddle-point where none of the algorithms could perform. However, if we set only one factor with zero entries and the other one with non-zero values, FPA could perform. In this situation, MUA cannot perform either because of the absorbing of zeros. Thus we need to add non-zero entries in both factors of MUA. Figure 1 (e) illustrates the warm restarts experiment for FPA and MUA. ADMM is not shown because it has a similar behavior as FPA. Note that we include the parameter c to control the effect of the non-zero entries in one matrix factor. As $c \to 0$, MUA gets stuck in poor local optima until it cannot perform. Then, as $c \to +\infty$, the objective value of MUA increases, reducing the advantage of an advanced starting point. Opposed to MUA, FPA uses the information gained from the first problem.

4.2. Real world data

4.2.1. MIT-CBCL Face Database #1 [20]

The CBCL face images database is composed of $m = 2\,429$ images of size $n = 361$ pixels. We solve the NMF problem with $r = 10$. The number of iterations is set to $N = 10\,000$, and $D = 20$. Figure 2 (a-b) show that all algorithms converge to the same local optima. However, the fastest algorithm is FPA with a run time of 20.9 min, then ADMM with 21.2 min, and finally MUA with 22.4 min.

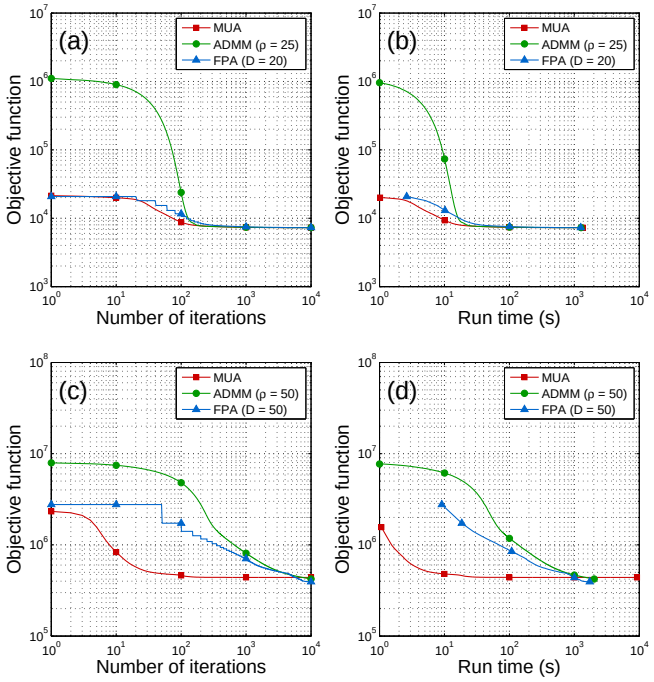

Fig. 2: NMF experiments on real world data. The objective function versus iteration number (left) and run time (right) is computed on (a-b) the MIT-CBCL Face Database #1 [20], and (c-d) "My Heart (Will Always Lead Me Back to You)" [5].

4.2.2. "My Heart (Will Always Lead Me Back to You)" [5]

The spectrogram of a 108-second-long music excerpt from "My Heart (Will Always Lead Me Back to You)" by Louis Armstrong & His Hot Five consists of $m = 9\,312$ frames and $n = 129$ frequency bins. We set r to 20 and solve NMF with all algorithms using $N = 10\,000$ and $D = 50$. Figure 2 (c-d) illustrates the results. Initially MUA obtains a low objective value, but as previously discussed, the algorithm shows slow tail convergence and gets stuck in a worse local optima than the other ones. ADMM has a slow initial performance, but then overpasses MUA. When doing alternating optimization, the local steps (early stopping ND problems) may either be exact or inexact [21]. Note that the inexact local steps of FPA end up being most efficient. FPA is both faster than MUA and ADMM, and leads to an improved local optima.

5. CONCLUSION

We have presented an alternating projected gradient descent technique for NMF that minimizes the KL divergence loss; this approach solves convex ND problems with the FPA. Our approach demonstrated faster convergence than the MUA [3] and ADMM [8]. An extension to latent Dirichlet allocation and probabilistic latent semantic indexing can be easily implemented using our proposed method, thus allowing to go beyond the potential slowness of the EM algorithm.

6. REFERENCES

[1] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error," *Environmetrics*, vol. 5, pp. 111–126, 1994.

[2] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[3] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, 2000.

[4] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Non-negative matrix factorization framework for face recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, pp. 495–511, 2005.

[5] 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, pp. 793–830, 2009.

[6] C. J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Computation*, vol. 19, pp. 2756–2779, 2007.

[7] D. Kim, S. Sra, and I. S. Dhillon, "Fast projection-based methods for the least squares nonnegative matrix approximation problem," *Statistical Analysis and Data Mining*, vol. 1, pp. 38–51, 2008.

[8] D. L. Sun and C. Févotte, "Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence," in *Proceedings of the 39th International Conference on Acoustic, Speech and Signal Processing*, 2014.

[9] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, pp. 120–145, 2011.

[10] C. Févotte and A. T. Cemgil, "Nonnegative matrix factorizations as probabilistic inference in composite models," in *Proceedings of the 17th European Signal Processing Conference*, 2009.

[11] A. Lefèvre, F. Bach, and C. Févotte, "Online algorithms for nonnegative matrix factorization with the Itakura-Saito divergence," in *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2011.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[13] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, 1997.

[14] F. Yanez and F. Bach, "Primal-Dual Algorithms for Non-negative Matrix Factorization with the Kullback-Leibler Divergence," Technical report, HAL-01079229, October 2014.

[15] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proceedings of the 8th International Conference on Data Mining*, 2008.

[16] H.-F. Yu, C.-J. Hsieh, S. Si, and I. Dhillon, "Scalable coordinate descent approaches to parallel matrix factorization for recommender systems," in *Proceedings of the 12th International Conference on Data Mining*, 2012.

[17] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, 1999.

[18] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[19] N. Maculan and G. Galdino de Paula, "A linear-time median-finding algorithm for projecting a vector on the simplex of $\mathbb{R}^n$," *Operations research letters*, vol. 8, pp. 219–222, 1989.

[20] K. K. Sung, *Learning and Example Selection for Object and Pattern Recognition*, Ph.D. thesis, Massachusetts Institute of Technology, 1996.

[21] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, pp. 1126–1153, 2013.