NON-NEGATIVE MATRIX FACTORIZATION BASED ON ALTERNATING NON-NEGATIVITY CONSTRAINED LEAST SQUARES AND ACTIVE SET METHOD*

HYUNSOO KIM AND HAESUN PARK†

Abstract.
The non-negative matrix factorization (NMF) determines a lower rank approximation of a matrix $A \in \mathbb{R}^{m\times n} \approx WH$ where an interger $k \ll \min(m,n)$ is given and nonnegativity is imposed on all components of the factors $W \in \mathbb{R}^{m\times k}$ and $H \in \mathbb{R}^{k\times n}$. The NMF has attracted much attention for over a decade and has been successfully applied to numerous data analysis problems. In applications where the components of the data are necessarily non-negative such as chemical concentrations in experimental results or pixels in digital images, the NMF provides a more relevant interpretation of the results since it gives non-subtractive combinations of non-negative basis vectors. In this paper, we introduce an algorithm for the NMF based on alternating non-negativity constrained least squares (NMF/ANLS) and the active set based fast algorithm for non-negativity constrained least squares with multiple right hand side vectors, and discuss its convergence properties and a rigorous convergence criterion based on the Karush-Kuhn-Tucker (KKT) conditions. In addition, we also describe algorithms for sparse NMFs and regularized NMF. We show how we impose a sparsity constraint on one of the factors by L_1-norm minimization and discuss its convergence properties. Our algorithms are compared to other commonly used NMF algorithms in the literature on several test data sets in terms of their convergence behavior.

Key words. Non-negative Matrix Factorization, Lower Rank Approximation, Two Block Coordinate Descent Method, Karush-Kuhn-Tucker (KKT) Conditions, Non-negativity constrained Least Squares, Active Set Method

AMS subject classifications. 15A23

1. Introduction.

Given a non-negative matrix $A \in \mathbb{R}^{m\times n}$ and a desired rank $k \ll \min(m,n)$, the non-negative matrix factorization (NMF) searches for non-negative factors W and H that give a lower rank approximation of A as

$$A \approx WH \;\; s.t. \;\; W, H \geq 0, \tag{1.1}$$

where $W, H \geq 0$ means that all elements of W and H are non-negative. The problem in Eqn. (1.1) is commonly reformulated as the following optimization problem:

$$\min_{W,H} f(W,H) \equiv \frac{1}{2}\|A - WH\|_F^2, \;\; s.t. \;\; W, H \geq 0, \tag{1.2}$$

where $W \in \mathbb{R}^{m\times k}$ is a basis matrix and $H \in \mathbb{R}^{k\times n}$ is a coefficient matrix. In many data analysis problems, typically each column of A corresponds to a data point in the m-dimensional space.

The non-negative matrix factorization (NMF) may give a simple interpretation due to non-subtractive combinations of non-negative basis vectors and has recently received much attention. Applications of the NMF are numerous including image processing [21], text data mining [31], subsystem identification [19], cancer class discovery [4, 8, 18], etc. It has been over a decade since the NMF was first proposed by Paatero and Tapper [27] (in fact, as *positive* matrix

*This material is based upon work supported in part by the National Science Foundation Grants CCF-0621889 and CCF-0732318. Any opinions, fi ndings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

†College of Computing, Georgia Institute of Technology, 266 Ferst Drive, Atlanta, GA 30332, USA (hskim@cc.gatech.edu, hpark@cc.gatech.edu).

factorization) in 1994. Various types of NMF techniques have been proposed in the literature [5, 13, 25, 32, 34], which include the popular Lee and Seung's iterative multiplicative update algorithms [21, 22], gradient descent methods [24], and alternating least squares [1]. Paatero and Tapper [27] originally proposed an algorithm for the NMF using a constrained alternating least squares algorithm to solve Eqn. (1.2). Unfortunately, this approach has not obtained wide attention especially after Lee-Seung's multiplicative update algorithm was proposed [1, 24]. The main difficulty was extremely slow speed caused by a vast amount of hidden redundant computation related to satisfying the non-negativity constraints *exactly*. One may try to deal with the non-negativity constraints in an approximate sense for faster algorithm. However, we will show that it is important to satisfy the constraints exactly for the overall convergence of the algorithm and that this property provides very practical and faster algorithm as well. In addition, faster algorithms that exactly satisfy the the non-negativity constraints in the least squares with multiple right hand sides already exist [27, 36], which we will discuss and utilize in our proposed NMF algorithms.

In this paper, we provide a framework of the two block coordinate descent method for the NMF. This framework provides a convenient way to explain and compare most of the existing commonly used NMF algorithms and to discuss their convergence properties. We then introduce an NMF algorithm which is based on alternating non-negativity constrained least squares (NMF/ANLS) and the active set method. Although many existing NMF algorithms produce the factors which are often sparse, the formulation of the NMF shown in Eqn. (1.2) does not guarantee the sparsity in the factors. We introduce an NMF formulation and algorithm that imposes sparsity constraint on one of the factors by L_1-norm minimization and discuss its convergence properties. The L_1-norm minimization term is formulated in such a way that the proposed sparse NMF algorithm also fits into the framework of the two block coordinate descent method and accordingly its convergence properties become easy to understand.

The rest of this paper is organized as follows. We present the framework of the two block coordinate descent method and provide a brief overview of various existing NMF algorithms in Section 2. In Section 3, we introduce our NMF algorithm based on alternating non-negativity constrained least squares and fast active set method, called NMF/ANLS and discuss its convergence properties. In Section 4, we describe some variations of the NMF/ANLS algorithm, which include the method designed to impose sparsity on one of the factors through the addition of an L_1-norm minimization term in the problem formulation. Our algorithms are compared to other commonly used NMF algorithms in the literature on several test data sets in Section 6. Finally, summary and discussion are given in Section 7.

2. A Two Block Coordinate Descent Framework for NMF Algorithms and Convergence Properties. In most of the currently existing algorithms for the NMF, the basic framework is to reformulate the non-convex minimization problem shown in Eqn. (1.2) as a two-block coordinate descent problem [2]. Given a non-negative matrix $A \in \mathbb{R}^{m \times n}$ and an integer $k < \min(m, n)$, one of the factors, say $H \in \mathbb{R}^{k \times n}$, is initialized with non-negative values. Then, one may iterate the following alternating non-negativity constrained least squares (ANLS) until a convergence criterion is satisfied:

$$\min_{W \geq 0} \|H^T W^T - A^T\|_F^2, \tag{2.1}$$

where H is fixed, and

$$\min_{H \geq 0} \|WH - A\|_F^2, \tag{2.2}$$

where W is fixed. Alternatively, after initializing W, one may iterate Eqn. (2.2) and then Eqn. (2.1) until a convergence criterion is satisfied. Each subproblem shown in Eqns. (2.1)-(2.2) can be solved by projected quasi-Newton optimization [37, 15], projected gradient descent optimization [24], or non-negativity constrained least squares [27, 16, 28].

Note that the original NMF problem of Eqn. (1.2) is non-convex and most non-convex optimization algorithms guarantee only the stationarity of limit points. Since the problem formulation is symmetric with respect to initialization of the factors H or W, for simplicity of discussion, we will assume that the iteration is performed with the initialization of the factor H. Then the above iteration can be expressed as follows:

- Initialize H with a non-negative matrix $H^{(0)}$; $t \leftarrow 0$
- Repeat until a stopping criterion is satisfied
  - $W^{(t+1)} = \arg\min_W f(W, H^{(t)}) \;\; s.t. \;\; W \geq 0$
  - $H^{(t+1)} = \arg\min_H f(W^{(t+1)}, H) \;\; s.t. \;\; H \geq 0$
  - $t \leftarrow t + 1$

According to the Karush-Kuhn-Tucker (KKT) optimality conditions, (W, H) is a stationary point of Eqn. (1.2) if and only if

$$\begin{array}{ll} W \geq 0, & H \geq 0, \\ \nabla_W f(W,H) = WHH^T - AH^T \geq 0, & \nabla_H f(W,H) = W^TWH - W^TA \geq 0, \\ W. * \nabla_W f(W,H) = 0, & H. * \nabla_H f(W,H) = 0, \end{array} \tag{2.3}$$

where $.*$ denotes component-wise multiplication [11].

For Eqn. (1.2), when the block coordinate descent algorithm is applied, then no matter how many sub-blocks into which the problem is partitioned, if the subproblems have unique solutions, then the limit point of the sequence is a stationary point [2]. For two block problems, Grippo and Siandrone [12] presented a stronger result. The result does not require uniqueness of the solution in each subproblem, which is that any limit point of the sequence generated based on the optimal solutions of each of the two sub-blocks is a stationary point. Since the subproblems Eqns. (2.1) and (2.2) are convex but not strongly convex, they do not necessarily have unique solutions. However, according to the two block result, it is still the case that any limit point will be a stationary point. We emphasize that for convergence to a stationary point, it is important to find an optimal solution for each subproblem.

In one of the most commonly utilized NMF algorithms due to Lee and Seung [21, 22], the NMF is computed using the following norm-based multiplicative update rules (NMF/NUR) of W and H, which is a variation of the gradient descent method:

$$W_{iq} \leftarrow W_{iq} \frac{(AH^T)_{iq}}{(W(HH^T))_{iq}}, \tag{2.4}$$

for $1 \leq i \leq m$ and $1 \leq q \leq k$,

$$H_{qj} \leftarrow H_{qj} \frac{(W^TA)_{qj}}{((W^TW)H)_{qj}}, \tag{2.5}$$

for $1 \leq q \leq k$ and $1 \leq j \leq n$. Each iteration may in fact break down since the denominators in both Eqns. (2.4) and (2.5) can be zeros. Accordingly, in practical algorithms, a small positive

number is added to each denominator to prevent division by zero. There are several variations of NMF/NUR [8, 30, 6].

Lee and Seung also designed an NMF algorithm using the divergence-based multiplicative update rules (NMF/DUR) [22] to minimize the divergence:

$$D(A||WH) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(A_{ij}\ln\frac{A_{ij}}{(WH)_{ij}} - A_{ij} + (WH)_{ij}\right), \quad s.t.\ W, H \geq 0. \tag{2.6}$$

Strictly speaking, this formulation is not a bound constrained problem, which requires the objective function to be well-defined at any point of the bounded region, since the log function is not well-defined if $A_{ij} = 0$ or $(WH)_{ij}$ = 0 [24]. The divergence is also nonincreasing during iterations. Gonzales and Zhang [11] claimed that these nonincreasing properties of multiplicative update rules may not imply the convergence to a stationary point within a realistic amount of run time for problems of meaningful sizes. Lin [24] devised an NMF algorithm based on projected gradient methods. However, it is known that gradient descent methods may suffer from slow convergence due to a possible zigzag phenomenon.

Berry *et al.* [1] proposed an NMF algorithm based on alternating least squares (NMF/ALS). This algorithm computes the solutions to the subproblems Eqn. (2.1) and (2.2) as an *unconstrained* least squares problems with multiple right hand sides and sets negative values in the solutions W and H to zeros during iterations to enforce non-negativity. Although this may give a faster algorithm for approximating each subproblem, the convergence of the overall algorithm is difficult to analyze since the subproblems are formulated as constrained least squares problems but the solutions are not those of the constrained least squares.

Zdunek and Cichocki [37] developed a quasi-Newton optimization approach with projection. In this algorithm, the negative values of W and H are replaced with a very small positive value. Again, setting negative values to zeros or small positive values for imposing non-negativity makes theoretical analysis of the convergence of the algorithm difficult [3]. The projection step can increase the objective function value and may lead to non-monotonic changes in the objective function value resulting in inaccurate approximations.

A more detailed review of NMF algorithms can be found in [1].

3. NMF based on Alternating Non-negativity constrained Least Squares (NMF/ANLS) and the Active Set Method. In this section, we describe our NMF algorithm based on alternating non-negativity constrained least squares (NMF/ANLS) that satisfies the non-negativity constraints in each of the subproblems in Eqn. (2.1) and (2.2) exactly and therefore has the convergence property that every limit point is a stationary point.

The structures of the two non-negativity constrained least squares (NLS) problems with multiple right hand sides shown in Eqns. (2.1) and (2.2) are essentially the same, therefore we will concentrate on a general form of the NLS with multiple right hand sides

$$\min_{G\geq 0} \|BG - Y\|_F^2 \tag{3.1}$$

where $B \in \mathbb{R}^{p\times q}$ and $Y \in \mathbb{R}^{p\times l}$ are given, which can be decoupled into l independent NLS problems each with single right hand side as

$$\min_{G\geq 0} \|BG - Y\|_F^2 \rightarrow \min_{\mathbf{g}_1\geq 0} \|B\mathbf{g}_1 - \mathbf{y}_1\|_2^2, \quad \ldots \quad , \min_{\mathbf{g}_l\geq 0} \|B\mathbf{g}_l - \mathbf{y}_l\|_2^2, \tag{3.2}$$

where $G = [\mathbf{g}_1, \ldots, \mathbf{g}_l] \in \mathbb{R}^{q \times l}$ and $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_l] \in \mathbb{R}^{p \times l}$. This objective function is not strictly convex so that it does not ensure a unique solution unless B is full column rank. In the context of the NMF computation, we implicitly assume that the fixed matrices H^T and W involved in Eqns. (2.1) and (2.2) are of full column rank since they are interpreted as basis matrices for A^T and A, respectively. Each of the NLS problems with single right hand side vector

$$\min_{\mathbf{g}_j \geq 0} \|B\mathbf{g}_j - \mathbf{y}_j\|_2, \tag{3.3}$$

for $1 \leq j \leq l$, can be solved by using the active set method of Lawson and Hanson [20], which is implemented in MATLAB [26] as function *lsqnonneg*. The algorithm is summarized in Algorithm NLS. The following theorem states the necessary and sufficient conditions for a vector g to be a solution for the problem NLS.

THEOREM 1. *(Kuhn-Thcker Conditions for Problem NLS) A vector $g \in \mathbb{R}^{n \times 1}$ is a solution for problem NLS defined as*

$$min \|Bg - b\| \;\; subject \;\; to \;\; g \geq 0 \tag{3.4}$$

if and only if there exists a vector $r \in \mathbb{R}^{m \times 1}$ and a partitioning of the integers 1 through m into subsets $\mathcal{E}$ and $\mathcal{S}$ such that with $r = B^T(Bg - y)$

$$g_i = 0 \;\text{ for }\; i \in \mathcal{E}, \quad g_i > 0 \;\text{ for }\; i \in \mathcal{S} \tag{3.5}$$

$$r_i \geq 0 \;\text{ for }\; i \in \mathcal{E}, \quad r_i = 0 \;\text{ for }\; i \in \mathcal{S} \tag{3.6}$$

On termination of Algorithm NLS, the solution vector g satisfies

$$g_i > 0, \;\; i \in \mathcal{S} \quad \text{and} \quad g_i = 0, \;\; i \in \mathcal{E} \tag{3.7}$$

and is a solution vector for the *unconstrained* least squares problem

$$\min_x \|B_{\mathcal{S}} g - y\|_2. \tag{3.8}$$

The dual vector $w = -r = B^T(y - Bg)$ satisfies

$$w_i = 0 \;\; i \in \mathcal{S} \quad \text{and} \quad w_i \leq 0 \;\; j \in \mathcal{E} \tag{3.9}$$

To enhance the computational speed in solving Eqn. (3.1) based on Algorithm NLS, we utilize the fast algorithms by Bro and de Jong [3] and Van Benthem and Keenan [36]. Bro and de Jong [3] made a substantial speed improvement for solving Eqn. (3.1) which has multiple right hand side vectors over a naive application of Algorithm NLS which is for a single right hand side problem, by precomputing cross-product terms that appear in the normal equations of the unconstrained least squares problems. Van Benthem and Keenan [36] devised an algorithm that further improves the performance of NLS for multivariate data by initializing the active set $\mathcal{E}$ based on the result from the unconstrained least squares solution and reorganizing the calculations to take advantage of the combinatorial nature of the active set based solution methods for the NLS with *multiple* right hand sides.

To illustrate the situation in a simpler context, let us for now assume that there is no non-negativity constraints in the least squares (LS) problems shown in Eqn. (3.2) and (3.3). Then,

Algorithm 1 NLS: This algorithm computes the solution for the problem $\min_{g \geq 0} \|Bg - y\|_2$ by Active Set method, where $B \in \mathbf{R}^{m \times n}$ and $y \in \mathbf{R}^{m \times 1}$ are given.

Initialization:
$g := 0$
$\mathcal{E} := \{1, 2, \cdots, n\}$ % Initially all indices belong to Active set since $g := 0$
$\mathcal{S} := \phi$ % Initially Passive set is empty
$w := B^T(b - Bg)$.

Do While ($\mathcal{E} \neq \phi$ and $\exists\ j \in \mathcal{E}$ such that $w_j > 0$)

1. Find an index $t \in \mathcal{E}$ such that $w_t = max\{w_j : j \in \mathcal{E}\}$ % t is the column index of B that can potentially reduce the objective function value by maximum when brought into the Passive set.
2. Move the index t from set $\mathcal{E}$ to set $\mathcal{S}$.
3. Let $B_{\mathcal{S}}$ denote the $m \times n$ matrix defined by
   Column j of $B_{\mathcal{S}} := \begin{cases} \text{column } j \text{ of } B & \text{if } j \in \mathcal{S} \\ 0 & \text{if } j \in \mathcal{E} \end{cases}$
   Solve $\min_z \|B_{\mathcal{S}} z - b\|_2$. (% Only the components z_j, $j \in \mathcal{S}$, are determined by this problem.)
   $z_j := 0$ for $j \in \mathcal{E}$
4. **Do While** ($z_j \leq 0$ for any $j \in \mathcal{S}$)
   (a) Find an index $q \in \mathcal{S}$ such that $g_q/(g_q - z_q) = min\{g_j/(g_j - z_j) :\ z_j \leq 0, j \in \mathcal{S}\}$
   (b) $\alpha := g_q/(g_q - z_q)$.
   (c) $g := g + \alpha(z - g)$.
   (d) Move from set $\mathcal{S}$ to set $\mathcal{E}$ all indices $j \in \mathcal{S}$ for which $g_j = 0$.
   (e) Define $B_{\mathcal{S}}$ as in Step 3 and
   Solve $\min_z \|B_{\mathcal{S}} z - b\|_2$.
5. **End While** (% $z_j > 0$ for all $j \in \mathcal{S}$)
6. g:=z
7. $w := B^T(b - Bg)$.

End While (% $\mathcal{E}$ is empty (All indices are passive) or $w_j \leq 0$ for all $j \in \mathcal{E}$ (Objective function value cannot be reduced any more))

since an optimal solution $\mathbf{g}_j^*$ for $\min_{\mathbf{g}_j} \|B\mathbf{g}_j - \mathbf{y}_j\|_2$ is $B^\dagger \mathbf{y}_j$ for $j = 1, \cdots, l$, the pseudo-inverse $B^\dagger$ of B [9] needs to be computed *only once* (in fact, we do not recommend forming the pseudo-inverse explicitly and it is used here only for explanation). Clearly, it would be extremely inefficient if we treat each subproblem independently and process the matrix B each time. In the case of the NLS with multiple right hand side vectors, the scenario is not this simple since the active set $\mathcal{E}$ may differ in each iteration and for each right hand side vector, and a solution is obtained based on a subset of columns of the matrix B that corresponds to the passive set in each iteration as shown in Step 3 of Algorithm NLS. However, much of the computation which is potentially redundant in each iteration can be identified and precomputed only once. For example, if the matrix B has full column rank, then by precomputing B^TB and B^TY only once and extracting the necessary components from these for each passive set, one can obtain the solution efficiently

by extracting the normal equations for each passive set avoiding redundant computations [3]. In addition, for the multiple right hand side case, the computations can be rearranged to be column parallel, i.e., the passive set columns in each step of the active set iteration for all right hand side vectors are identified collectively at once. Thus, larger sets of common passive sets can be found and more redundant computations can be avoided. More detailed explanations of this algorithm can be found in [36].

As we stated earlier, with the above mentioned solution method NMF/ANLS, which satisfies the non-negativity constraint exactly, any limit point will be a stationary point [2, 12]. Lin [24] also discussed the convergence properties of alternating non-negativity constrained least squares and showed that any limit point of the sequence (W,H) generated by alternating non-negativity constrained least squares is a stationary point of Eqn. (1.2) when the objective function is convex, and not necessarily strictly convex. The NMF is clearly not unique since there exist nonsingular matrices $X \in \mathbb{R}^{k\times k}$ including scaling and permutation matrices satisfying $WX \geq 0$ and $X^{-1}H \geq 0$ and these factors give $\|A - WH\|_F = \|A - WXX^{-1}H\|_F$. To provide a fair comparison among the computed factors based on various algorithms in the presence of this non-uniqueness, after convergence, the columns of the basis matrix W are often normalized to unit L_2-norm and the rows of H are adjusted so that the objective function value is not changed. However, we would like to note that normalizing the computed factors after each iteration makes the convergence results of the two block coordinate descent method not applicable since the normalization alters the objective function of the subproblems expressed in Eqns. (2.1) and (2.2).

4. Algorithms for Sparse NMF based on Alternating Non-negativity constrained Least Squares. One of the interesting properties of the NMF is that it often generates sparse factors that allow us to discover parts-based basis vectors. Although the results presented in [21] show that the computed NMF generated parts-based basis vectors, the generation of a parts-based basis by the NMF depends on the data and the algorithm [14, 23]. Several approaches [7, 14, 29, 30] have been proposed to explicitly control the degree of sparseness in the factors of the NMF. In this section, we propose algorithms for the sparse NMF that follows the framework of the two block coordinate descent methods and therefore guarantees that every limit point is a stationary point. In particular, we propose an L_1-norm based constrained NMF formulation to control the sparsity on one of the factors.

4.1. Constrained NMF based on Alternating Non-negativity constrained Least Squares (CNMF/ANLS). Pauca *et al.* [30] proposed the following constrained NMF (CNMF) formulation for the purpose of obtaining a sparse NMF,

$$\min_{W,H} \frac{1}{2}\{\|A - WH\|_F^2 + \alpha\|W\|_F^2 + \beta\|H\|_F^2\}, \quad s.t. \ W, H \geq 0, \tag{4.1}$$

where $\alpha \geq 0$ and $\beta \geq 0$ are the parameters to be chosen and are supposed to control the sparsity of W and H, respectively. An algorithm was developed based on multiplicative update rules for the CNMF formulation.

We now show how the formulation in Eqn. (4.1) can be recast into the ANLS framework and developed into an algorithm CNMF/ANLS for which every limit point is a stationary point. The algorithm CNMF/ANLS begins with the initialization of H with non-negative values. Then, the

following ANLS can be iterated:

$$\min_{W\geq 0}\left\|\begin{pmatrix} H^T \\ \sqrt{\alpha}I_k \end{pmatrix} W^T - \begin{pmatrix} A^T \\ 0_{k\times m} \end{pmatrix}\right\|_F^2, \tag{4.2}$$

where I_k is a $k \times k$ identity matrix and $0_{k\times m}$ is a zero matrix of size $k \times m$, and

$$\min_{H\geq 0}\left\|\begin{pmatrix} W \\ \sqrt{\beta}I_k \end{pmatrix} H - \begin{pmatrix} A \\ 0_{k\times n} \end{pmatrix}\right\|_F^2, \tag{4.3}$$

where $0_{k\times n}$ is a zero matrix of size $k \times n$. Similarly, one may initialize $W \in \mathbb{R}^{m\times k}$ and alternate the above in the order of solving Eqn. (4.3) and Eqn. (4.2). Eqn. (4.1) is differentiable in the feasible region and Eqns. (4.2)-(4.3) are strictly convex. Then again according to convergence analysis for block coordinate descent methods [2], any limit point of our CNMF/ANLS algorithm will be a stationary point.

4.2. Sparse NMF with L_1-norm Constraint. The idea of imposing L_1-norm based constraints for the purpose of achieving sparsity in the solution has been successfully utilized in a variety of problems [35]. For the NMF, we propose the following formulation of the NMF that imposes sparsity on the right side factor H (SNMF/R) [16, 18],

$$\min_{W,H}\frac{1}{2}\{\|A - WH\|_F^2 + \eta\|W\|_F^2 + \beta\sum_{j=1}^{n}\|\mathbf{h}_j\|_1^2\},\ \ s.t.\ W, H \geq 0, \tag{4.4}$$

where $\mathbf{h}_j$ is the j-th column vector of H, the parameter $\eta \geq 0$ suppress the growth of W, and the parameter $\beta \geq 0$ balances the trade-off between the accuracy of the approximation and the sparseness of H. Note that due to the non-negativity constraint on H, the last term in Eqn. (4.4) becomes equivalent to $\beta\sum_{j=1}^{n}(\sum_{i=1}^{k} h_{ij})^2$ and accordingly Eqn. (4.4) is differentiable in the feasible domain. The SNMF/R algorithm begins with the initialization of W with non-negative values. Then, it iterates the following ANLS until a convergence criterion is satisfied:

$$\min_{H\geq 0}\left\|\begin{pmatrix} W \\ \sqrt{\beta}\mathbf{e}_{1\times k} \end{pmatrix} H - \begin{pmatrix} A \\ \mathbf{0}_{1\times n} \end{pmatrix}\right\|_F^2, \tag{4.5}$$

where $\mathbf{e}_{1\times k} \in \mathbb{R}^{1\times k}$ is a row vector with all components equal to one and $\mathbf{0}_{1\times n} \in \mathbb{R}^{1\times n}$ is a zero vector, and

$$\min_{W\geq 0}\left\|\begin{pmatrix} H^T \\ \sqrt{\eta}I_k \end{pmatrix} W^T - \begin{pmatrix} A^T \\ 0_{k\times m} \end{pmatrix}\right\|_F^2, \tag{4.6}$$

where $0_{k\times m}$ is a zero matrix of size $k \times m$. Eqn. (4.5) minimizes the L_1-norm of each column of $H \in \mathbb{R}^{k\times n}$.

Similarly, sparsity in the NMF can be imposed on the left side factor W (SNMF/L) through the following formulation:

$$\min_{W,H}\frac{1}{2}\{\|A - WH\|_F^2 + \zeta\|H\|_F^2 + \alpha\sum_{i=1}^{m}\|\mathbf{w}_i\|_1^2\},\ \ s.t.\ W, H \geq 0, \tag{4.7}$$

where $\mathbf{w}_i^T$ is the i-th row vector of W, $\zeta \geq 0$ is a parameter to suppress $\|H\|_F^2$, and $\alpha \geq 0$ is a parameter to balance the trade-off between accuracy of approximation and sparseness of W. The corresponding algorithm SNMF/L begins with an initialization of the non-negative matrix H. Then, it iterates the following ANLS until a convergence criterion is satisfied:

$$\min_W \left\| \begin{pmatrix} H^T \\ \sqrt{\alpha}\mathbf{e}_{1\times k} \end{pmatrix} W^T - \begin{pmatrix} A^T \\ \mathbf{0}_{1\times m} \end{pmatrix} \right\|_F^2, \quad s.t. \ W \geq 0, \tag{4.8}$$

where $\mathbf{e}_{1\times k} \in \mathbb{R}^{1\times k}$ is a row vector whose elements are all one and $\mathbf{0}_{1\times m} \in \mathbb{R}^{1\times m}$ is a zero vector, and

$$\min_H \left\| \begin{pmatrix} W \\ \sqrt{\zeta} I_k \end{pmatrix} H - \begin{pmatrix} A \\ 0_{k\times n} \end{pmatrix} \right\|_F^2, \quad s.t. \ H \geq 0, \tag{4.9}$$

where I_k is a $k \times k$ identity matrix and $0_{k\times n}$ is a zero matrix of size $k \times n$. Note that Eqn. (4.8) can be rewritten as

$$\min_W \|H^T W^T - A^T\|_2^2 + \alpha \sum_{i=1}^{m} \left(\sum_{q=1}^{k} W^T(q,i) \right)^2 \quad s.t. \ W \geq 0, \tag{4.10}$$

and since all elements in W are non-negative, Eqn. (4.10) in turn becomes the following by the definition of the L_1-norm of a vector:

$$\min_{W\geq 0} \{ \|H^T W^T - A^T\|_2^2 + \alpha \sum_{i=1}^{m} \|\mathbf{w}_i\|_1^2 \}, \tag{4.11}$$

which involves the L_1-norm minimization of each row of W.

An advantage of the above formulation and algorithms is that they follow the framework of the two block coordinate descent method and therefore guarantee convergence of limit points to a stationary point. Imposing additional sparsity constraints on W or H may provide sparser factors and a simpler interpretation. However, imposing sparsity in the factors does not necessarily improve the solution or interpretation. Indeed, as the sparse constraints become stronger, the magnitude of perturbations to the basic NMF solution may become larger and the degree of simplification becomes higher.

5. Regularized NMF based on Alternating Non-negativity constrained Least Squares (RNMF/ANLS). As shown in Section 2, in the algorithm NMF/ANLS, one of the factors W and H is initialized and the iterations are repeated fixing one of the factors. Let us assume that H is initialized. In the NMF, the columns of the computed factor W are interpreted as basis vectors, therefore, implicitly assumed to be of full rank and, in fact, many of the NMF algorithms are designed assuming that the fixed matrices H^T and W involved in the subproblems are of full rank. We propose the following regularized version of the NMF/ANLS, which we call RNMF/ANLS, where the terms $\sqrt{\alpha}I$ and $\sqrt{\beta}I$ with very small parameters $\alpha > 0$ and $\beta > 0$ are attached to the fixed matrices for the purpose of numerical stability. In RNMF/ANLS, after the matrix H is initialized the following steps are iterated:

solve

$$\min_{W\geq 0} \left\| \begin{pmatrix} H^T \\ \sqrt{\alpha} I_k \end{pmatrix} W^T - \begin{pmatrix} A^T \\ 0_{k\times m} \end{pmatrix} \right\|_F^2, \tag{5.1}$$

where I_k is a $k \times k$ identity matrix and $0_{k\times m}$ is a zero matrix of size $k \times m$, and solve

$$\min_{H\geq 0} \left\| \begin{pmatrix} W \\ \sqrt{\beta} I_k \end{pmatrix} H - \begin{pmatrix} A \\ 0_{k\times n} \end{pmatrix} \right\|_F^2, \tag{5.2}$$

where $0_{k\times n}$ is a zero matrix of size $k \times n$. Similarly, one may initialize $W \in \mathbb{R}^{m\times k}$ and alternate the above in the order of solving Eqn. (5.2) and then Eqn. (5.1).

The above RNMF/ANLS is one way to formulate a two block coordinate descent method for the objective function

$$\min_{W,H} \frac{1}{2}\{\|A - WH\|_F^2 + \alpha\|W\|_F^2 + \beta\|H\|_F^2\}, \quad s.t. \ W, H \geq 0, \tag{5.3}$$

where $\alpha \geq$ and $\beta \geq$ are very small regularization parameters. Note that the objective function Eqn. (5.3) and ANLS iterations Eqns. (5.1) and (5.2) are identical to the CNMF formulation and our proposed CNMF/ANLS algorithm presented in Section 4.1. However, the purpose of the CNMF [30] was to obtain a sparser NMF and the role of the parameters α and β was supposed to control the sparsity of W and H. On the other hand, the purpose of the RNMF/ANLS is to impose strong convexity on the subproblems of NMF/ANLS. The role of the parameters α and β with very small values is to impose full rank on the matrices on the left hand of solution matrices in the NLS subproblems. Consequently, we can guarantee that the symmetric square matrix appearing in the normal equations for solving least squares subproblems in the fast NLS algorithm [36] is symmetric positive definite with any passive set of columns, so that the solution can be computed via the Cholesky factorization.

6. Numerical Experiments and Discussion. In this section, we present several numerical experimental results to illustrate the behavior of our proposed algorithms and compare them to two of the most commonly used algorithms, NMF/NUR [21, 22] and NMF/ALS [1] in the literature. We implemented all algorithms in MATLAB 6.5 [26] on a P3 600MHz machine with 512MB memory.

6.1. Data Sets in Experiments. We have used four data sets for our empirical tests, of which two are from microarray analysis and are presented in [8, 16, 18] and the others are artificially generated. All data sets contain only non-negative entries.

I. Data Set ALLAML: The leukemia gene expression data set ALLAML [10] contains acute lymphoblastic leukemia (ALL) that has B and T cell subtypes, and acute myelogenous leukemia (AML) that occurs more commonly in adults than in children. This gene expression data set consists of 38 bone marrow samples (19 ALL-B, 8 ALL-T, and 11 AML) with 5,000 genes forming a data matrix $A \in \mathbb{R}^{5,000\times 38}$. The gene expression values were in the range between 20 and 61,225, where a lower cutoff threshold value of 20 was used to eliminate noisy fluctuations.

II. Data Set CNS: The central nervous system tumors data set CNS [33], is composed of four categories of CNS tumors with 5,597 genes. It consists of 34 samples representing four distinct

TABLE 6.1

*Performance comparison among NMF/NUR [22], NMF/ALS [1], and NMF/ANLS on the leukemia ALLAML data set with $k = 3$. We present the percentages of zero elements in W and H, relative approximation error, the number of iterations, and computing time. *For NMF/NUR, the computed W and H factors were not sparse, so the percentages of the number of the non-negative elements that are smaller than 10^{-8} in W and H are shown instead.*

Algorithms	NMF/NUR	NMF/ALS	NMF/ANLS
#($W = 0$) (%)	2.71%*	2.83%	2.71%
#($H = 0$) (%)	18.42%*	16.67%	18.42%
$\|A - WH\|_F / \|A\|_F$	0.5027	0.5032	0.5027
# of iterations	5385	3670	90
Computing time	284.0 sec.	192.8 sec.	8.3 sec.

morphologies: 10 classic medulloblastomas, 10 malignant gliomas, 10 rhabdoids, and 4 normals, forming a data matrix $A \in \mathbb{R}^{5,597\times 34}$. In addition to a lower cutoff threshold value of 20, an upper cutoff threshold value of 16,000 was used to eliminate expression values that are too high and may undesirably dominate the objective function value in Eqn. (1.2).

III. Artificial Data Sets with Zero Residual: We generated the first artificial data matrix A_a of size 200×50 by $A_a = W_a H_a$, where $W_a \in \mathbb{R}^{200\times 6}$ and $H_a \in \mathbb{R}^{6\times 50}$ are artificial positive matrices. The rank of A_a is 6 and a zero residual solution for the NMF with $k = 6$ exists. Accordingly, the NMF algorithms are expected to produce the solutions W and H, which give very small relative residual $\|A_a - WH\|_F / \|A_a\|_F$ with $k = 6$. We generated another artificial data matrix A_s of size $2,500 \times 28$ by $A_s = W_s H_s$, where $W_s \in \mathbb{R}^{2,500\times 3}$ and $H_s \in \mathbb{R}^{3\times 28}$ are artificial non-negative matrices. The basis matrix W_s has columns of unit L_2-norm. The maximal value in H_s is 10^5. The rank of A_s is 3 and a zero residual solution for the NMF with $k = 3$ exists.

6.2. Convergence Criteria. Reaching a smaller approximation error $\|A - W_* H_*\|_F$, where W_* and H_* are the solution matrices obtained from an algorithm for the NMF formulation in Eqn. (1.2), indicates the superiority of an algorithm in terms of approximation capability. Accordingly, the convergence of the proposed algorithms may be tested by checking the decrease in the residual of the objective function $f(W, H)$. We may also test the convergence to a stationary point by checking the Karush-Kuhn-Tucker (KKT) optimality conditions. The KKT conditions shown in Eqn. (2.3) can be rewritten as

$$\begin{array}{l} \min(W, \partial f(W,H)/\partial W) = 0, \\ \min(H, \partial f(W,H)/\partial H) = 0, \end{array} \tag{6.1}$$

where the minimum is taken component wise [11]. The normalized KKT residual Δ is then defined as $\Delta = \frac{\Delta_o}{\delta_W + \delta_H}$ which reflects the average of convergence errors for elements in W and H that did not converge, where

$$\begin{array}{rl} \Delta_o = & \sum_{i=1}^{m} \sum_{q=1}^{k} |\min(W_{iq}, (\partial f(W,H)/\partial W)_{iq})| + \\ & \sum_{q=1}^{k} \sum_{j=1}^{n} |\min(H_{qj}, (\partial f(W,H)/\partial H)_{qj})|, \end{array} \tag{6.2}$$

FIG. 6.1. *The values of Δ vs. the number of iterations for NMF/ANLS, NMF/NUR [22], and NMF/ALS [1] on the leukemia ALLAML data set with $k = 3$. We used the KKT convergence criterion with $\epsilon = 10^{-9}$.*

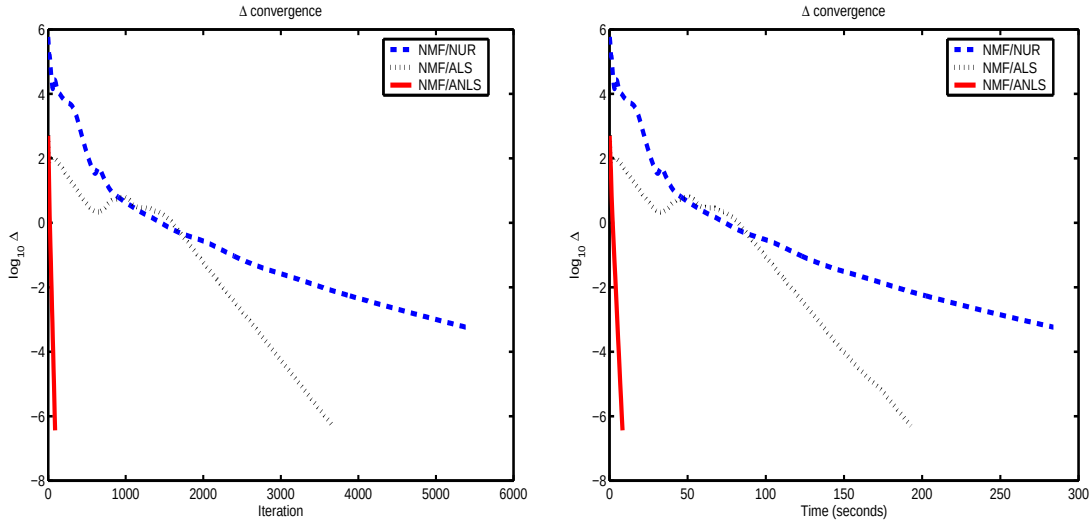
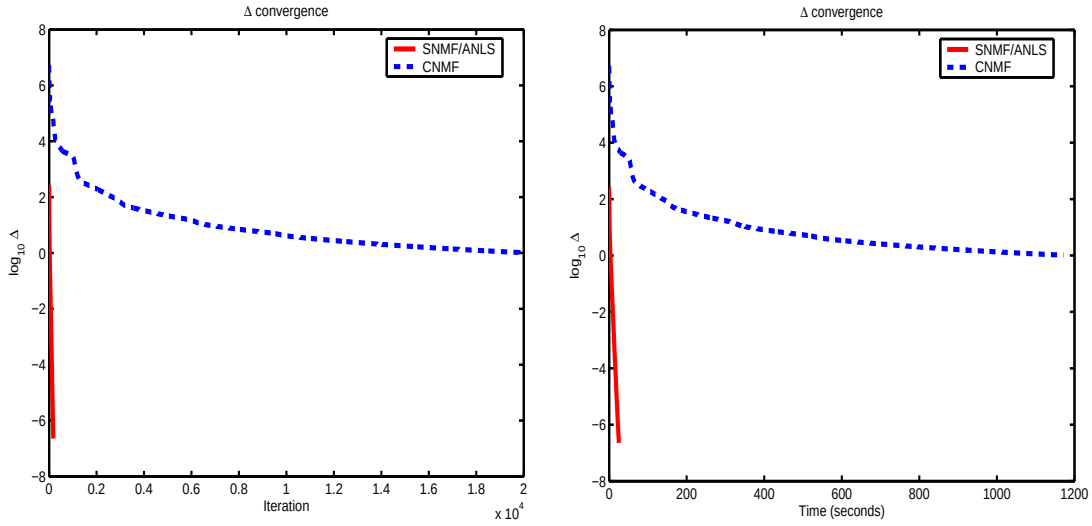


FIG. 6.2. *The values of Δ vs. the number of iterations for SNMF/R [18] with $\beta = 0.01$ and CNMF based on multiplicative update rules [30] with $\alpha = 0$ and $\beta = 0.01$ on the leukemia ALLAML data set with $k = 3$. We used the KKT convergence criterion with $\epsilon = 10^{-9}$.*



δ_W=#(min(W,$\partial f(W,H) / \partial W$) $\neq 0$), and δ_H=#(min(H,$\partial f(W,H) / \partial H$) $\neq 0$). Then the convergence criterion is defined as

$$\Delta \leq \epsilon \Delta_1, \tag{6.3}$$

where Δ_1 is the value of Δ after one iteration and ϵ is an assigned tolerance.

6.3. Performance Comparisons. In this subsection, we present performance results based on the three data sets described earlier. In the tests, we used the KKT convergence criterion shown in Eqn. (6.3) with $\epsilon = 10^{-9}$.

I. Test Results on the ALLAML Data Set: Table 6.1 shows the performance comparison among NMF/NUR, NMF/ALS, and NMF/ANLS on the ALLAML leukemia data matrix with $k = 3$. There are three clusters in this data set[1]. We report the percentage of zero elements in the computed factors W and H, relative approximation error (i.e. $\|A - WH\|_F / \|A\|_F$), the number of iterations, and computing time. The results show that to reach the same convergence criterion, NMF/NUR and NMF/ALS took much longer than NMF/ANLS, and the NMF/ALS generated the solutions with the largest relative approximation error among them. We believe that the overall faster performance of the NMF/ANLS is a result of its convergence properties. In the factors W and H, the NMF/NUR produced very small non-negative elements ($< 10^{-8}$) in W and H, which are not necessarily zeros, while NMF/ANLS generated the exact zero elements. This is an interesting property of the NMF algorithms and illustrates that the NMF/ANLS does better at generating sparser factors, which can be helpful in reducing computing complexity and storage requirement for handling sparse data sets.

Figure 6.1 further illustrates the convergence behavior of NMF/ANLS, NMF/NUR, and NMF/ALS on the ALLAML data set with $k = 3$. As for NMF/ALS, we solved each least squares subproblem by normal equations and set the negative values to zeros. All three algorithms began with the same random initial matrix of H_o. An additional random initial matrix of W_o was needed for NMF/NUR. The NMF/ALS generated the smallest Δ_1 (Δ value after the first iteration), whereas NMF/NUR produced the largest Δ_1. While NMF/NUR converged after more than 5,000 iterations from relatively large Δ_1, the final Δ value is still larger than those of other algorithms. We observed that the NMF/ALS algorithm required more running time than NMF/ANLS even though its subproblem (unconstrained least squares problem) requires less floating point operations. This slower computational time can be ascribed to the lack of convergence property of the NMF/ALS algorithm. In this test, NMF/ANLS outperformed the others in terms of convergence speed.

Figure 6.2 illustrates the converge behavior of SNMF/R with $\beta = 0.01$ and CNMF with $\alpha = 0$ and $\beta = 0.01$ on the ALLAML data set with $k = 3$. We used the KKT convergence criterion corresponding to each of the objective functions for SNMF/R and CNMF. The η parameter in SNMF/R was set to the square of the maximal value in the ALLAML data matrix. As for CNMF, we used the CNMF algorithm based on multiplicative update rules [30] without column normalization of W in each iteration. Two algorithms began with a random initial matrix W_o that has columns of unit L_2-norm. A random initial matrix of H_o was used only in CNMF. SNMF/R generated much smaller Δ than CNMF within a short time. The percentages of zero elements in W and H obtained from SNMF/R were 2.17% and 30.70%. On the other hand, the percentages of elements in the range of $[0, 10^{-8})$ in W and H obtained from CNMF were 2.71% and 18.42% and only a small number of elements in W were exactly zeros. It illustrates that the SNMF/R is more effective in producing a sparser H.

II. Test Results on the CNS Tumors Data Set: Table 6.2 shows the performance comparison on the CNS tumors data set with various k values where NMF/ANLS was a few orders of magnitude faster than NMF/NUR. NMF/NUR did not satisfy the KKT convergence criterion within 20,000

[1]The results of NMF algorithms with $k = 4$ and $k = 5$ can be found in our paper [17]

TABLE 6.2

*Performance comparison between NMF/NUR [22] and NMF/ANLS on the CNS tumors data set. We report the percentages of zero elements in W and H, relative approximation error, the number of iterations, and computing time (in seconds). *For NMF/NUR, the computed W and H factors were not sparse, so the percentages of the number of the non-negative elements that are smaller than 10^{-8} in W and H are shown instead.*

Algorithm	NMF/NUR		
Reduced rank k	3	4	5
#($\|W_{ij}\| < 10^{-8}$) (%)	8.70%*	9.05%*	12.32%*
#($\|H_{ij}\| < 10^{-8}$) (%)	18.63%*	25.00%*	25.29%*
$\|A - WH\|_F / \|A\|_F$	0.40246175083	0.37312046970	0.35409585961
# of iterations	20000	20000	20000
Computing time	1310.0 sec.	1523.0 sec.	1913.9 sec.
Algorithm	NMF/ANLS		
Reduced rank k	3	4	5
#($W = 0$) (%)	8.69%	9.03%	12.07%
#($H = 0$) (%)	18.63%	25.00%	27.06%
$\|A - WH\|_F / \|A\|_F$	0.40246175028	0.37312046948	0.35409574992
# of iterations	150	130	130
Computing time	14.8 sec.	16.6 sec.	20.4 sec.

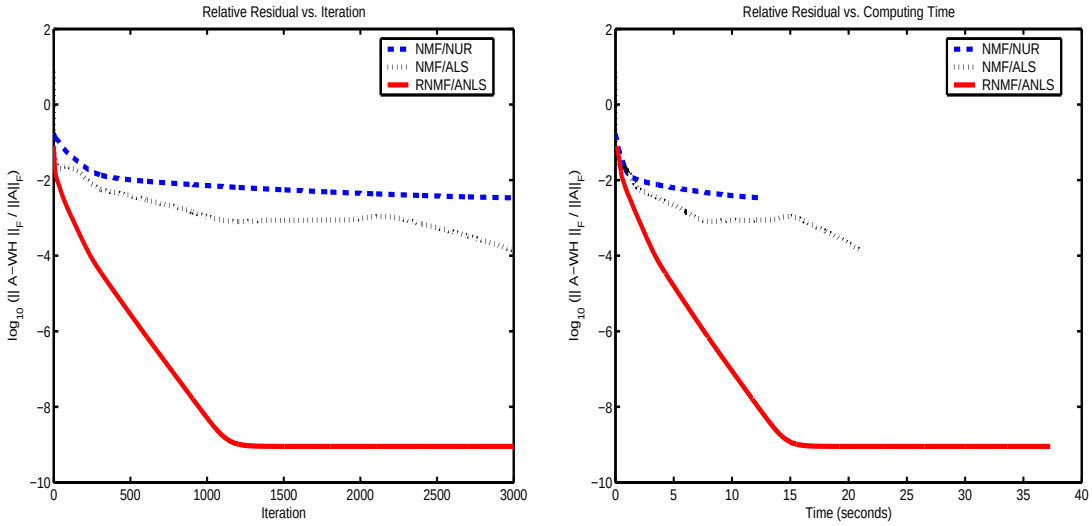
iterations. The relative approximation errors of NMF/NUR at the last iteration were still slightly larger than those of NMF/ANLS after less than 200 iterations.

The sparsity obtained from NMF/NUR or NMF/ANLS is a general result due to non-negativity constraints. Even when the original data set has no zero element, the factors W and H may have zero components. In case of NMF/ANLS, this becomes clear when we note that at the core of NMF/ANLS is the active set based iterations and in each iteration, the solution components that correspond to active set index are set to be zeros.

III. Test Results on the Zero Residual Artificial Data Sets: Figure 6.3 shows the performance of the three NMF algorithms, RNMF/ANLS, NMF/NUR, and NMF/ALS, on the first artificial data matrix $A_a = W_a H_a$ of size 200×50 where $W_a \in \mathbb{R}^{200\times 6}$ and $H_a \in \mathbb{R}^{6\times 50}$ are artificial positive matrices. The relative residuals versus iteration or computing time are shown. We used $\alpha = \beta = 10^{-8}$ for the RNMF/ANLS, and implemented NMF/ALS with pseudo-inverse. We note that NMF/ALS sometimes generated ill-conditioned W and H when negative values are set to zeros, which may happen even when we solve the least squares problem by a stable algorithm. In the worst case, the entire row or column in the matrices W or H may become zero. The RNMF/ANLS rapidly converged, while NMF/NUR did not converge to near zero residual within 3,000 iterations. The relative residual in the middle of iterative optimization of NMF/ALS sometimes increased.

Figure 6.4 shows the comparison between the truncated SVD [9] and NMF/ANLS on the second artificial data matrix $A_s = W_s H_s$ of size 2,500 $\times$ 28, where $W_s \in \mathbb{R}^{2,500\times 3}$ and $H_s \in \mathbb{R}^{3\times 28}$

FIG. 6.3. *The relative residuals vs. the number of iterations for RNMF/ANLS with* $\alpha = \beta = 10^{-8}$*, NMF/NUR [22], and NMF/ALS [1] with* $k = 6$ *for 3,000 iterations on the first artificial data matrix* $A_a = W_a H_a$ *of size* 200×50*, where* $W_a \in \mathbb{R}^{200\times 6}$ *and* $H_a \in \mathbb{R}^{6\times 50}$ *are artificial positive matrices, and* $rank(A_a) = 6$*.*



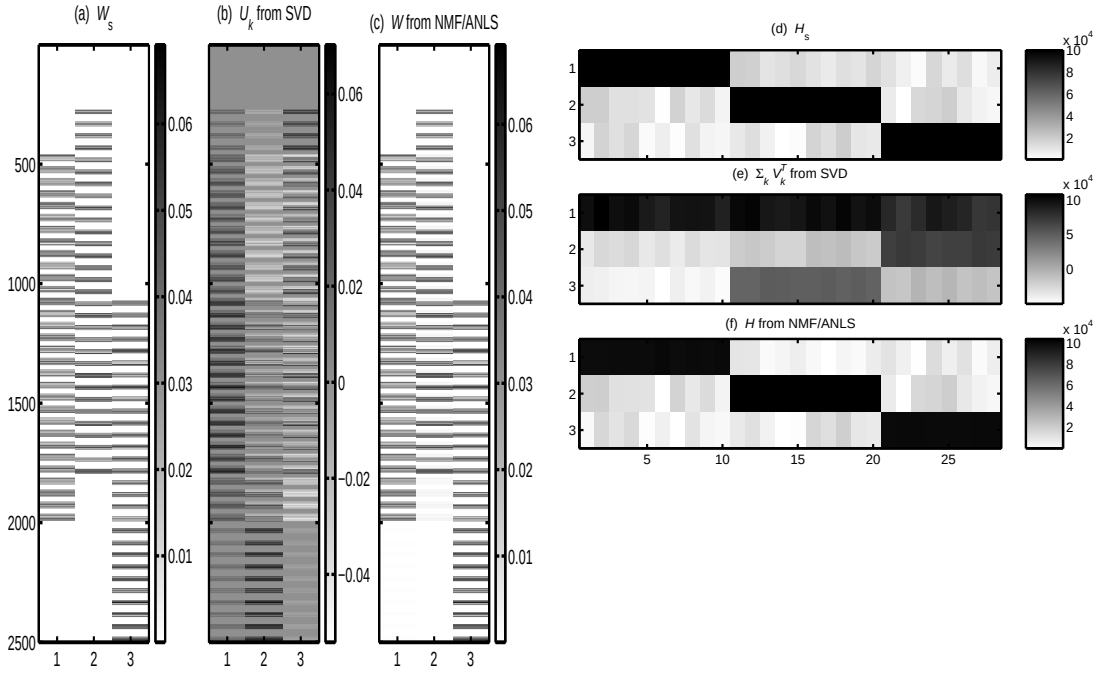
are artificial non-negative matrices. We presented U_k and $\Sigma_k V_k^T$ obtained from the truncated SVD ($A_s \approx U_k \Sigma_k V_k^T$) with $k = 3$. We also illustrated W and H obtained from NMF/ANLS ($A_s \approx WH$ s.t. $W, H \geq 0$) with $k = 3$. Although the approximation error of NMF/ANLS was larger than that of the truncated SVD, it surprisingly recovered W_s and H_s factors much better. Our NMF algorithm can be utilized for blind source separation when basis vectors are non-negative and observations are non-subtractive combinations of basis vectors.

6.4. Summary of Experimental Results. In our tests, the convergence of NMF/NUR was slower and, due to this, the algorithm was often prematurely terminated before it reaches a convergence criterion, whether it was based on the relative residual or KKT residual. The NMF/ALS does not provide a solution in a least squares sense for each non-negativity constrained subproblem although the problem is formulated as a least squares problem. Therefore, its convergence is difficult to analyze and exhibits non-monotonic changes in the objective function value throughout the iterations. On the other hand, NMF/ANLS generated solutions with satisfactory accuracy within a reasonable time.

An algorithm for non-negativity constrained least squares is an essential component of NMF/ANLS. There are several ways to solve the NLS problem with multiple right hand sides and we chose Van Benthem and Keenan's NLS algorithm [36]. This algorithm is based on the active set method that is guaranteed to terminate in *a finite number of steps* of matrix computations. Some other implementations of NLS are based on traditional gradient descent or quasi-Newton optimization methods. They are iterative methods that require explicit convergence check parameters. Their speed and accuracy depend on their convergence check parameters.

7. Summary and Discussion. We have introduced the NMF algorithms based on alternating non-negativity constrained least squares, for which every limit point is a stationary point. The core of our algorithm is the non-negativity constrained least squares algorithm for multiple

FIG. 6.4. *The factors obtained from the truncated SVD [9] ($A_s \approx U_k \Sigma_k V_k^T$) and NMF/ANLS ($A_s \approx WH$ s.t. $W, H \geq 0$) with $k = 3$ on the second artificial data matrix $A_s = W_s H_s$ of size 2,500 × 28, where $W_s \in \mathbb{R}^{2,500 \times 3}$ and $H_s \in \mathbb{R}^{3 \times 28}$ are artificial non-negative matrices, and $rank(A_s) = 3$. The gray scale indicates the values of the elements in the matrix.*



right hand sides based on the active set method, which terminates in *a finite number of steps*. We applied the well known convergence theory for block coordinate descent methods in bound constrained optimization and built a rigorous convergence criterion based on the KKT conditions.

We have established a framework of NMF/ANLS, which is theoretically sound and practically efficient. This framework was utilized to design formulations and algorithms for sparse NMFs and regularized NMF. Some theoretical characteristics of our proposed algorithms explain their superior behavior shown in the test results. The NMF algorithms based on gradient descent method exhibit slow convergence. Thus, it is possible to undesirably use premature solutions for data analysis owing to termination before convergence, which may sometimes lead to unreliable conclusions. The *inexact* NMF/ALS algorithm [1] sets the negative components in the unconstrained least squares solution to zeros. Although the inexact method may solve the subproblems faster, its convergence behavior is problematic. On the other hand, our algorithm satisfies the non-negativity constraints exactly in each subproblem and shows faster overall convergence. The converged solutions obtained from our algorithms make it possible to reach more physically reliable conclusions in many applications of NMF. The NMF/ANLS can be applied to a wide variety of practical problems in the fields of text data mining, image analysis, bioinformatics, computational biology, and so forth, especially when preserving non-negativity is beneficial to meaningful interpretation.

Acknowledgments. We would like to thank Prof. Chih-Jen Lin and Prof. Luigi Grippo for discussions on the convergence properties.

REFERENCES

[1] M. W. BERRY, M. BROWNE, A. N. LANGVILLE, V. P. PAUCA, AND R. J. PLEMMONS, *Algorithms and applications for approximate nonnegative matrix factorization*, 2006. *Computational Statistics and Data Analysis*, to appear.

[2] D. P. BERTSEKAS, *Nonlinear Programming, second edition*, Athena Scientifi c, Belmont, MA 02178-9998, 1999.

[3] R. BRO AND S. DE JONG, *A fast non-negativity-constrained least squares algorithm*, J. Chemometrics, 11 (1997), pp. 393–401.

[4] J. P. BRUNET, P. TAMAYO, T. R. GOLUB, AND J. P. MESIROV, *Metagenes and molecular pattern discovery using matrix factorization*, Proc. Natl Acad. Sci. USA, 101 (2004), pp. 4164–4169.

[5] M. CHU, F. DIELE, R. PLEMMONS, AND S. RAGNI, *Optimality, computation and interpretation of nonnegative matrix factorization*, 2004. preprint.

[6] C. DING, T. LI, W. PENG, AND H. PARK, *Orthogonal nonnegative matrix tri-factorizations for clustering*, in Proc. Int'l Conf. on Knowledge Discovery and Data Mining (KDD 2006), Aug. 2006.

[7] D. DUECK, Q. D. MORRIS, AND B. J. FREY, *Multi-way clustering of microarray data using probabilistic sparse matrix factorization*, Bioinformatics, 21 (2005), pp. i144–i151.

[8] Y. GAO AND G. CHURCH, *Improving molecular cancer class discovery through sparse non-negative matrix factorization*, Bioinformatics, 21 (2005), pp. 3970–3975.

[9] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations, third edition*, Johns Hopkins University Press, Baltimore, 1996.

[10] T. R. GOLUB, D. K. SLONIM, P. TAMAYO, C. HUARD, M. GAASENBEEK, J. P. MESIROV, H. COLLER, M. L. LOH, J. R. DOWNING, M. A. CALIGIURI, C. D. BLOOMFIELD, AND E. S. LANDER, *Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring*, Science, 286 (1999), pp. 531–537.

[11] E. F. GONZALES AND Y. ZHANG, *Accelerating the Lee-Seung algorithm for non-negative matrix factorization*, tech. report, Department of Computational and Applied Mathematics, Rice University, 2005.

[12] L. GRIPPO AND M. SCIANDRONE, *On the convergence of the block nonlinear Gauss-Seidel method under convex constraints*, Operations Research Letters, 26 (2000), pp. 127–136.

[13] P. O. HOYER, *Non-negative sparse coding*, in Proc. IEEE Workshop on Neural Networks for Signal Processing, 2002, pp. 557–565.

[14] ———, *Non-negative matrix factorization with sparseness constraints*, Journal of Machine Learning Research, 5 (2004), pp. 1457–1469.

[15] D. KIM, S. SRA, AND I. S. DHILLON, *Fast Newton-type methods for the least squares nonnegative matrix approximation problem*, in Proceedings of the 2007 SIAM International Conference on Data Mining (SDM07), 2007, pp. 343–354.

[16] H. KIM AND H. PARK, *Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares*, in Proceedings of the IASTED International Conference on Computational and Systems Biology (CASB2006), D.-Z. Du, ed., Nov. 2006, pp. 95–100.

[17] ———, *Cancer class discovery using non-negative matrix factorization based on alternating non-negativity-constrained least squares*, in Springer Verlag Lecture Notes in Bioinformatics (LNBI), vol. 4463, May 2007, pp. 477–487.

[18] ———, *Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis*, Bioinformatics, 23 (2007), pp. 1495–1502.

[19] P. M. KIM AND B. TIDOR, *Subsystem identification through dimensionality reduction of large-scale gene expression data*, Genome Research, 13 (2003), pp. 1706–1718.

[20] C. L. LAWSON AND R. J. HANSON, *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ, 1974.

[21] D. D. LEE AND H. S. SEUNG, *Learning the parts of objects by non-negative matrix factorization*, Nature, 401 (1999), pp. 788–791.

[22] ——, *Algorithms for non-negative matrix factorization*, in Proceedings of Neural Information Processing Systems, 2000, pp. 556–562.
[23] S. Z. LI, X. HOU, H. ZHANG, AND Q. CHENG, *Learning spatially localized parts-based representations*, in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2001, pp. 207–212.
[24] C. J. LIN, *Projected gradient methods for non-negative matrix factorization*, Tech. Report Information and Support Service ISSTECH-95-013, Department of Computer Science, National Taiwan University, 2005.
[25] W. LIU AND J. YI, *Existing and new algorithms for nonnegative matrix factorization*, tech. report, University of Texas at Austin, 2003.
[26] MATLAB, *User's Guide*, The MathWorks, Inc., Natick, MA 01760, 1992.
[27] P. PAATERO AND U. TAPPER, *Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics, 5 (1994), pp. 111–126.
[28] H. PARK AND H. KIM, *One-sided non-negative matrix factorization and non-negative centroid dimension reduction for text classification*, in Proceedings of the Workshop on Text Mining at the 6th SIAM International Conference on Data Mining (SDM06), M. Castellanos and M. W. Berry, eds., 2006.
[29] A. PASCUAL-MONTANO, J. M. CARAZO, K. KOCHI, D. LEHMANN, AND R. D. PASCUAL-MARQUI, *Nonsmooth nonnegative matrix factorization (nsNMF)*, IEEE Trans. Pattern Anal. Machine Intell., 28 (2006), pp. 403–415.
[30] V. P. PAUCA, J. PIPER, AND R. J. PLEMMONS, *Nonnegative matrix factorization for spectral data analysis*, 2006. *Linear Algebra and Applications*, to appear.
[31] V. P. PAUCA, F. SHAHNAZ, M. W. BERRY, AND R. J. PLEMMONS, *Text mining using non-negative matrix factorizations*, in Proc. SIAM Int'l Conf. Data Mining (SDM'04), April 2004.
[32] J. PIPER, V. P. PAUCA, R. J. PLEMMONS, AND M. GIFFIN., *Object characterization from spectral data using nonnegative factorization and information theory*, in Proc. Amos Technical Conf., 2004.
[33] S. L. POMEROY, P. TAMAYO, M. GAASENBEEK, L. M. STURLA, M. ANGELO, M. E. MCLAUGHLIN, J. Y. KIM, L. C.GOUMNEROVA, P. M. BLACK, C. LAU, J. C. ALLEN, D. ZAGZAG, J. M. OLSON, T. CURRAN, C. WETMORE, J. A. BIEGEL, T. POGGIO, S. MUKHERJEE, R. RIFKIN, A. CALIFANO, G. STOLOVITZKY, D. N. LOUIS, J. P. MESIROV, E. S. LANDER, AND T. R. GOLUB, *Prediction of central nervous system embryonal tumour outcome based on gene expression*, Nature, 415 (2002), pp. 436–442.
[34] S. SRA AND I. S. DHILLON, *Nonnegative matrix approximation: algorithms and applications*, Tech. Report 06-27, University of Texas at Austin, 2006.
[35] R. TIBSHIRANI, *Regression shrinkage and selection via LASSO*, J. Roy. Statist. Soc. B, 58 (1996), pp. 267–288.
[36] M. H. VAN BENTHEM AND M. R. KEENAN, *Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems*, J. Chemometrics, 18 (2004), pp. 441–450.
[37] R. ZDUNEK AND A. CICHOCKI, *Non-negative matrix factorization with quasi-Newton optimization*, in The Eighth International Conference on Artifi cial Intelligence and Soft Computing (ICAISC), 2006, pp. 870–879.