Operators
and
Matrices

Volume 3, Number 1 (2009), 41–58

ON THE RATE OF CONVERGENCE OF THE IMAGE SPACE RECONSTRUCTION ALGORITHM

JIAN HAN, LIXING HAN *, MICHAEL NEUMANN ** AND UPENDRA PRASAD

(*communicated by L. Rodman*)

Abstract. The Image Space Reconstruction Algorithm (ISRA) of Daube–Witherspoon and Muehllehner is a multiplicative algorithm for solving nonnegative least squares problems. Eggermont has proved the global convergence of this algorithm. In this paper, we analyze its rate of convergence. We show that if at the minimum the strict complementarity condition is satisfied and the reduced Hessian matrix is positive definite, then the ISRA algorithm which converges to it does so at a linear rate of convergence. If, however, the ISRA algorithm converges to a minimum which does not satisfy the strict complementarity condition, then the rate of convergence of the algorithm can degenerate to being sublinear. Our results here therefor hold under more general assumptions than in the work of Archer and Titterington who assume that at a minimum point all Lagrange multipliers are zero.

We provide numerical examples to illustrate our rate of convergence results and to explain why the ISRA algorithm usually appears to converge slowly. Our work here heuristically justifies why the Lee–Seung algorithm for solving nonnegative matrix factorization problems has a slow rate of convergence.

1. Introduction

Let m and n be positive integers. Let $b \in \mathbb{R}^{\mathrm{m}}$ be a nonnegative vector and $A = (a_{i,j}) \in \mathbb{R}^{\mathrm{m,n}}$ a nonnegative matrix such that

$$\sum_{i=1}^{m} a_{i,j} > 0, \quad \text{for } j = 1, 2, \ldots, n. \tag{1.1}$$

We consider the following nonnegative least squares problem:

$$\min \frac{1}{2} \|Ax - b\|_2^2, \quad \text{subject to} \quad x \geqslant 0, \tag{1.2}$$

where $x \in \mathbb{R}^{\mathrm{n}}$.

Mathematics subject classification (2000): 65F20, 65Y20, 15A48, 15A23.

Keywords and phrases: nonnegative least squares, ISRA (Image Space Reconstruction Algorithm), rate of convergence, nonnegative matrix factorization.

* Work of this author was supported in part by a research grant from the Office of Research, University of Michigan–Flint.

** Research partially supported by NSA Grant No. 06G–232.

The gradient and the Hessian matrix of the objective function of problem (1.2) are

$$g(x) \ = \ A^TAx - A^Tb \quad \text{and} \quad G(x) \ = \ A^TA,$$

respectively. As the Hessian matrix is positive semidefinite, problem (1.2) is a convex optimization problem. Therefore, each of its local minimizers is a global minimizer, see for example, Fletcher [13] and Nocedal and Wright [25]. We shall refer to a global minimizer a *minimizer* from now on.

The nonnegative least squares problem arises in various applications, see for instance, Berman and Plemmons [5] and Vogel [31]. Several algorithms have been developed to solve this problem (see Bellavia [4] and Vogel [31] and references therein). Among these algorithms, the so called *Image Space Reconstruction Algorithm* (ISRA) distinguishes itself from others by its use of a multiplicative updating approach.

The ISRA algorithm was proposed by Daube–Witherspoon and Muehllehner [9] in 1986. Starting with an initial vector $x^0 > 0$ (that is, x^0 is a vector whose components are all positive), this algorithm generates a sequence of $\{x^k\}$ using the following updates:

$$x_j^{k+1} \ = \ x_j^k \frac{(A^Tb)_j}{(A^TAx^k)_j}, \ \ j = 1, 2, \ldots, n. \tag{1.3}$$

Since matrix A has positive column sums (1.1), we have that

$$(A^TA)_{j,j} > 0,$$

for each $j = 1, 2, \cdots, n$. If $(A^Tb)_j > 0$ for index j, then x_j^k is well defined and $x_j^k > 0$ for all k because $(A^TAx^k)_j \geqslant (A^TA)_{j,j} \cdot x_j^k > 0$. If the $(A^Tb)_{j_0} = 0$ for some j_0, then we have that $x_{j_0}^1 = 0$. Although we may still have that $(A^TAx^k)_{j_0} > 0$ and therefore $x_{j_0}^k = 0$ for all k, we can use the following pre-processing strategy to avoid a zero denominator in iteration (1.3): Divide the index set $I = \{1, 2, \cdots, n\}$ into two subsets: $I_0 = \{j \in I | (A^Tb)_j = 0\}$ and $I_1 = I \backslash I_0$. Introduce the reduced matrix $\tilde{A}$ and the reduced vector $\tilde{x}$ by deleting the columns of A and the components of x corresponding to indices in I_0 respectively. We then update $\tilde{x}$ using the ISRA iteration:

$$\tilde{x}_j^{k+1} \ = \ \tilde{x}_j^k \frac{(\tilde{A}^Tb)_j}{(\tilde{A}^T\tilde{A}\tilde{x}^k)_j}, \ \ j \in I_1, \tag{1.4}$$

where $\tilde{x}^0$ is chosen to be a positive vector.

REMARK 1.1. (1) We remark that there is no need to update x_j, for $j \in I_0$, when this pre–processing strategy is used. By a convergence theorem of Eggermont [11] (see Theorem 1.2 later in this section) we can show that the sequence $\{\tilde{x}^k\}$ generated by (1.4) converges to some $\tilde{x}^*$ which is a minimizer of the following reduced nonnegative least squares problem:

$$\min \ \frac{1}{2} \ \|\tilde{A}\tilde{x} - b\|_2^2, \quad \text{subject to} \quad \tilde{x} \ \geqslant \ 0, \tag{1.5}$$

Moreover, if we define

$$x^* = \begin{cases} 0, & \text{if } j \in I_0, \\ \tilde{x}_j^*, & \text{if } j \in I_1, \end{cases}$$

then x^* is a minimizer of the original problem (1.2). For this reason, we will assume, without loss of generality, that $(A^T b)_j > 0$, for every $j = 1, \ldots, n$, from now on.

(2) In practice, one can also use the following modified ISRA formula to avoid division by zero (see for example, [6, 8]):

$$x_j^{k+1} = x_j^k \frac{(A^T b)_j}{(A^T A x^k)_j + \epsilon}, \; j = 1, 2, \ldots, n,$$

where $\epsilon > 0$ is a small number. A typical choice is $\epsilon = 10^{-9}$ (see [6]). Since we are concerned with the convergence and the rate of convergence of the ISRA method in this paper, we will not pursue further this modified ISRA method.

Throughout this paper it will be convenient to adopt the notation $\mathbb{R}_+^{k,\ell}$ to denote the set of all $k \times \ell$ matrices with nonnegative entries and to adopt the notation $\mathbb{R}_+^{k}$ to denote all the k–dimensional vectors with nonnegative entries.

The ISRA algorithm and its variants have been used in positron emission tomography, image deblurring, image reconstruction, and other areas, see for example, Anconelli, Bertero, Boccacci, Carbiller, and Lanteri[1], Anderson, Yust, and Mair [2], Archer and Titterington [3], Bertero and Boccacci [7], Daub–Witherspoon and Muehllehner [9], Holte, Schmidlin, Linen, Rosenqvist, and Eriksson [15], Kontaxakis, Strauss, and van Kaick [22], Kontaxakis, Strass, Thiereou, Ledesma-Carbayo, Santos, Pavlopoulos, and Dimitrakopoulou–Strauss [23], Morales, Medero, Santiago, and Sosa [24], Ollinger and Karp [26], and Vio, Nagy, Tenorio, and Wamsteker [30]. The performance of the ISRA algorithm is comparable to that of the popular EMML algorithm, see Shepp and Vardi [27] and Vardi, Shepp, and L. Kaufmann [28], which is based on minimizing the Kullback–Leibler divergence of a nonnegative linear system.

We point out that the ISRA algorithm has several additional interesting features: First, it can be easily parallelized, see, for example, Elsner, Koltracht, and Neumann [12]. Second, the algorithm can be readily adapted to solve the nonnegative least squares problem in matrix form:

$$\min \frac{1}{2}\|AX - B\|_F^2, \; \text{subject to } X \geqslant 0, \tag{1.6}$$

where $A \in \mathbb{R}_+^{m,n}$ and $B \in \mathbb{R}_+^{m,p}$ are given matrices, $X \in \mathbb{R}^{n,p}$ is the unknown matrix, and $\|\cdot\|_F$ is the Frobenius norm. For instance, the ISRA updates for solving (1.6) can be implemented using the following Matlab command:

$$X = X.*((A'*B)./(A'*A*X)); \tag{1.7}$$

Third, the ISRA method lays a foundation for understanding the important Lee and Seung [21] algorithm for solving the nonnegative matrix factorization (NNMF) problem [20, 21, 6, 8]:

$$\min \frac{1}{2}\|V - WH\|_F^2, \tag{1.8}$$

over all nonnegative matrix pairs $W \in \mathbb{R}^{m,r}$ and $H \in \mathbb{R}^{r,n}$, for a given $1 \leqslant r \leqslant \operatorname{rank}(V)$. The Lee–Seung [21] algorithm generates a sequence of approximation pairs (W^k, H^k) by updating W and H as follows:

1. *Initialization.* Choose matrices $W^0 \in \mathbb{R}_+^{m,r}$ and $H^0 \in \mathbb{R}_+^{r,n}$, and set $k = 0$.
2. *Iteration.* Iterate until convergence:

(1) Update H with W fixed:

$$H^{k+1} = H^k \; .* \; ((W^k)^T V) \; ./ \; ((W^k)^T W^k H^k) \tag{1.9}$$

(2) Update W with H fixed:

$$W^{k+1} = W^k \; .* \; (V(H^{k+1})^T) \; ./ \; ((W^k H^{k+1}(H^{k+1})^T) \tag{1.10}$$

(3) Set $k = k+1$.

It can be checked that the Lee–Seung algorithm updates W and H alternately, using the ISRA iteration formula in matrix form (1.7). The consequence of this observation is that if we are able, as we intend to do in this paper, to show that the rate of convergence of the ISRA algorithm is, at best, linear, then this is at least a heuristic proof that the rate of convergence of the Lee–Seung algorithm is at best linear. We comment that it is not known if the Lee–Seung algorithm converges to a Karush–Kuhn–Tucker point of the constrained optimization problem (1.8). For more information about the nonnegative matrix factorization problem and its algorithms and applications, see recent surveys [6, 8].

Continuing, De Pierro [10] and Eggermont [11] have studied the global convergence of the ISRA algorithm. In particular, Eggermont proves:

THEOREM 1.2. (Eggermont [11]) *Let $A \in \mathbb{R}_+^{m,n}$ and $b \in \mathbb{R}_+^n$. Then for any initial $x^0 > 0$, the ISRA algorithm, given in (1.3), generates a sequence of iterates $\{x^k\}$ such that*

$$\lim_{k\to\infty} x^k = x^*,$$

where x^* is a (global) minimizer of (1.2).

REMARK 1.3. (1) The original convergence theorem in [11] assumes that $\sum_{i=1}^{m} a_{i,j} = 1$, for $j = 1, 2, \ldots, n$. Note that this assumption can be easily relaxed as $\sum_{i=1}^{m} a_{i,j} > 0$ for $j = 1, 2, \ldots, n$, by right multiplying A by the diagonal matrix

$$\operatorname{diag}\left(\left[1/\sum_{i=1}^{m} a_{i,1}, \ldots, 1/\sum_{i=1}^{m} a_{i,n}\right]\right).$$

(2) The theorem above holds for the ISRA algorithm in its matrix form (1.7).

From Theorem 1.2 we see that the global convergence of the ISRA algorithm is well established. Thus here we shall focus on its rate of convergence. We begin by mentioning that Archer and Titterington obtained a rate of convergence matrix in [3, Page 94]. However, their analysis does not take into consideration the case of nonzero Lagrange multipliers which can arise in the course of constrained optimization. Hence the Archer and Titterington results are not applicable when nonzero Lagrange multipliers arise (see Remark 3.4 for more details below) and so in this paper we analyze the rate of convergence of the ISRA algorithm regardless of whether some or all the Lagrange multipliers that arise in the course of the algorithm are zero or nonzero.

The plan of our paper is as follows. In Section 2 we shall develop some optimization and matrix theoretic preliminaries necessary to achieve the goals of this paper. In Section 3 we prove our main results. We shall show that if at the minimizer the strict complementarity condition (cf. Definition 2.2) is satisfied and the reduced Hessian matrix (cf. (2.3)) is positive definite, then the ISRA algorithm which converges to it does so at a linear rate of convergence. If, however, the ISRA algorithm converges to a minimizer which does not satisfy the strict complementarity condition, then the rate of convergence of the algorithm can deteriorate to being sublinear. In Section 4 we will use numerical experiments to illustrate the results of this paper and expain why the ISRA algorithm converges slowly. As explained earlier, our work here can be used to explain heuristically why the Lee–Seung algorithm for solving NNMF problems has a slow rate of convergence.

2. Preliminaries

In this section we present some preliminary results in optimization and matrix theories which will be useful to establish the rate of convergence of ISRA algorithm.

Since problem (1.2) is a convex optimization problem, we have the following necessary and sufficient condition for x^* to be a minimizer of the problem:

LEMMA 2.1. (See Fletcher [13]) *Let $A \in \mathbb{R}_+^{m,n}$ and $b \in \mathbb{R}_+^m$. Then the point $x^* \in \mathbb{R}^n$ is a minimizer of* (1.2) *if and only if it is a Karush–Kuhn–Tucker* (KKT) *point of* (1.2)*, that is, it satisfies the following KKT conditions:*

$$(A^TAx^*)_j - (A^Tb)_j = \mu_j, \quad \mu_j x_j^* = 0, \quad \mu_j \geqslant 0, \quad x_j^* \geqslant 0, \tag{2.1}$$

for $j = 1, 2, \ldots, n$, where the μ_j's are the Lagrange multipliers corresponding to x^*.

The condition that $\mu_j x_j^* = 0$, $j = 1, 2, \ldots, n$, is often called the *complementary condition*. A useful concept in optimization theory is the so–called *strict complementarity*.

DEFINITION 2.2. (Strict complementarity) Suppose that x^* is a KKT point with the Lagrange multipliers $\{\mu_j\}$. We say that x^* satisfies the *strict complementarity condition* if

$$x_j^* = 0 \text{ implies } \mu_j > 0 \text{ and } \mu_j = 0 \text{ implies } x_j^* > 0. \tag{2.2}$$

For any $x \in \mathbb{R}_+^n$, we shall denote by $\mathscr{A}(x) = \{j : x_j = 0\}$ to denote the *active constraint index set* and by $\mathscr{I}(x) = \{j : x_j > 0\}$ to denote the *inactive constraint index set* of problem (1.2). Clearly $\mathscr{A}(x) \bigcup \mathscr{I}(x) = \{1, 2, \ldots, n\}$.

Another useful concept in optimization theory is that of the *reduced Hessian matrix*. If the indices in $\mathscr{I}(x)$ are $i_1, i_2, \ldots, i_{r(x)}$ are ordered such that $i_1 < i_2 < \ldots < i_{r(x)}$, then the *reduced Hessian matrix* of (1.2) at $x \in \mathbb{R}^n_+$ is given by:

$$G_R(x) = \begin{bmatrix} (A^TA)_{i_1 i_1} & (A^TA)_{i_1 i_2} & \dots & (A^TA)_{i_1 i_{r(x)}} \\ (A^TA)_{i_2 i_1} & (A^TA)_{i_2 i_2} & \dots & (A^TA)_{i_2 i_{r(x)}} \\ \vdots & \vdots & \ddots & \vdots \\ (A^TA)_{i_{r(x)} i_1} & (A^TA)_{i_{r(x)} i_2} & \dots & (A^TA)_{i_{r(x)} i_{r(x)}} \end{bmatrix}. \tag{2.3}$$

Obviously $G_R(x)$ is positive semidefinite.

We remark that it is often assumed that the strict complementarity holds and the reduced Hessian matrix is positive definite in the convergence rate analysis for algorithms for constrained optimization, see for example, [13, 25].

We will also need the following two lemmas from matrix theory.

LEMMA 2.3. *If $T \in \mathbb{R}^{r,r}$ is a positive definite matrix and $D = \mathrm{diag}(d_1, d_2, \dots, d_r)$ is a diagonal matrix such that $d_i > 0$, for $i = 1, \dots, r$, then all eigenvalues of the matrix DT are positive real numbers. Moreover,*

$$\min_{1 \leqslant i \leqslant r}\{\lambda_i(DT)\} \geqslant \min_{1 \leqslant i \leqslant r}\{d_i\} \min_{1 \leqslant i \leqslant r}\{\lambda_i(T)\}, \tag{2.4}$$

where $\lambda_i(\cdot)$ denotes the i–th eigenvalue of a matrix.

Proof. It is a well known result of Wigner that if A and B are positive semidefinite matrices, then the eigenvalues of their product are all nonnegative. Put $D^{1/2} = \mathrm{diag}(d_1^{1/2}, d_2^{1/2}, \dots, d_r^{1/2})$ and note that the spectrum of DT is identical to the spectrum of $D^{1/2}TD^{1/2}$. Let $z \neq 0$ be the eigenvector corresponding to the eigenvalue $\lambda = \min_{1 \leqslant i \leqslant r}\{\lambda_i(D^{1/2}TD^{1/2})\}$. Then, from the Raleigh ratios we find that

$$\lambda z^T z = z^T D^{1/2} T D^{1/2} z \geqslant \min_{1 \leqslant i \leqslant r}\{\lambda_i(T)\} z^T D z \geqslant \min_{1 \leqslant i \leqslant r}\{\lambda_i(T)\} \min_{1 \leqslant i \leqslant r}\{d_i\} z^T z.$$

Therefore, we have $\lambda \geqslant \min_{1 \leqslant i \leqslant r}\{\lambda_i(T)\} \min_{1 \leqslant i \leqslant r}\{d_i\}$. This implies (2.4). □

With Lemma 2.3 at hand we can prove:

LEMMA 2.4. *Let $T \in \mathbb{R}^{r,r}_+$ be a positive definite matrix and $y \in \mathbb{R}^r$ a vector with positive entries. Let $d_1, \dots, d_r$ be numbers such that*

$$0 < d_i \leqslant y_i/(Ty)_i, \tag{2.5}$$

for $i = 1, 2, \dots, r$, and put

$$D = \mathrm{diag}(d_1, d_2, \dots, d_r).$$

Then

$$\rho(I - DT) \leqslant 1 - \min_{1 \leqslant i \leqslant r}\{d_i\} \min_{1 \leqslant i \leqslant r}\{\lambda_i(T)\}, \tag{2.6}$$

where $\rho(\cdot)$ denotes the spectral radius of a matrix.

Proof. Let $D_0 = \mathrm{diag}((Ty)_1/y_1, (Ty)_2/y_2, \ldots, (Ty)_r/y_r)$. Then the matrix $D_0 - T$ has nonpositive off–diagonal entries and, as $(D_0 - T)y = 0$, $D_0 - T$ a singular M–matrix (see for example, [5, Chapter 6]). Since D_0^{-1} is a diagonal M–matrix, we see that $I - D_0^{-1}T = D_0^{-1}(D_0 - T)$ is also a singular M–matrix since it has nonpositive off–diagonal entries and is a product of two M–matrices.

Next (2.5) implies that

$$DT \leqslant D_0^{-1}T$$

showing that $I - DT$ is also an M–matrix. By Lemma 2.3, all eigenvalues of DT are positive real numbers. It follows that all eigenvalues of the $I - DT$ are nonnegative real numbers. Moreover, using Lemma 2.3 we have that:

$$\rho(I - DT) \leqslant 1 - \min_{1\leqslant i\leqslant r}\{\lambda_i(DT)\} \leqslant 1 - \min_{1\leqslant i\leqslant r}\{d_i\} \min_{1\leqslant i\leqslant r}\{\lambda_i(T)\}.$$

□

3. Rate of Convergence of ISRA

We are now ready to begin analyzing the rate of convergence of the ISRA algorithm. Assume that the ISRA algorithm generates the sequence $\{x^k\} \in \mathbb{R}^{\mathrm{n}}$ using a starting point $x^0 > 0$. According to Theorem 1.2,

$$\lim_{k\to\infty} x^k = x^*,$$

where x^* is a minimizer of (1.2). As discussed in Remark 1.3, without loss of generality, we assume that $(A^Tb)_j > 0$, for all j. Then, for each k, $x_j^k > 0$, for all j.

We consider two cases.

3.1. When Strict Complementarity Holds

We first consider the situation that the strict complementarity holds at x^*. We begin our analysis by assuming that:

$$\mathscr{I}(x^*) = \{1, 2, \ldots, r\}. \tag{3.1}$$

Then the Lagrange multipliers corresponding to x^* satisfy

$$\mu_j = 0, \quad j = 1, 2, \ldots, r, \tag{3.2}$$

and

$$\mu_j > 0, \quad j = r+1, r+2, \ldots, n. \tag{3.3}$$

Moreover, if we partition the Hessian matrix into four blocks:

$$A^TA = \begin{bmatrix} T & S \\ R & Q \end{bmatrix},$$

where $T \in \mathbb{R}^{\mathrm{r,r}}$, $Q \in \mathbb{R}^{\mathrm{n-r,n-r}}$, $S \in \mathbb{R}^{\mathrm{r,n-r}}$, and $R \in \mathbb{R}^{\mathrm{n-r,r}}$, then T is the reduced Hessian matrix at x^*.

We have the following rate of convergence result.

LEMMA 3.1. *Let $A \in \mathbb{R}_+^{m,n}$ and $b \in \mathbb{R}_+^m$ and suppose that the ISRA algorithm* (1.3) *applied to solving problem* (1.2) *generates a sequence of approximations $\{x^k\}$ which converges to a minimizer x^*. Suppose that $\mathscr{I}(x^*) = \{1, 2, \ldots, r\}$. Define*

$$M^* = \begin{bmatrix} I_r - D_1^* T & -D_1^* S \\ 0_{n-r,r} & D_2^* \end{bmatrix}, \tag{3.4}$$

$$D_1^* = \operatorname{diag}\left(\frac{x_1^*}{(A^T A x^*)_1}, \frac{x_2^*}{(A^T A x^*)_2}, \ldots, \frac{x_r^*}{(A^T A x^*)_r} \right), \tag{3.5}$$

and

$$D_2^* = \operatorname{diag}\left(\frac{(A^T b)_{r+1}}{(A^T A x^*)_{r+1}}, \frac{(A^T b)_{r+2}}{(A^T A x^*)_{r+2}}, \ldots, \frac{(A^T b)_n}{(A^T A x^*)_n} \right). \tag{3.6}$$

If at x^*, the strict complementarity holds and the reduced Hessian matrix T is positive definite, then the asymptotic rate of convergence of $\{x^k\}$ is $\rho(M^*)$ and it satisfies

$$\rho(M^*) \leqslant \max\left\{ 1 - \min_{1 \leqslant i \leqslant r} \left\{ \frac{x_i^*}{(A^T A x^*)_i} \right\} \min_{1 \leqslant i \leqslant r} \{\lambda_i(T)\}, \max_{r+1 \leqslant i \leqslant n} \left\{ \frac{(A^T b)_i}{(A^T A x^*)_i} \right\} \right\} < 1. \tag{3.7}$$

In particular, the ISRA algorithm converges linearly if $\rho(M^*) \neq 0$.

Proof. We begin by noticing that the ISRA updates as given in (1.3) can be rewritten as

$$x_j^{k+1} = x_j^k - \frac{x_j^k}{(A^T A x^k)_j} \left((A^T A x^k)_j - (A^T b)_j \right), \ j = 1, 2, \ldots, n. \tag{3.8}$$

As $\mathscr{I}(x^*) = \{1, 2, \ldots, r\}$, the Lagrange multipliers satisfy that $\mu_j = 0$, for $j = 1, 2, \ldots, r$. This implies that

$$(A^T A x^*)_j = (A^T b)_j, \ j = 1, 2, \ldots, r. \tag{3.9}$$

On using (3.9), the first r equations in (3.8) can be rewritten as

$$x_j^{k+1} - x_j^* = x_j^k - x_j^* - \frac{x_j^k}{(A^T A x^k)_j} \left(A^T A (x^k - x^*) \right)_j, \ j = 1, 2, \ldots, r. \tag{3.10}$$

For indices $j = r+1, r+2, \ldots, n$, we know that $x_j^* = 0$. As $x_j^{k+1} = x_j^{k+1} - x_j^*$ and $x_j^k = x_j^k - x_j^*$ for these indices, we can rewrite the corresponding updates in (1.3) as

$$x_j^{k+1} - x_j^* = (x_j^k - x_j^*) \frac{(A^T b)_j}{(A^T A x^k)_j}, \ j = r+1, r+2, \ldots, n. \tag{3.11}$$

We can thus rewrite (3.10) and (3.11) in the matrix–vector form:

$$x^{k+1} - x^* = M^k (x^k - x^*), \tag{3.12}$$

where

$$M^k = \begin{bmatrix} I_r - D_1^k T & -D_1^k S \\ 0_{n-r,r} & D_2^k \end{bmatrix}, \tag{3.13}$$

$$D_1^k = \operatorname{diag}\left(\frac{x_1^k}{(A^TAx^k)_1}, \frac{x_2^k}{(A^TAx^k)_2}, \ldots, \frac{x_r^k}{(A^TAx^k)_r}\right), \tag{3.14}$$

and

$$D_2^k = \operatorname{diag}\left(\frac{(A^Tb)_{r+1}}{(A^TAx^k)_{r+1}}, \frac{(A^Tb)_{r+2}}{(A^TAx^k)_{r+2}}, \ldots, \frac{(A^Tb)_n}{(A^TAx^k)_n}\right) \tag{3.15}$$

Because A is a nonnegative matrix and T is positive definite, we have that $(A^TAx^*)_i > 0$, for $i = 1, 2, \ldots, r$. Moreover, $(A^TAx^*)_i = (A^Tb)_i + \mu_i \geqslant \mu_i > 0$, for $i = r+1, r+2, \ldots, n$. Taking the limit of (3.13) which clearly exists, we obtain that

$$\lim_{k\to\infty} M^k = M^*,$$

where M^* is defined in (3.4). Thus the asymptotic rate of convergence of $\{x^k\}$ is $\rho(M^*)$.

To prove (3.7), we first note that

$$\rho(M^*) = \max\{\rho(I_r - D_1^*T), \rho(D_2^*)\}. \tag{3.16}$$

Define $\hat{x} = [x_1^*, x_2^*, \ldots, x_r^*]^T$. Then clearly we have

$$0 < \frac{x_i^*}{(A^TAx^*)_i} = \frac{\hat{x}_i}{(T\hat{x})_i}, \quad i = 1, 2, \ldots, r.$$

Thus from Lemma 2.4 it follows that

$$\rho(I_r - D_1^*T) \leqslant 1 - \min_{1\leqslant i\leqslant r}\left\{\frac{x_i^*}{(A^TAx^*)_i}\right\} \min_{1\leqslant i\leqslant r}\{\lambda_i(T)\} < 1. \tag{3.17}$$

On the other hand, $\mu_i > 0$, for $i = r+1, r+2, \ldots, n$, due to the strict complementarity. Thus the KKT conditions (2.1) imply

$$(A^TAx^*)_i - (A^Tb)_i > 0, \quad i = r+1, r+2, \ldots, n.$$

Hence we have

$$\rho(D_2^*) = \max_{r+1\leqslant i\leqslant n}\left\{\frac{(A^Tb)_i}{(A^TAx^*)_i}\right\} < 1. \tag{3.18}$$

Combining (3.16), (3.17) and (3.18), we have (3.7). The proof is complete. □

In Lemma 3.1, we assume that the first r constraints are inactive and the rest are active. Now we consider the the general case when

$$\mathscr{I}(x^*) = \{i_1, i_2, \ldots, i_r\}, \tag{3.19}$$

with $i_1 < i_2 < \ldots < i_r$. In this case, there is a permutation P such that

$$P(A^TA)P^T = \begin{bmatrix} \hat{T} & \hat{S} \\ \hat{R} & \hat{Q} \end{bmatrix},$$

where $\hat{T} = G_R(x^*) \in \mathbb{R}^{\mathrm{r,r}}$ is the reduced Hessian matrix at x^* as defined in (2.3) and $\hat{Q} \in \mathbb{R}^{\mathrm{n-r,n-r}}$, $\hat{S} \in \mathbb{R}^{\mathrm{r,n-r}}$, and $\hat{R} \in \mathbb{R}^{\mathrm{n-r,r}}$. In this case, we have

$$P(x^{k+1} - x^*) = \hat{M}^* P(x^k - x^*), \tag{3.20}$$

where

$$\hat{M}^* = \begin{bmatrix} I_r - \hat{D}_1^*\hat{T} & -\hat{D}_1^*\hat{S} \\ 0_{n-r,r} & \hat{D}_2^* \end{bmatrix}, \tag{3.21}$$

$$\hat{D}_1^* = \mathrm{diag}\left(\frac{x_i^*}{(A^TAx^*)_i}, i \in \mathscr{I}(x^*)\right),$$

and

$$\hat{D}_2^* = \mathrm{diag}\left(\frac{(A^Tb)_i}{(A^TAx^*)_i}, i \in \mathscr{A}(x^*)\right).$$

Note that the the sequesces of $\{x^k\}$ and $\{Px^k\}$ have the same asymptotic rate of convergence.

We can now state the major result of this paper.

THEOREM 3.2. *Let $A \in \mathbb{R}_+^{m,n}$ and $b \in \mathbb{R}_+^m$ and suppose that the ISRA algorithm* (1.3) *applied to solving problem* (1.2) *generates a sequence of approximations $\{x^k\}$ which converges to a minimizer x^*. If at x^*, the strict complementarity holds and the reduced Hessian matrix $\hat{T} = G_R(x^*)$ is positive definite, then the asymptotic rate of convergence of $\{x^k\}$ is*

$$\rho(\hat{M}^*) = \max\left\{\rho(I_r - \hat{D}_1^* G_R(x^*)), \rho(\hat{D}_2^*)\right\} \tag{3.22}$$

and it satisfies

$$\rho(\hat{M}^*) \leqslant \max\left\{1 - \min_{i\in\mathscr{I}(x^*)}\left\{\frac{x_i^*}{(A^TAx^*)_i}\right\}\min_{1\leqslant i\leqslant r}\{\lambda_i(G_R(x^*))\}, \max_{i\in\mathscr{A}(x^*)}\left\{\frac{(A^Tb)_i}{(A^TAx^*)_i}\right\}\right\} < 1. \tag{3.23}$$

In particular, the ISRA algorithm possesses a linear rate of convergence if $\rho(\hat{M}^*) \neq 0$.

REMARK 3.3. If $\rho(\hat{M}^*) = 0$, then the rate of convergence of the ISRA algorithm is better than a linear. The reason for this is that under the assumptions of Theorem 3.2, $\rho(\hat{M}^*) = 0$ if and only if A^TA is a positive definite diagonal matrix and $x_i^* > 0$, for all $i : 1 \leqslant i \leqslant n$. In this case,

$$\hat{M}^* = I_n - \mathrm{diag}\left(\frac{x_1^*}{(A^TAx^*)_1}, \ldots, \frac{x_n^*}{(A^TAx^*)_n}\right)A^TA = 0.$$

Since A is a nonnegative matrix, that A^TA is a positive definite diagonal matrix implies that A is up to a permutation a rectangular diagonal matrix. However, this situation hardly occurs in real applications.

3.2. When Strict Complementarity Does Not Hold

In contrast to the assumption made in Theorem 3.2, suppose that the minimizing point generated by the ISRA algorithm *does not* possess the strict complementarity condition at x^*. Then there is an index $1 \leqslant j_1 \leqslant n$ such that

$$x_{j_1}^* = 0 \text{ and } \mu_{j_1} = 0.$$

The KKT condition corresponding to this index then satisfies that:

$$(A^TAx^*)_{j_1} - (A^Tb)_{j_1} = 0.$$

Thus we have

$$\lim_{k\to\infty} \frac{(A^Tb)_{j_1}}{(A^TAx^k)_{j_1}} = 1.$$

This and the error equation:

$$x_{j_1}^{k+1} - x_{j_1}^* = (x_{j_1}^k - x_{j_1}^*)\frac{(A^Tb)_{j_1}}{(A^TAx^k)_{j_1}}$$

imply that the rate of convergence of $\{x^k\}$ is sublinear.

REMARK 3.4. In [3, (5.2)] Archer and Titterington obtained the following rate matrix using the notations of this paper:

$$M^* = I_n - \operatorname{diag}\left(\frac{x_1^*}{(A^Tb)_1}, \dots, \frac{x_n^*}{(A^Tb)_n}\right) A^TA. \tag{3.24}$$

The rate matrix becomes

$$M^* = I_n - \operatorname{diag}\left(\frac{x_1^*}{(A^TAx^*)_1}, \dots, \frac{x_n^*}{(A^TAx^*)_n}\right) A^TA, \tag{3.25}$$

if all the Lagrange multipliers μ_j corresponding to x^* are zero. However, nonnegative least squares problems often have nonzero Lagrange multipliers at x^*, as we shall exhibit in the next section. Thus, the rate matrix (3.24) does not generally give the correct rate except in the following two cases:

Case 1: $x_j^* > 0$, for $j = 1, 2, \dots, n$. Note that this implies $\mu_j = 0$, for all $j = 1, \dots, n$. We see that in this case our results in Theorem 3.2 confirm the Archer–Terrington finding in (3.24) that the convergence rate is linear.

Case 2: There is some index $1 \leqslant j \leqslant n$ such that both $x_j^* = 0$ and $\mu_j = 0$. In this case, the rate matrix (3.24) gives the correct sublinear rate of convergence which agrees with the rate obtained in Subsection 3.2.

It should be remarked that in many nonnegative least squares problems, such as in Examples 1 and 2 in the next section, we have that $x_j^* = 0$ and $\mu_j > 0$, for indices j's, and the assumptions of Theorem 3.2 hold. In this case, the rate matrix (3.24) gives a sublinear rate while the correct rate should be linear.

4. Numerical Results

A major complaint about the ISRA algorithm from practitioners is its slow convergence (see for example [26, 29, 30]). In this section, we present some numerical results to show how that the theoretical convergence results obtained in Section 3 explain to some extent this slow convergence behavior.

We mention that all of our numerical experiments were done on randomly generated nonnegative least squares problems, using MATLAB 7.5. As often the case in practice, we used $m \geqslant n$ in our computation.

Our numerical results largely confirmed that the ISRA algorithm often converges slowly. For instance, for randomly generated $A = \text{rand}(m,n)$ and $b = \text{rand}(m,1)$, it is typical that the rate of convergence

$$\lim_{k\to\infty} \frac{\|x^{k+1} - x^*\|_2}{\|x^k - x^*\|_2} \geqslant 0.99,$$

for $5n \geqslant m \geqslant n \geqslant 10$.

According to Subsection 3.2, one may wonder whether the slow convergence behavior of the ISRA algorithm is due to that the strict complementarity does not hold at the minimizer x^* found by the algorithm, which results in a sublinear rate. Our numerical results indicate that, in general, this is not the case. They also show that the algorithm converges at a slow linear rate. To illustrate this we consider two examples which are displayed in an appendix.

For the first example and starting vector displayed in the Appendix, the ISRA algorithm generated the minimizer

$$x^* = [0\ 0\ 0\ 0.2222\ 0\ 0.4381\ 0.0512\ 0\ 0.0690\ 0]^T$$

for the nonnegative least squares problem (1.2) and the corresponding Lagrange multipliers were

$$\mu = [0.0705\ 0.1008\ 0.2622\ 0\ 0.2338\ 0\ 0\ 0.4596\ 0\ 0.0169]^T.$$

Note that the strict complementarity holds. However, $x_{10}^* = 0$ and $\mu_{10} \approx 0$. We computed the rate of convergence, which is

$$\lim_{k\to\infty} \frac{\|x^{k+1} - x^*\|_2}{\|x^k - x^*\|_2} \approx 0.9951.$$

Thus the rate is linear. However, it is quite slow as 0.9951 is close to 1. We also computed $\rho(I_r - \hat{D}_1^* G_R(x^*))$ and $\rho(\hat{D}_2^*)$. They are 0.9829 and 0.9951 respectively. Therefore, our theoretical rate is $\rho(\hat{M}^*) \approx 0.9951$, which is the same as the computed rate.

For the second example and starting vector displayed in the Appendix, the ISRA algorithm generated the minimizer:

$$x^* = [0.1447\ 1.5040\ 1.0536\ 0\ 0\ 0\ 0.8190\ 0\ 0\ 0.4455]^T$$

for the nonnegative least squares problem (1.2) and the corresponding Lagrange multipliers

$$\mu = [0\ 0\ 0\ 0.5510\ 0.6194\ 1.9888\ 0\ 0.6589\ 0.7666\ 0]^T.$$

Note that once again the strict complementarity condition holds. Upon the computation of the rate of convergence one obtains that:

$$\lim_{k\to\infty} \frac{\|x^{k+1} - x^*\|_2}{\|x^k - x^*\|_2} \approx 0.9923.$$

Thus for this example too ISRA has slow linear rate of 0.9923 which is close to 1. We also computed $\rho(I_r - \hat{D}_1^* G_R(x^*))$ and $\rho(\hat{D}_2^*)$. They are 0.9923 and 0.9776

respectively. Therefore, our theoretical rate is $\rho(\hat{M}^*) \approx 0.9923$ which agrees with the computed rate.

Examples 1 and 2 and more numerical experiments have led us to the following observation: when the ISRA algorithm is used to solve randomly generated nonnegative least squares problems, it typically converges at a slow linear rate with the asymptotic rate $\rho(\hat{M}^*)$ being close to 1.

We have also observed that, although both $\rho(\hat{M}^*) = \rho(\hat{D}_2^*) > \rho(I_r - \hat{D}_1^* G_R(x^*))$ (as in Example 1) and $\rho(\hat{M}^*) = \rho(I_r - \hat{D}_1^* G_R(x^*)) > \rho(\hat{D}_2^*)$ (as in Example 2) can occur, the later situation seems to appear more frequently.

A natural question is why the rate of convergence of the ISRA algorithm is slow. From the rate (3.22) and the above observations, we expect that this is due to $\rho(I_r - \hat{D}_1^* G_R(x^*))$ or $\rho(\hat{D}_2^*)$ being close to 1. In our next experiment, we computed

$$\rho_{r,t} \;=\; \rho\left(I_r - \operatorname{diag}\left(\frac{x_1}{(A^TAx)_1}, \frac{x_2}{(A^TAx)_2}, \ldots, \frac{x_r}{(A^TAx)_r}\right) A^TA\right). \tag{4.1}$$

for randomly generated matrix $A = \operatorname{rand}(tr, r)$ and vector $x = \operatorname{rand}(r, 1)$, where $t \geqslant 1$ is an integer.

Our numerical tests using various values of r and t indeed confirm that $\rho_{r,t}$ can be very close to 1. To illustrate this, we report two cases here.

Case 1. For fixed $t = 2$ and for each $r : r = 1, 2, \ldots, 30$, we computed twenty $\rho_{r,2}$ values using twenty different matrices A and vectors x and then computed $\bar{\rho}_{r,2}$: the average of these values. The values of $\bar{\rho}_{r,2}$ for $r = 2 : 30$ are plotted in Figure 1 ($\bar{\rho}_{1,2}$ is not plotted here as $\rho_{1,2}$ is always zero).

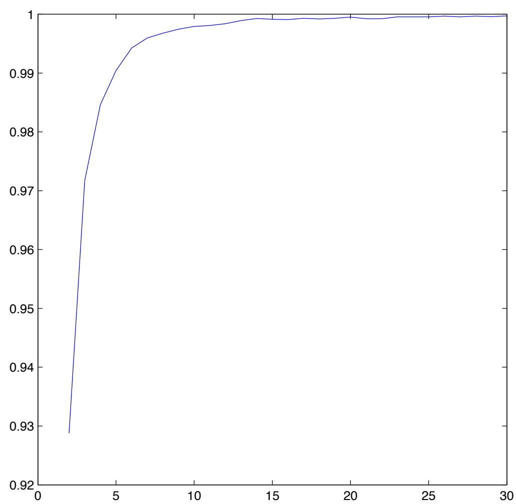

Figure 1. $\bar{\rho}_{r,2}$ *for* $r = 2, 3, \ldots, 30$

Case 2. For fixed $r = 10$ and for each $t = 2, 3, \ldots, 200$, we computed twenty $\rho_{10,t}$ values using twenty different matrices A and vectors x and then computed $\bar{\rho}_{10,t}$: the average of these values. The values of $\bar{\rho}_{10,t}$ for $t = 2:200$ are plotted in Figure 2.

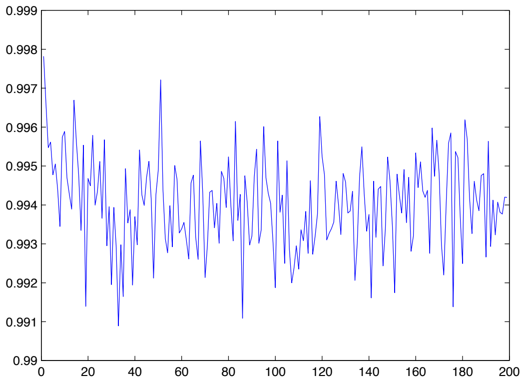

Figure 2. $\bar{\rho}_{10,t}$ *for* $t = 2, 3, \ldots, 200$

From Figure 1, we see that $\bar{\rho}_{2,2} > 0.92$, $\bar{\rho}_{3,2} > 0.97$, $\bar{\rho}_{4,2} > 0.98$, and $\bar{\rho}_{r,2} \geqslant 0.99$ for $r \geqslant 5$. From Figure 2, $\bar{\rho}_{10,t} \geqslant 0.99$ for $t \geqslant 2$. These numerical results indicate that for randomly generated nonnegative least squares problems, the ISRA algorithm converges at a slow linear rate, because $\rho(\hat{M}^*) \geqslant \rho(I_r - \hat{D}_1^* G_R(x^*))$ and $\rho(I_r - \hat{D}_1^* G_R(x^*))$ is highly likely to be close to 1. This numerical simulation also helps explain why the ISRA algorithm has a slow rate of convergence when applied to the nonnegative least squares problems arising in the real world applications, although these problems may not always have the same structure as randomly generated problems.

5. Final Remarks

We have shown that under standard assumptions, the rate of convergence of the ISRA algorithm is linear. We have also shown the rate can be sublinear if the strict complementarity condition does not hold. Our numerical tests have confirmed the folklore that the speed of the ISRA algorithm is slow. We comment that faster and more robust methods for the nonnegative least squares problem now exist (see for example, [4, 18]).

An interesting question is for what types of nonnegative least squares problems the ISRA algorithm converges at a faster linear rate. An extreme case is when $A^T A$ is a diagonal positive definite matrix. In this case, it finds a minimizer in one step. In general, the algorithm may have a faster linear rate if $G_R(x^*)$ is more diagonally dominant since this leads to a smaller $\rho(I_r - \hat{D}_1^* G_R(x^*))$.

Another interesting question is how to extend our rate of convergence results for the ISRA algorithm to the Lee–Seung [21] algorithm for nonnegative matrix factorization. As the Lee–Seung algorithm updates W and H alternately, using exactly the ISRA iteration formula in matrix form (1.7), we expect that this algorithm converges at a rate not faster than linear. We also expect that the Lee–Seung algorithm has a slow rate of convergence based on our explanation in Section 4 about the rate of convergence of ISRA algorithm. Several authors have observed that the Lee–Seung algorithm is often very slow (see for example, [6, 8, 14, 19]). A rigorous analysis of global convergence and rate of convergence of the Lee–Seung algorithm is desirable.

6. Appendix

EXAMPLE 1. In this example, $A \in \mathbb{R}^{20,10}$, $b \in \mathbb{R}^{20}$, and $x_0 \in \mathbb{R}^{10}$ are the following:

$$A = \begin{bmatrix}
0.1759 & 0.6476 & 0.5822 & 0.4046 & 0.3477 & 0.8217 & 0.5144 & 0.8507 & 0.7386 & 0.5523 \\
0.7218 & 0.6790 & 0.5407 & 0.4484 & 0.1500 & 0.4299 & 0.8843 & 0.5606 & 0.5860 & 0.6299 \\
0.4735 & 0.6358 & 0.8699 & 0.3658 & 0.5861 & 0.8878 & 0.5880 & 0.9296 & 0.2467 & 0.03200 \\
0.1527 & 0.9452 & 0.2648 & 0.7635 & 0.2621 & 0.3912 & 0.1548 & 0.6967 & 0.6664 & 0.6147 \\
0.3411 & 0.2089 & 0.3181 & 0.6279 & 0.04450 & 0.7691 & 0.1999 & 0.5828 & 0.08350 & 0.3624 \\
0.6074 & 0.7093 & 0.1192 & 0.7720 & 0.7549 & 0.3968 & 0.4070 & 0.8154 & 0.6260 & 0.04950 \\
0.1917 & 0.2362 & 0.9398 & 0.9329 & 0.2428 & 0.8085 & 0.7487 & 0.8790 & 0.6609 & 0.4896 \\
0.7384 & 0.1194 & 0.6456 & 0.9727 & 0.4424 & 0.7551 & 0.8256 & 0.9889 & 0.7298 & 0.1925 \\
0.2428 & 0.6073 & 0.4795 & 0.1920 & 0.6878 & 0.3774 & 0.7900 & 0.0005000 & 0.8908 & 0.1231 \\
0.9174 & 0.4501 & 0.6393 & 0.1389 & 0.3592 & 0.2160 & 0.3185 & 0.8654 & 0.9823 & 0.2055 \\
0.2691 & 0.4587 & 0.5447 & 0.6963 & 0.7363 & 0.7904 & 0.5341 & 0.6126 & 0.7690 & 0.1465 \\
0.7655 & 0.6619 & 0.6473 & 0.09380 & 0.3947 & 0.9493 & 0.09000 & 0.9900 & 0.5814 & 0.1891 \\
0.1887 & 0.7703 & 0.5439 & 0.5254 & 0.6834 & 0.3276 & 0.1117 & 0.5277 & 0.9283 & 0.04270 \\
0.2875 & 0.3502 & 0.7210 & 0.5303 & 0.7040 & 0.6713 & 0.1363 & 0.4795 & 0.5801 & 0.6352 \\
0.09110 & 0.6620 & 0.5225 & 0.8611 & 0.4423 & 0.4386 & 0.6787 & 0.8013 & 0.01700 & 0.2819 \\
0.5762 & 0.4162 & 0.9937 & 0.4849 & 0.01960 & 0.8335 & 0.4952 & 0.2278 & 0.1209 & 0.5386 \\
0.6834 & 0.8419 & 0.2187 & 0.3935 & 0.3309 & 0.7689 & 0.1897 & 0.4981 & 0.8627 & 0.6952 \\
0.5466 & 0.8329 & 0.1058 & 0.6714 & 0.4243 & 0.1673 & 0.4950 & 0.9009 & 0.4843 & 0.4991 \\
0.4257 & 0.2564 & 0.1097 & 0.7413 & 0.2703 & 0.8620 & 0.1476 & 0.5747 & 0.8449 & 0.5358 \\
0.6444 & 0.6135 & 0.06360 & 0.5201 & 0.1971 & 0.9899 & 0.05500 & 0.8452 & 0.2094 & 0.4452
\end{bmatrix},$$

$$b = \begin{bmatrix}
0.12393227759807 \\
0.49035729346802 \\
0.85299815534082 \\
0.87392740586173 \\
0.27029433229270 \\
0.20846135875131 \\
0.56497957073820 \\
0.64031182516276 \\
0.41702895164289 \\
0.20597551553224 \\
0.94793312129317 \\
0.082071207097726 \\
0.10570942658172 \\
0.14204112190400 \\
0.16646044087642 \\
0.62095864393531 \\
0.57370976484120 \\
0.052077890285870 \\
0.93120138460825 \\
0.72866168167827
\end{bmatrix}, \text{ and } x_0 = \begin{bmatrix}
0.737841653797590 \\
0.0634045006928180 \\
0.860440563038232 \\
0.934405118961213 \\
0.984398312240972 \\
0.858938816683866 \\
0.785558989265031 \\
0.513377418587575 \\
0.177602460505865 \\
0.398589496735843
\end{bmatrix}.$$

EXAMPLE 2. In this example, $A \in \mathbb{R}^{20,10}$, $b \in R^{20}$, and $x_0 \in R^{10}$ are the following:

$$A = \begin{bmatrix} 0.7136 & 0.4923 & 0.7150 & 0.5864 & 0.9300 & 0.2859 & 0.02160 & 0.8295 & 0.3889 & 0.2753 \\ 0.6183 & 0.6947 & 0.8562 & 0.6751 & 0.3990 & 0.5437 & 0.5598 & 0.8491 & 0.4547 & 0.7167 \\ 0.3433 & 0.9727 & 0.2815 & 0.3610 & 0.04740 & 0.9848 & 0.3008 & 0.3725 & 0.2467 & 0.2834 \\ 0.9360 & 0.3278 & 0.7311 & 0.6203 & 0.3424 & 0.7157 & 0.9394 & 0.5932 & 0.7844 & 0.8962 \\ 0.1248 & 0.8378 & 0.1378 & 0.8112 & 0.7360 & 0.8390 & 0.9809 & 0.8726 & 0.8828 & 0.8266 \\ 0.7306 & 0.7391 & 0.8367 & 0.01930 & 0.7947 & 0.4333 & 0.2866 & 0.9335 & 0.9137 & 0.3900 \\ 0.6465 & 0.9542 & 0.1386 & 0.08390 & 0.5449 & 0.4706 & 0.8008 & 0.6685 & 0.5583 & 0.4979 \\ 0.8332 & 0.03190 & 0.5882 & 0.9748 & 0.6862 & 0.5607 & 0.8961 & 0.2068 & 0.5989 & 0.6948 \\ 0.3983 & 0.3569 & 0.3662 & 0.6513 & 0.8936 & 0.2691 & 0.5975 & 0.6539 & 0.1489 & 0.8344 \\ 0.7498 & 0.6627 & 0.8068 & 0.2312 & 0.05480 & 0.7490 & 0.8840 & 0.07210 & 0.8997 & 0.6096 \\ 0.8352 & 0.2815 & 0.5038 & 0.4035 & 0.3037 & 0.5039 & 0.9437 & 0.4067 & 0.4504 & 0.5747 \\ 0.3225 & 0.2304 & 0.4896 & 0.1220 & 0.04620 & 0.6468 & 0.5492 & 0.6669 & 0.2057 & 0.3260 \\ 0.5523 & 0.7111 & 0.8770 & 0.2684 & 0.1955 & 0.3077 & 0.7284 & 0.9337 & 0.8997 & 0.4564 \\ 0.9791 & 0.6246 & 0.3531 & 0.2578 & 0.7202 & 0.1387 & 0.5768 & 0.8110 & 0.7626 & 0.7138 \\ 0.5493 & 0.5906 & 0.4494 & 0.3317 & 0.7218 & 0.4756 & 0.02590 & 0.4845 & 0.8825 & 0.8844 \\ 0.3304 & 0.6604 & 0.9635 & 0.1522 & 0.8778 & 0.3625 & 0.4465 & 0.7567 & 0.2850 & 0.7209 \\ 0.6195 & 0.04760 & 0.04230 & 0.3480 & 0.5824 & 0.7881 & 0.6463 & 0.4170 & 0.6732 & 0.01860 \\ 0.3606 & 0.3488 & 0.9730 & 0.1217 & 0.07070 & 0.7803 & 0.5212 & 0.9718 & 0.6643 & 0.6748 \\ 0.7565 & 0.4513 & 0.1892 & 0.8842 & 0.9227 & 0.6685 & 0.3723 & 0.9880 & 0.1228 & 0.4385 \\ 0.4139 & 0.2409 & 0.6671 & 0.09430 & 0.8004 & 0.1335 & 0.9371 & 0.8641 & 0.4073 & 0.4378 \end{bmatrix},$$

$$b = \begin{bmatrix} 0.585184097528289 \\ 4.07340844759291 \\ 1.62427719315255 \\ 1.23114057117321 \\ 1.71356610561640 \\ 1.87846070004794 \\ 2.73276896541479 \\ 2.80960077132988 \\ 1.97911114312001 \\ 1.99065439765668 \\ 2.57683609605786 \\ 3.28765271047956 \\ 4.75457599132480 \\ 3.61174257163749 \\ 2.00039872681169 \\ 4.15935669664905 \\ 0.671691708643541 \\ 0.302333859699142 \\ 0.421235261573298 \\ 0.819491591648541 \end{bmatrix}, \text{ and } x_0 = \begin{bmatrix} 0.32421992029405 \\ 0.30172677720465 \\ 0.011680991130340 \\ 0.53990509384963 \\ 0.095372692627822 \\ 0.14651485644223 \\ 0.63114120701496 \\ 0.85932041142879 \\ 0.97422163123871 \\ 0.57083842747218 \end{bmatrix}.$$

REFERENCES

[1] B. ANCONELLI, M. BERTERO, P. BOCCACCI, M. CARBILLER, AND H. LANTERI, *Iteative methods for the reconstruction of astronomical images with high dynamic range*, Journal of Computational and Applied Mathematics, 198 (2007), 321–331.

[2] J. M. M. ANDERSON, C. K. YUST, AND B. A. MAIR, *Minimum distance methods for transmission tomography*, 1996 IEEE Nuclear Science Symposium, Vol. 3 (1996), 1767–1771.

[3] G. E. B. ARCHER AND D. M. TITTERINGTON, *The Image Space Reconstruction Algorithm (ISRA) as an alternative to the EM algorithm for solving positive linear inverse problems*, Statistica Sinica, 5 (1995), 77–96.

[4] S. BELLAVIA, M. MACCONI, AND B. MORINI, *An interior point Newton-like method for non-negative least squares problems with degenerate solution*, Numerical Linear Algebra with Applications, Vol. 13, No. 10 (2006), 825–846.

[5] A. BERMAN AND R. J. PLEMMONS, *Nonnegative Matrices in the Mathematical Sciences*, SIAM, 1994.

[6] M. BERRY, M. BROWNE, A. LANGVILLE, P. PAUCA, AND R. J. PLEMMONS, *Algorithms and applications for approximate nonnegative matrix factorization*, Computational Statistics and Data Analysis, 52 (2007), 155–173.

[7] M. BERTERO AND P. BOCCACCI, *Image restoration methods for the Large Binocular Telescope (LBT)*, Astronomy & Astrophysics Supplementary Series, 147 (2000), 323–333.

[8] A. CICHOCKI, R. ZDUNEK, AND S. AMARI, *Nonnegative matrix and tensor factorization*, Signal Processing Magazine, IEEE, Vol. 25, No. 1 (2008), 142–145.

[9] M. E. DAUBE-WITHERSPOON AND G. MUEHLLEHNER, *An iterative image space reconstruction algorithm suitable for volume ECT*, IEEE Trans. Med. Imaging, MI-5 (1986), 61–66.

[10] A. R. DE PIERRO, *On the convergence of the iterative image space reconstruction algorithm for volume ECT*, IEEE Trans. Med. Imaging, MI-6 (1987), 174–175.

[11] P. P. B. EGGERMONT, *Multiplicative iterative algorithms for convex programming*, Linear Algebra and its Applications, 130 (1990), 25–42.

[12] L. ELSNER, I. KOLTRACHT, AND M. NEUMANN, *On the convergence of asynchronous paracontractions with application to tomographic reconstruction from incomplete data*, Linear Algebra and its Applications, 130 (1990), 65–82.

[13] R. FLETCHER, *Practical Methods of Optimization, Second edition*, John Wiley & Sons, 1987.

[14] E. F. GONZALEZ AND Y. ZHANG, *Accelerating the Lee-Seung algorithm for nonnegative matrix factorization*, Technical Report TR05-02, Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005, 2005

[15] S. HOLTE, P. SCHMIDLIN, A. LINDEN, G. ROSENQVIST, AND L. ERIKSSON, *Iterative image reconstruction for positive emission tomography: A study of convergence and quantitation problems*, IEEE Transactions on Nuclear Science, Vol. 37, No. 2 (1990), 629–635.

[16] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, 1985.

[17] L. KAUFMAN, *Maximum likelihood, least squares, and the penalized least squares for PET*, IEEE Transactions on Medical Imaging, 12 (1993), 200–214.

[18] D. KIM, S. SRA, AND I. DHILLON, *Fast Newton-type methods for the least squares non-negative matrix approximation problem*, Proceedings of SIAM Conference on Data Mining, 2007.

[19] C. J. LIN, *Projected gradient methods for nonnegative matrix factorization*, Neural Computation, 19 (2007), 2756–2779.

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

[21] D. D. LEE AND H. S. SEUNG, *Algorithms for non-negative matrix factorization*, Advances in Neural Information Processing Systems 13, 2000.

[22] G. KONTAXAKIS, L. G. STRAUSS, AND G. VAN KAICK, *Optimized image reconstruction for emission tomography using ordered subsets, median root prior and a web-based interface*, 1998 IEEE Nuclear Science Symposium, Vol. 2 (1998), 1347–1352.

[23] G. KONTAXAKIS, L. STRAUSS, T. THIEREOU, AND M. J. LEDESMA-CARBAYO, A. SANTOS, A. A. PAVLOPOULOS, AND A. DIMITRAKOPOULOU-STRAUSS, *Iterative image reconstruction for clinical PET using oedered-subsets, median root prior, and a web-based interface*, Molecular Imaging and Biology, Vol. 4, No. 3 (2002), 219–231.

[24] J. MORALES, N. MEDERO, N. G. SANTIAGO, AND J. SOSA, *Hardware Implementation of Image Reconstruction Algorithm using FPGAs*, MWSCAS '06. 49th IEEE International Midwest Symposium on Circuits and Systems, Vol. 1 (2006), 433–436. 2006.

[25] J. NOCEDAL AND S. WRIGHT, *Numerical Optimization*, Springer, 1999.

[26] J. M. OLLINGER AND J. S. KARP, *An evaluation of three algorithms for reconstructing images from data with missing projections*, IEEE Transactions on Nuclear Science, Vol. 35, No. 1 (1988), 629–634.
[27] L. A. SHEPP AND Y. VARDI, *Maximum likelihood reconstruction in emission tomography*, IEEE Trans. Med. Imaging, MI-1 (1982), 113–122.
[28] Y. VARDI, L. A. SHEPP, AND L. KAUFMAN, *A statistical model for positron emission tomography*, J. Amer. Statist. Assoc. 80 (1985), 8–38.
[29] M. VELEZ-REYES, A. PUETZ, P. HOKE, R. B. LOCKWOOD, AND S. ROSARIO, *Iterative Algorithms for unmixing hyperspectral imagery*, Proceedings of IPIE, Vol. 5093 (2003), 418–429.
[30] R. VIO, J. NAGY, L. TENORIO, AND W. WAMSTEKER, *A simple but efficient algorithm for multiple-image deblurring*, Astronomy & Astrophysics, 416 (2004), 403–410.
[31] C. R. VOGEL, *Computational Methods for Inverse Problems*, SIAM, Philadelphia, 2002.

(Received February 1, 2008)

Jian Han
Department of Mathematics
Tianjin University
Tianjin 300072, China
e-mail: hanjian@tju.edu.cn

Lixing Han
Department of Mathematics
University of Michigan–Flint
Flint, Michigan 48502, USA
e-mail: lxhan@umflint.edu

Michael Neumann
Department of Mathematics
University of Connecticut
Storrs, Connecticut 06269–3009, USA
e-mail: neumann@math.uconn.edu

Upendra Prasad
Department of Mathematics
University of Connecticut
Storrs, Connecticut 06269–3009, USA
e-mail: upendra@math.uconn.edu