Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 11, No. 3, pp. 459–469, July 2015

NEW ALGORITHMS FOR NONNEGATIVE MATRIX COMPLETION

Fang Xu* and Guoping He†

Abstract: Nonnegative matrix completion aims to find nonnegative low-rank matrices from a subset of entries of a matrix. It is widely applicable in many fields, such as image and video processing, recommendation systems, and national economy. This task can be conducted by solving the nuclear norm regularized linear least squares model with nonnegative constraints. We apply the widely used alternating direction method of multipliers to solve the model and get two novel algorithms. The applicability and efficiency of the new algorithms are demonstrated in numerical experiments. Recovery results show that our algorithms are helpful.

Key words: *nonnegative matrix completion, alternating direction method of multipliers*

Mathematics Subject Classification: *49Q99, 65K05*

1 Introduction

Matrix completion (MC) is the process of recovering the unknown or missing elements of a matrix. Under certain assumptions on the matrix, e.g. low-rank or approximately low-rank, the incomplete matrix can be reconstructed very well [2, 12]. Matrix completion is widely applicable in many fields, such as machine learning, control, image and video processing [10], where matrices with low rank or approximately low rank are widely used in the model construction.

Recently, there have been extensive research on the problems of low-rank matrix completion (LRMC). The affine rank minimization problem consists of finding a matrix of minimum rank that satisfies a given system of linear constraints. However, it is NP-hard (non-deterministic polynomial-time hard) due to the combinatorial nature of the rank function. [2, 12] showed that the solution of LRMC could be found by solving a nuclear norm minimization problem under some reasonable conditions. The singular value thresholding (SVT) method [1] and fixed point continuation method using approximate singular value decomposition (FPCA) [11] are two well-known algorithms because of their good recoverability, fast speed and robustness. SVT applied the linearized Bregman iterations to solve the unconstrained nuclear norm regularized linear least squares problem. FPCA used iterations

*Corresponding author. The work of F. Xu was supported in part by NSFC (Grant No. 11101274 and 11322109).

†The work of G. He was supported in part by NSFC (Grant No. 10971122), Scientific and Technological Projects (2009GG10001012) of Shandong Province, and Excellent Young Scientist Foundation (Grant No. 2010BSE06047 and BF2012SF025) of Shandong Province.

based on an iterative shrinkage-thresholding algorithm and used the continuation technique together with an approximate singular value decomposition procedure to accelerate the algorithm. [13] proposed an accelerated proximal gradient singular value thresholding algorithm. A completely different model was developed in LMaFit [15], which was a nonlinear successive over-relaxation algorithm that only requires solving a linear least squares problem per iteration. More details on LRMC can be found in [2, 3, 5, 7] and references therein.

In practice, the completed matrix is often required to be nonnegative. For example, the utility matrix of recommendation systems is nonnegative. If we ignore the minor by-product issue in life-cycle assessment and input-output analysis, the technology matrix is also nonnegative. They all motivate the development of nonnegative matrix completion (NMC). [17] considered the model in LMaFit with nonnegative constraints and used the alternating direction method of multipliers (ADMM) [4, 6, 9, 14] to solve the model. [16] considered the nuclear norm regularized linear least squares model with nonnegative constraints and also applied ADMM to solve the model. However, they replaced the quadratic term of a sub-problem by its first-order Taylor expansion and only obtained an approximate solution.

Our main contribution in this work is the development of two efficient algorithms of NMC. First of all, we present the nuclear norm regularized linear least squares model with nonnegative constraints. Because of its robustness, we choose it as the model of NMC in this paper. The structure of the model suggests an alternating minimization scheme, which is very suitable for solving large-scale problems. We give two exact ADMM-based algorithms, whose subproblems are solved exactly. We test new ADMM-based algorithms on two kinds of problems: random matrix completion problems and random low-rank approximation problems. Numerical experiments show that all our proposed algorithms output satisfactory results. The paper is organized as follows. Section 2 presents models and algorithms of NMC. Some numerical results are given in section 3.

The following notations will be used throughout this paper. Upper (lower) face letters are used for matrices (column vectors). All vectors are column vectors, the subscript $(\cdot)^T$ denotes matrix and vector transposition. $Diag(x)$ denotes a diagonal matrix with x on its main diagonal. $\mathbf{0}$ is a matrix of all zeros of proper dimension, I_n stands for the $n\times n$ identity matrix. The trace of $X \in \mathbb{R}^{m\times n}$, i.e., the sum of the diagonal elements of X, is denoted by $\mathrm{tr}(X)$. The Frobenius norm of $X \in \mathbb{R}^{m\times n}$ is defined as $\|X\|_F = \sqrt{\sum_{i,j} |X_{i,j}|^2}$. The Euclidean inner product between two matrices $X \in \mathbb{R}^{m\times n}$ and $Z \in \mathbb{R}^{m\times n}$ is defined as $\langle X, Z\rangle = \sum_{i,j}(X_{i,j}Z_{i,j}) = \mathrm{tr}(X^\top Z)$. The inequality $X \geq \mathbf{0}$ is element-wise, which means $X_{ij} \geq 0$ for all entries (i, j). Likewise, the equality $X = Z$ means $X_{ij} = Z_{ij}$ for all entries (i, j).

2 ADMM-Based Methods for NMC

2.1 The model of NMC

The matrix completion problem of recovering a nonnegative low-rank matrix from a subset of its entries is:

$$\begin{aligned} &\min \mathrm{rank}(X) \\ &\text{s.t. } X_{ij} = M_{ij},\ \forall (i,j) \in \Omega \\ &\quad X \geq \mathbf{0}, \end{aligned} \tag{2.1}$$

where $X \in \mathbb{R}^{m\times n}$ is the decision variable, Ω is the index set of p known elements of X.

Let $\mathcal{P}$ be the projection onto the subspace of sparse matrices with non-zeros restricted

to the index set Ω, i.e.,

$$\mathcal{P}_\Omega(X)_{ij} = \begin{cases} X_{ij}, & \text{if } (i,j) \in \Omega, \\ 0, & \text{otherwise.} \end{cases}$$

From the definition of $\mathcal{P}_\Omega$, we can reformulate the equality constrain in Model (2.1) in terms of $\mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M)$. Due to the combinational property of the objective function rank$(\cdot)$, Model (2.1) is NP-hard in general. Inspired by the success of matrix completion under nuclear norm in [2, 3, 12], we use the nuclear norm as an approximation to rank(X) to estimate the optimal solution X^* of Model (2.1) from the following model:

$$\begin{aligned} &\min \|X\|_* \\ &\text{s.t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M) \\ &\quad X \geq \mathbf{0}, \end{aligned} \tag{2.2}$$

where the nuclear norm $\|X\|_*$ of X is defined as the summation of the singular values of X, i.e.,

$$\|X\|_* = \sum_{i=1}^{\min(m,n)} \sigma_i(X),$$

where $\sigma_i(X)$ is the ith largest singular value.

If the known elements of the matrix X are noise free, that is to say $\mathcal{P}_\Omega(M)$ is reliable, we will directly solve Model (2.2) to conduct NMC. On the contrary, if the vector of known elements $\mathcal{P}_\Omega(M)$ are contaminated by noise, the constrains $\mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M)$ must be relaxed, resulting in the following problem:

$$\min_X \ \|X\|_*, \ \text{ s.t. } \|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F \leqslant \delta, \ X \geq \mathbf{0}, \tag{2.3}$$

or the nuclear norm regularized linear least squares model with nonnegative constraints:

$$\min_X \ \mu\|X\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2, \ \text{ s.t. } X \geq \mathbf{0}. \tag{2.4}$$

Here, δ and μ are given parameter, whose values should be set according to the noise level. When the values of μ and δ are set properly, (2.3) and (2.4) are equivalent. Model (2.4) is usually preferred over (2.3) for the case of noisy observations. Our algorithms can be extended to treat (2.3) with minor modifications.

Actually, Model (2.4) is especially useful in practice. The reason is that the known information is usually gotten from large surveys and contaminated by sampling error inevitably. In this paper, we choose Model (2.4) as the model to conduct NMC.

2.2 An ADMM-based method for Model (2.4)

In this subsection, we present an algorithm developed for Model (2.4). To facilitate an efficient use of ADMM, we introduce two new matrix (splitting) variables Y and Z, and consider an equivalent form of Model (2.4):

$$\begin{aligned} \min_{X,Y,Z} \ & \mu\|Y\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(Z) - \mathcal{P}_\Omega(M)\|_F^2, \\ \text{s.t. } & X \geq \mathbf{0}, \\ & Y = X, \\ & Z = X, \end{aligned} \tag{2.5}$$

where X, Y, $Z \in \mathbb{R}^{m\times n}$. The augmented Lagrangian function of Model (2.5) is:

$$\begin{aligned}\mathcal{L}(X,Y,Z,\Pi,\Lambda) = \mu\|Y\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(Z) - \mathcal{P}_\Omega(M)\|_F^2 + \langle \Pi, Y - X\rangle \\ + \langle \Lambda, Z - X\rangle + \frac{\alpha}{2}\|Y - X\|_F^2 + \frac{\beta}{2}\|Z - X\|_F^2,\end{aligned} \tag{2.6}$$

where Π, $\Lambda \in \mathbb{R}^{m\times n}$ are both Lagrangian multipliers. α, $\beta > 0$ are both penalty parameters.

The alternating direction method of multipliers for Model (2.5) is derived by successively minimizing $\mathcal{L}$ with respect to X, Y and Z in an alternating fashion, namely,

$$X_{k+1} := \arg\min_{X\geq \mathbf{0}} \mathcal{L}(X, Y_k, Z_k, \Pi_k, \Lambda_k), \tag{2.7a}$$

$$Y_{k+1} := \arg\min_{Y} \mathcal{L}(X_{k+1}, Y, Z_k, \Pi_k, \Lambda_k), \tag{2.7b}$$

$$Z_{k+1} := \arg\min_{Z} \mathcal{L}(X_{k+1}, Y_{k+1}, Z, \Pi_k, \Lambda_k), \tag{2.7c}$$

$$\Pi_{k+1} := \Pi_k + \gamma\alpha\,(Y_{k+1} - X_{k+1})\,, \tag{2.7d}$$

$$\Lambda_{k+1} := \Lambda_k + \gamma\beta\,(Z_{k+1} - X_{k+1})\,, \tag{2.7e}$$

where $\gamma \in (0, 1.618)$. By rearranging the terms of (2.7a), it is equivalent to

$$\min_{X\geq \mathbf{0}} \|X - \frac{\alpha Y_k + \beta Z_k + \Pi_k + \Lambda_k}{\alpha + \beta}\|_F^2,$$

whose solution is

$$X_{k+1} = \mathcal{P}_+(\frac{\alpha Y_k + \beta Z_k + \Pi_k + \Lambda_k}{\alpha + \beta}),$$

where $\mathcal{P}_+$ is the projection onto the nonnegative matrix subspace, i.e.,

$$\mathcal{P}_+(X)_{ij} = \begin{cases} X_{ij}, & \text{if } X_{ij} > 0, \\ 0, & \text{otherwise.}\end{cases}$$

By deleting the constant terms of problem (2.7b), we can get a more concise form:

$$\min_{Y} \ \mu\|Y\|_* + \langle \Pi_k, Y - X_{k+1}\rangle + \frac{\alpha}{2}\|Y - X_{k+1}\|_F^2,$$

which is equivalent to

$$\min_{Y} \ \mu\|Y\|_* + \frac{\alpha}{2}\|Y - (X_{k+1} - \Pi_k/\alpha)\|_F^2. \tag{2.8}$$

Lemma 2.1. *(Theorem 3 in [11]) Given a matrix $Y \in \mathbb{R}^{m\times n}$ with $rank(Y) = t$, let its Singular Value Decomposition (SVD) be $Y = U_Y Diag(w) V_Y^T$, where $U_Y \in \mathbb{R}^{m\times t}$, $w \in R_+^t$, $V_Y \in \mathbb{R}^{n\times t}$, and $\nu \geq 0$. Define the shrinkage operator $s_\nu(\cdot)$ as*

$$s_\nu(w) = \bar{w}, \ \textit{with } \bar{w}_i = \begin{cases} w_i - \nu, & \textit{if } w_i - \nu > 0, \\ 0, & \textit{otherwise.}\end{cases}$$

Then

$$A := \mathcal{S}_\nu(Y) = U_Y Diag(s_\nu(w)) V_Y^T$$

is an optimal solution of the problem

$$\min_{A\in R^{m\times n}} \ f(A) := \nu\|A\|_* + \frac{1}{2}\|A - Y\|_F^2.$$

Based on Lemma 2.1, we can get the solution of the Model (2.8):

$$Y_{k+1} = S_{\frac{\mu}{\alpha}}(X_{k+1} - \Pi_k/\alpha).$$

Note that Z_* is an optimal solution to (2.7c):

$$\min_Z \ \frac{1}{2}\|\mathcal{P}_\Omega(Z) - \mathcal{P}_\Omega(M)\|_F^2 + \langle \Lambda, Z - X_{k+1}\rangle + \frac{\beta}{2}\|Z - X_{k+1}\|_F^2,$$

if and only if

$$\mathcal{P}_\Omega^*(\mathcal{P}_\Omega(Z_*) - \mathcal{P}_\Omega(M)) + \Lambda_k + \beta(Z_* - X_{k+1}) = 0,$$

therefore the closed solution of (2.7c) can be written as

$$Z_{k+1} = (\mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{I})^{-1}(\mathcal{P}_\Omega(M) - \Lambda_k + \beta X_{k+1}),$$

where $\mathcal{I}$ is the identity operator and $\mathcal{P}_\Omega^*$ is the adjoint operator of $\mathcal{P}_\Omega$.

In short, ADMM applied to Model (2.5) yields the iteration:

$$\begin{aligned}
X_{k+1} &:= \mathcal{P}_+(\frac{\alpha Y_k + \beta Z_k + \Pi_k + \Lambda_k}{\alpha + \beta}), && (2.9a)\\
Y_{k+1} &:= S_{\frac{\mu}{\alpha}}(X_{k+1} - \Pi_k/\alpha), && (2.9b)\\
Z_{k+1} &:= (\mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{I})^{-1}(\mathcal{P}_\Omega(M) - \Lambda_k + \beta X_{k+1}), && (2.9c)\\
\Pi_{k+1} &:= \Pi_k + \gamma\alpha\,(Y_{k+1} - X_{k+1})\,, && (2.9d)\\
\Lambda_{k+1} &:= \Lambda_k + \gamma\beta\,(Z_{k+1} - X_{k+1})\,. && (2.9e)
\end{aligned}$$

From the above considerations, we arrive at Algorithms 1 below.

Algorithm 1: An exact ADMM-based algorithm of NMC

1 Input $\mathcal{P}_\Omega(M)$, $I_m \geq 0$, and $tol \geq 0$.
2 Set μ, γ, α, and $\beta \geq 0$. Set X_0, Y_0 and Z_0 as random matrices, and Π_0 and Λ_0 as zero matrices of appropriate sizes.
3 **while** *not converge* **do**
4 update $X_k, Y_k, Z_k, \Pi_k, \Lambda_k$ by the formulas (2.9).

2.3 Another ADMM-based method for Model (2.4)

In this subsection, we develop another algorithm for Model (2.4), which is also based on ADMM. Here, we only introduce a new matrix splitting variable Z and consider another equivalent form of Model (2.4):

$$\begin{aligned}
\min_{X,Z} \quad & \mu\|Z\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2\\
\text{s.t.} \quad & X = Z,\\
& X \geq \mathbf{0},
\end{aligned} \tag{2.10}$$

where $X, Z \in \mathbb{R}^{m\times n}$. The augmented lagrangian of Model (2.10) is:

$$\mathcal{L}(X, Z, \Lambda) = \mu\|Z\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2 + \langle \Lambda, X - Z\rangle + \frac{\rho}{2}\|X - Z\|_F^2,$$

where $\Lambda \in \mathbb{R}^{m\times n}$ is a lagrangian multiplier, $\rho > 0$ is a penalty parameter.

The alternating direction method of multipliers for Model (2.10) is derived by:

$$\begin{aligned} X_{k+1} &:= \arg\min_{X\geq \mathbf{0}} \ \mathcal{L}(X, Z_k, \Lambda_k), && \text{(2.11a)}\\ Z_{k+1} &:= \arg\min \ \mathcal{L}(X_{k+1}, Z, \Lambda_k), && \text{(2.11b)}\\ \Lambda_{k+1} &:= \Lambda_k + \gamma\rho(X_{k+1} - Z_{k+1}). && \text{(2.11c)} \end{aligned}$$

By rearranging the terms of (2.11a), it is equivalent to

$$\begin{aligned} \min_{X\in\mathbb{R}^{m\times n}} \ & \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2 + \frac{\rho}{2}\|X - (Z_k - \frac{1}{\rho}\Lambda_k)\|_F^2 \\ \text{s.t.} \ \ & X \geq \mathbf{0}. \end{aligned} \tag{2.12}$$

Model (2.12) can be split into two subproblems:

$$\begin{aligned} \min \ & \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2 + \frac{\rho}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(Z_k - \frac{1}{\rho}\Lambda_k)\|_F^2 \\ \text{s.t.} \ \ & \mathcal{P}_\Omega(X) \geq \mathbf{0}, \end{aligned}$$

and

$$\begin{aligned} \min \ & \|\mathcal{P}_{\hat{\Omega}}(X) - \mathcal{P}_{\hat{\Omega}}(Z_k - \frac{1}{\rho}\Lambda_k)\|_F^2 \\ \text{s.t.} \ \ & \mathcal{P}_{\hat{\Omega}}(X) \geq \mathbf{0}, \end{aligned}$$

where $\hat{\Omega}$ is the complement of Ω. Finally, we can get the solution of (2.11a) by solving the above two subproblems.

$$\begin{aligned} (X_{k+1})_\Omega =& \mathcal{P}_+(\frac{1}{\rho+1}\mathcal{P}_\Omega(M + \rho E_k)),\\ (X_{k+1})_{\hat{\Omega}} =& \mathcal{P}_+(\mathcal{P}_{\hat{\Omega}}(E_k)), \end{aligned}$$

where $E_k = Z_k - \frac{1}{\rho}\Lambda_k$.

By deleting the constant terms of (2.11b), we can get a more concise form:

$$Z_{k+1} := \arg\min \ \mu\|Z\|_* + \frac{\rho}{2}\|Z - (X_{k+1} + \frac{1}{\rho}\Lambda_k)\|_F^2,$$

whose solution is

$$Z_{k+1} = S_{\frac{\mu}{\rho}}(X_{k+1} + \frac{1}{\rho}\Lambda_k).$$

In short, ADMM applied to Model (2.10) produces the iteration:

$$\begin{aligned} (X_{k+1})_\Omega &:= \mathcal{P}_+(\frac{1}{\rho+1}\mathcal{P}_\Omega(M + \rho E_k)),\ (X_{k+1})_{\hat{\Omega}} := \mathcal{P}_+(\mathcal{P}_{\hat{\Omega}}(E_k)), && \text{(2.13a)}\\ Z_{k+1} &:= S_{\frac{\mu}{\rho}}(X_{k+1} + \frac{1}{\rho}\Lambda_k), && \text{(2.13b)}\\ \Lambda_{k+1} &:= \Lambda_k + \gamma\rho(X_{k+1} - Z_{k+1}). && \text{(2.13c)} \end{aligned}$$

From the above considerations, we arrive at Algorithms 2 below.

Algorithm 2: Another exact ADMM-based algorithm of NMC

1 Input $\mathcal{P}_\Omega(M)$, $I_m \geq 0$, and $tol \geq 0$.
2 Set μ, γ, $\rho \geq 0$. Set X_0 and Z_0 as random matrices, Λ_0 as zero matrices of appropriate sizes.
3 **while** *not converge* **do**
4 update X_k, Z_k, Λ_k by the formulas (2.13).

3 Numerical Results

In this section, we report on the application of our proposed ADMM-based algorithms to a series of matrix problems to demonstrate their ability. To illustrate the performance of our algorithmic approaches combined with different procedures, we test the following three solvers.

1. ADMM-TayA. Algorithm 2 in [16].

2. ADMM-TwoV. Algorithm 1, the exact ADMM-based method for Model (2.4) with two new matrix splitting variables.

3. ADMM-OneV. Algorithm 2, the exact ADMM-based method for Model (2.4) with only one new matrix splitting variable.

We implement our algorithms in MATLAB. All the experiments are performed on a Dell Precision T5500 workstation with Intel Xenon(R) E5620 CPU at 2.40GHz (×4) and 12G of memory running Ubuntu 12.04 and MATLAB 2011b.

3.1 Implementation, parameters of the three solvers

We test the above three solvers on random nonnegative matrix problems. We do numerical experiments by the following procedure: firstly, we create a low-rank or approximately low-rank nonnegative matrix $M \in \mathbb{R}_+^{m\times n}$, with i.i.d. Gaussian entries. Secondly, we select a subset of p elements uniformly at random from the $m \times n$ elements of M and denote their index set as Ω. The index set of unknown elements is denoted as $\hat{\Omega}$. The ratio $p/(mn)$ between the number of measurements and the number of entries in the matrix is denoted as "SR" (sampling ratio). We use $\mathcal{P}_\Omega(M)$ to complete X, the result is denoted as $\hat{X}$. Finally, we will check the differences between $\hat{X}$ with its actual value M.

$$err1 = \|\mathcal{P}_\Omega(\hat{X}) - \mathcal{P}_\Omega(M)\|_F/|\Omega|, \tag{3.1a}$$

$$err2 = \|\mathcal{P}_{\hat{\Omega}}(\hat{X}) - \mathcal{P}_{\hat{\Omega}}(M)\|_F/|\,\hat{\Omega}|, \tag{3.1b}$$

$$err = err1 + err2, \tag{3.1c}$$

where $|\Omega|$ and $|\hat{\Omega}|$ are the cardinality of Ω and $\hat{\Omega}$, respectively.

The most important algorithmic parameters in Algorithm 1-2 are μ, γ, tol, α, β, ρ, and the maximal number of iterations I_m. In our implementation, we set $\gamma = 1.618$, $tol = 10^{-4}$, $\alpha = \beta = \rho = 0.1$, and $I_m = 10^5$. However, the value of μ is very difficult to set since it can be neither too large nor too small. μ is usually chosen to be a moderate value. Three sets of results are computed when μ is equal to three different values. They are shown in Table 1.

We can use continuation technique employed in [8, 11, 13] to accelerate the convergence of ADMM-TwoV, ADMM-OneV, and ADMM-FBS. If Model (2.4) is to be solved with the target parameter value $\mu = \bar{\mu}$, we propose solving a sequence of Model (2.4) by a decreasing

Table 1: Numerical Results of different values of μ (p=100,n=100,r=3)

Problems	ADMM-TwoV				ADMM-OneV			
SR	err1	err2	err	r	err1	err2	err	r
$\mu = 10$								
0.30	6.44e-03	4.50e-03	1.09e-02	88	6.43e-03	4.50e-03	1.09e-02	1
0.40	4.31e-03	3.65e-03	7.96e-03	31	4.31e-03	3.65e-03	7.96e-03	1
0.50	3.20e-03	3.31e-03	6.51e-03	1	3.20e-03	3.31e-03	6.51e-03	1
0.60	2.55e-03	3.20e-03	5.75e-03	1	2.55e-03	3.20e-03	5.75e-03	1
0.70	2.13e-03	3.36e-03	5.49e-03	1	2.13e-03	3.36e-03	5.49e-03	1
0.80	1.84e-03	3.75e-03	5.59e-03	1	1.84e-03	3.75e-03	5.59e-03	1
0.90	1.63e-03	5.01e-03	6.64e-03	1	1.63e-03	5.01e-03	6.64e-03	1
$\mu = 1$								
0.30	1.52e-03	1.76e-03	3.28e-03	100	1.08e-03	8.47e-04	1.93e-03	3
0.40	7.47e-04	7.08e-04	1.46e-03	100	7.02e-04	6.54e-04	1.36e-03	3
0.50	5.05e-04	5.78e-04	1.08e-03	96	5.03e-04	5.76e-04	1.08e-03	3
0.60	3.79e-04	5.16e-04	8.95e-04	62	3.79e-04	5.16e-04	8.95e-04	3
0.70	3.00e-04	5.10e-04	8.11e-04	5	3.00e-04	5.10e-04	8.11e-04	3
0.80	2.43e-04	5.22e-04	7.66e-04	3	2.43e-04	5.22e-04	7.66e-04	3
0.90	2.03e-04	6.53e-04	8.57e-04	3	2.03e-04	6.53e-04	8.57e-04	3
$\mu = 10^{-1}$								
0.30	2.55e-04	8.76e-03	9.02e-03	100	1.17e-04	1.05e-04	2.21e-04	3
0.40	1.88e-04	9.03e-03	9.22e-03	99	7.25e-05	7.19e-05	1.44e-04	3
0.50	1.48e-04	9.29e-03	9.44e-03	100	5.13e-05	6.13e-05	1.13e-04	3
0.60	1.21e-04	9.63e-03	9.75e-03	100	3.83e-05	5.35e-05	9.17e-05	3
0.70	1.00e-04	1.01e-02	1.02e-02	100	3.02e-05	5.23e-05	8.25e-05	3
0.80	8.37e-05	1.03e-02	1.04e-02	100	2.44e-05	5.29e-05	7.73e-05	3
0.90	6.00e-05	9.79e-03	9.85e-03	100	2.04e-05	6.60e-05	8.63e-05	3

sequence $\{\mu^0, \mu^1, \ldots\}$. When a new problem, associated with μ^{j+1} is to be solved, the approximate solution for the current problem with $\mu = \mu^j$ is used as the starting point. In our numerical experiments in Sections 3.2 and 3.3, we set the initial $\mu^0 = 1$, and update $\mu^k = \mu^{k-1}/1.001$ at iteration k. A stopping criterion of Algorithm 1 is met as long as all the following four conditions are satisfied:

$$\|X_k - X_{k-1}\|_F \leq tol, \tag{3.2a}$$
$$\|X_k - Y_k\|_F \leq tol, \tag{3.2b}$$
$$\|X_k - Z_k\|_F \leq tol, \tag{3.2c}$$
$$\|\mathcal{P}_\Omega(X_k) - \mathcal{P}_\Omega(M)\|_F \leq tol. \tag{3.2d}$$

A stopping criterion of Algorithm 2 is met as long as all the following three conditions are satisfied:

$$\|X_k - X_{k-1}\|_F \leq tol, \tag{3.3a}$$
$$\|X_k - Z_k\|_F \leq tol, \tag{3.3b}$$
$$\|\mathcal{P}_\Omega(X_k) - \mathcal{P}_\Omega(M)\|_F \leq tol. \tag{3.3c}$$

3.2 Experiments on random matrix completion problems

The matrix $M \in \mathbb{R}_+^{m\times n}$ with rank r in this subsection is created randomly by the following procedure (see also [11, 15]): two random nonnegative matrices $M_L \in \mathbb{R}_+^{m\times r}$, $M_R \in R_+^{r\times n}$ with i.i.d. standard Gaussian entries are first generated, then $M = M_L M_R$ is assembled. From the creation process of M, its rank is no larger than r.

Table 2: Numerical results on medium randomly created matrix completion problems (p=100, n=100, r=3)

	ADMM-TayA				ADMM-TwoV				ADMM-OneV			
SR	err1	err2	err	r	err1	err2	err	r	err1	err2	err.X	r
0.30	1.83e-03	1.37e-03	3.19e-03	100	1.63e-03	2.02e-03	3.65e-03	100	1.10e-03	8.63e-04	1.96e-03	3
0.40	1.20e-03	1.10e-03	2.30e-03	100	8.04e-04	7.94e-04	1.60e-03	100	7.10e-04	6.75e-04	1.39e-03	3
0.50	8.54e-04	9.50e-04	1.80e-03	100	5.07e-04	5.78e-04	1.09e-03	97	5.03e-04	5.71e-04	1.07e-03	3
0.60	6.47e-04	8.74e-04	1.52e-03	100	3.80e-04	5.21e-04	9.01e-04	73	3.80e-04	5.21e-04	9.01e-04	3
0.70	5.12e-04	8.58e-04	1.37e-03	100	3.00e-04	5.09e-04	8.08e-04	5	3.00e-04	5.09e-04	8.08e-04	3
0.80	4.17e-04	9.13e-04	1.33e-03	100	2.44e-04	5.40e-04	7.84e-04	3	2.44e-04	5.40e-04	7.84e-04	3
0.90	3.49e-04	1.15e-03	1.50e-03	100	2.04e-04	6.77e-04	8.81e-04	3	2.04e-04	6.77e-04	8.81e-04	3

The computational results of matrix completion are presented in Table 2. We can observe that all our proposed solvers perform better than ADMM-TayA. ADMM-TwoV converges more slowly than ADMM-OneV. The reason is that there are two unknown penalty parameters α and β in ADMM-TwoV, whose values are difficult to set. When $SR > 0.5$, numerical results of our proposed solvers are similar. All of them can output satisfactory results.

3.3 Experiments on random low-rank approximation problems

We next consider applying our proposed algorithms to randomly generated low-rank approximation problems. The goal is to find a low-rank approximation to a mathematically full-rank matrix X_0 whose singular values gradually tend to zero, though none is exactly zero.

In this subsection, $M \in \mathbb{R}_+^{m\times n}$ with rank r is created as follows: two matrices $M_L \in \mathbb{R}_+^{m\times r}$ and $M_R \in \mathbb{R}_+^{r\times n}$ with i.i.d. standard Gaussian entries are first generated randomly. Then the matrix $M = M_L S M_R$ is assembled, here $S \in \mathbb{R}^{r\times r}$ is a diagonal matrix satisfying: $\mathrm{diag}(S) = (1, 2, \ldots, r)^\top$. Some numerical results are gotten when SR increases from 0.3 to 0.9, they are shown in Table 3. From Table 3, we can get similar conclusion as above subsection. In conclusion, recovering results show our proposed algorithms are helpful for NMC.

Table 3: Numerical results on medium randomly created low-rank approximation problems (p=100,n=100,r=5)

	ADMM-TayA				ADMM-TwoV				ADMM-OneV			
SR	err1	err2	err	r	err1	err2	err	r	err1	err2	err.X	r
0.30	2.47e-03	2.14e-03	4.61e-03	15	2.37e-03	3.09e-02	3.33e-02	100	1.49e-03	1.37e-03	2.87e-03	5
0.40	1.60e-03	1.69e-03	3.29e-03	94	1.78e-03	2.73e-02	2.91e-02	100	9.53e-04	1.05e-03	2.00e-03	5
0.50	1.12e-03	1.37e-03	2.49e-03	100	1.38e-03	2.18e-02	2.32e-02	100	6.63e-04	8.26e-04	1.49e-03	5
0.60	8.52e-04	1.25e-03	2.10e-03	100	1.05e-03	1.49e-02	1.60e-02	100	5.00e-04	7.42e-04	1.24e-03	5
0.70	6.71e-04	1.21e-03	1.88e-03	100	6.27e-04	5.51e-03	6.13e-03	100	3.93e-04	7.17e-04	1.11e-03	5
0.80	5.44e-04	1.26e-03	1.81e-03	100	3.23e-04	7.73e-04	1.10e-03	98	3.18e-04	7.43e-04	1.06e-03	5
0.90	4.54e-04	1.63e-03	2.08e-03	100	2.65e-04	9.59e-04	1.22e-03	6	2.65e-04	9.59e-04	1.22e-03	5

Acknowledgements

We would like to thank Prof. Zaiwen Wen for the discussions on matrix completion. We thank Prof. Bingsheng He for the discussions on the alternating direction method of multipliers. The authors are grateful to two anonymous referees for their detailed and valuable comments and suggestions, Algorithm 2 is suggested by one of the referees.

References

[1] J. Cai, E.J. Candés and Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.* 20 (2010) 1956–1982.

[2] E.J. Candès and B. Recht, Exact matrix completion via convex optimization, *Foundations of Computational Mathematics* 9 (2009) 717–772.

[3] E.J. Candès and T. Tao, The power of convex relaxation: near-optimal matrix completion, *IEEE Trans. Inform. Theory* 56 (2010) 2053–2080.

[4] W. Deng and W. Yin, On the global and linear convergence of the generalized alternating direction method of multipliers, Technical report, Rice University CAAM, 2012.

[5] M. Fazel, *Matrix rank minimization with application*, PhD thesis, Stanford University, 2002.

[6] M. Fortin and R. Glowinski, *Augmented Lagrangian Methods*, volume 15 of *Studies in Mathematics and its Applications*, North-Holland Publishing Co., Amsterdam, 1983. Applications to the numerical solution of boundary value problems, Translated from the French by B. Hunt and D. C. Spicer.

[7] D. Goldfarb, S. Ma, and Z. Wen, Solving low-rank matrix completion problems efficiently, in *Proceedings of the 47th annual Allerton conference on Communication, control, and computing*, Allerton'09, 2009, pp. 1013–1020.

[8] E.T. Hale, W. Yin, and Y. Zhang, Fixed-point continuation for l_1-minimization: methodology and convergence, *SIAM J. Optim.* 19 (2008) 1107–1130.

[9] B. He, H. Yang and S. Wang, Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities, *J. Optim. Theory Appl.* 106 (2000) 337–356.

[10] H. Ji, C. Liu, Z. Shen and Y. Xu, Robust video denoising using low rank matrix completion, in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE*, 2010, pp. 1791–1798.

[11] S. Ma, D. Goldfarb and L. Chen, Fixed point and bregman iterative methods for matrix rank minimization, *Mathematical Programming* 128 (2011) 321–353.

[12] B. Recht, M. Fazel and P. A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, *SIAM Rev.* 52 (2010) 471–501.

[13] K.C. Toh and S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems, *Pacific J. Optim.* 6 (2010) 615–640.

[14] Z. Wen, D. Goldfarb,and W. Yin, Alternating direction augmented lagrangian methods for semidefinite programming, *Math. Prog. Comp.* 2 (2010) 203–230.

[15] Z. Wen, W. Yin and Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm, *Math. Prog. Comp.* 4 (2012) 333–361.

[16] F. Xu, C. Lin, G. He and Z. Wen, Nonnegative matrix completion for life-cycle assessment and input-output analysis, Working paper of The Center for Economic Research Shandong University, 2013.

[17] Y. Xu, W. Yin, Z. Wen and Y. Zhang, An alternating direction algorithm for matrix completion with nonnegative factors, *Frontiers of Mathematics in China* 7 (2012) 365–384.

Manuscript received 21 September 2013
revised 7 April 2014
accepted for publication 17 April 2014

Fangfang Xu
Department of Mathematics, Shanghai Jiao Tong University
Shanghai 200240, China
E-mail address: xufangfang1984@sjtu.edu.cn

Guoping He
College of Mathematics and Systems Science
Shandong University of Science and Technology
Qingdao 266590, China
E-mail address: hegp@263.net