Factor Matrix Nuclear Norm Minimization for Low-Rank Tensor Completion

Yuanyuan Liu* Fanhua Shang† Hong Cheng* James Cheng† Hanghang Tong‡

Abstract

Most existing low-n-rank minimization algorithms for tensor completion suffer from high computational cost due to involving multiple singular value decompositions (SVDs) at each iteration. To address this issue, we propose a novel factor matrix rank minimization method for tensor completion problems. Based on the CANDECOMP/PARAFAC (CP) decomposition, we first formulate a factor matrix rank minimization model by deducing the relation between the rank of each factor matrix and the mode-n rank of a tensor. Then, we introduce a tractable relaxation of our rank function, which leads to a convex combination problem of much smaller scale matrix nuclear norm minimization. Finally, we develop an efficient alternating direction method of multipliers (ADMM) scheme to solve the proposed problem. The experimental results on both synthetic and real-world data validate the effectiveness of our approach. Moreover, our method is significantly faster than the state-of-the-art approaches and scales well to handle large datasets.

1 Introduction

Multi-way data analysis is an important topic in signal processing [7], numerical linear algebra [6], computer vision [19], [20], data mining [16], [29], machine learning [27], neuroscience [22], and so on. As a generalization of scalars, vectors (first-order tensors) and matrices (second-order tensors), higher-order tensors are becoming increasingly popular. However, the values of their entries may be missing due to problems in the acquisition process, e.g., loss of information or cost of experiments to obtain complete data too high. In this paper, we study the *tensor completion problem*, which is to estimate the missing values in the tensor. The tensor completion problem has been successfully applied to a wide range of real-world problems, such as visual data [19], [20], EEG data [1], [22], and hyperspectral data analysis [10], social network analysis [29] and link prediction [33].

In recent years, sparse vector recovery and low-rank matrix completion problems have been intensively studied. Although the l_0-norm and the rank minimization have been proven to be strong global constraints and good measures of sparsity, the optimization problem involving the l_0-norm or the rank minimization is NP-hard in general due to their discrete nature. The l_1-norm and the nuclear norm (also known as the trace norm) are widely used to approximate the l_0-norm and the rank of a matrix, and the resulting problems are convex optimization problems. In addition, some researchers [4], [5], [24] have provided theoretical guarantee that the task of the l_0-norm minimization or the rank minimization problem can be accomplished via solving the l_1-norm or the nuclear norm minimization under some reasonable conditions. In fact, the l_1-norm and the nuclear norm have been shown to be the tightest convex surrogates to the l_0-norm and the rank function, respectively [5], [8].

As the generalization of sparse vector recovery and low-rank matrix completion, the low-rank tensor recovery problem has drawn lots of attention from researchers in the past several years [15]. Compared with vectors and matrices, a tensor can be used to express more complicated intrinsic structures of higher-order data. In this paper we are particularly interested in the low-rank tensor completion problem, which is to find a tensor $\mathcal{X}$ of the (nearly) lowest rank from a subset of the entries of the tensor $\mathcal{T}$ such that $\mathcal{X}_\Omega = \mathcal{T}_\Omega$, where $\mathcal{X}_\Omega$ denotes the restriction of the tensor on the known entries given by Ω. Liu et al. [19] indicated that tensor completion utilizes all information along all dimensions, while matrix completion only considers the constraints along two particular dimensions.

Recently, several works [10], [19], [20], [27], [28] extended the framework of trace norm regularization to the estimation of a partially observed low-rank tensor. Liu et al. [19] first introduced an extension of the trace norm to the tensor completion problem. In a subsequent paper [20], they defined the trace norm of a tensor as a convex combination of trace norms of its unfolded matrices. There are some similar tensor completion methods in [10], [27]. As a result, the tensor completion problem is converted into a convex combination of the nuclear

*Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong. {yyliu, hcheng}@se.cuhk.edu.hk

†Department of Computer Science and Engineering, The Chinese University of Hong Kong. {fhshang, jcheng}@cse.cuhk.edu.hk

‡Computer Science Department, City College, The City University of New York. tong@cs.ccny.cuny.edu

norm minimization of the involved matrices. However, the tensor nuclear norm minimization (TNNM) problem has to be solved iteratively and involves multiple SVDs at each iteration. Therefore, all these algorithms for TNNM suffer from high computational cost of multiple SVDs.

Tensor decomposition is a type of classical higher-order data analysis method and gives a concise representation of the underlying structure of the tensor, revealing that the tensor data can be modeled as lying close to a low-dimensional subspace. Two most popular tensor decomposition methods, the Tucker decomposition [9], [31] and the CANDECOMP/PARAFAC (CP) decomposition [1], [11], are also used to solve tensor completion problems. However, those tensor decomposition models do require the ability to reliably estimate the rank of the involved tensor.

In this paper, we focus on addressing two issues for the low-rank tensor completion problem, i.e., the robustness of the given tensor rank and the computational cost. We propose a novel factor matrix nuclear norm minimization (NNCP) method based on the CP decomposition for tensor completion problems. The experimental results on both synthetic and real-world data validate both the efficiency and effectiveness of our NNCP approach. The main contributions of this paper are as follows:

1. We deduce the relation of the rank of between each factor matrix and the corresponding unfolded matrix of a tensor, and formulate a factor matrix rank minimization model.

2. We introduce a tractable relaxation of the rank function into our factor matrix rank minimization model, and then obtain a convex combination problem of much smaller factor matrix nuclear norm minimization.

3. We present an efficient alternating direction method of multipliers (ADMM) scheme to solve the proposed problem.

2 Notions and Related Work

Before reviewing previous work, we first introduce some basic tensor notions and terminologies. A matrix is denoted by an upper-case letter, e.g., X, the entries as lower-case letters, e.g., x_{ij}. An N-order tensor is denoted by a calligraphic letter, e.g., $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$, and its entries are denoted by $x_{i_1, i_2 \cdots, i_N}$. The order N of a tensor is the number of dimensions, also known as ways or modes. Fibers are the higher-order analogue of matrix rows and columns. A fiber is defined by fixing every index but one. The mode-n fibers are all vectors $\mathrm{x}_{i_1, \cdots, i_{n-1}, i_{n+1}, \cdots, i_N}$ that are obtained by fixing the values of $\{i_1, \cdots, i_N\} \backslash i_n$.

The mode-n unfolding, also known as matricization, of an N-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ is denoted by $X_{(n)} \in \mathbb{R}^{I_n \times \Pi_{j \neq n} I_j}$, which is a rearrangement of the entries of $\mathcal{X}$ into the matrix $X_{(n)}$ such that the mode-n fiber becomes the row index and all other $(N-1)$ modes become column indices in lexicographical order. The tensor element $(i_1, i_2, \cdots, i_N)$ is mapped to the matrix element (i_n, j), where

$$j = 1 + \textstyle\sum_{k=1, k \neq n}^{N} (i_k - 1) J_k \text{ with } J_k = \prod_{m=1, m \neq n}^{k-1} I_m.$$

The inner product of two same-sized tensors $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ and $\mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ is the sum of the product of their entries, $< \mathcal{A}, \mathcal{B} >= \Sigma_{i_1, \cdots i_N} a_{i_1, \cdots i_N} b_{i_1, \cdots i_N}$. The Frobenius norm of an N-order $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ is defined as:

$$\|\mathcal{X}\|_F := \sqrt{\textstyle\sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} x_{i_1, \cdots, i_N}^2}.$$

Let A and B be two matrices of size $m \times n$ and $p \times q$ respectively. The Kronecker product of two matrices A and B, denoted by $A \otimes B$, is an $mp \times nq$ matrix given by: $A \otimes B = [a_{ij} B]_{mp \times nq}$.

Let $A = [\mathrm{a}_1 \, \mathrm{a}_2 \, \cdots \, \mathrm{a}_r]$ and $B = [\mathrm{b}_1 \, \mathrm{b}_2 \, \cdots \, \mathrm{b}_r]$ be two column matrices of size $m \times r$ and $n \times r$ respectively. Then the Khatri-Rao product of two matrices A and B is defined as the column-wise Kronecker product and represented by $\odot$:

$$A \odot B = [\mathrm{a}_1 \otimes \mathrm{b}_1 \; \mathrm{a}_2 \otimes \mathrm{b}_2 \; \cdots \; \mathrm{a}_r \otimes \mathrm{b}_r].$$

2.1 Low-n-Rank Tensor Estimation For the tensor completion problem, Liu et al. [19] and Signoretto et al. [28] have proposed an extension of low-rank matrix completion concept to tensor data. With an exact analogue to the definition of the matrix rank, the rank of a tensor $\mathcal{X}$, denoted by rank($\mathcal{X}$), is defined as follows.

Definition 2.1. *The rank of a tensor is the smallest number of rank-one tensors that generate the tensor as their sum, i.e., the smallest R such that*

$$\mathcal{X} = \textstyle\sum_{i=1}^{R} \mathrm{a}_i^1 \circ \mathrm{a}_i^2 \circ \cdots \circ \mathrm{a}_i^N,$$

where $\circ$ denotes the outer product of some vectors, i.e., $(\mathrm{a}_i^1 \circ \mathrm{a}_i^2 \circ \cdots \circ \mathrm{a}_i^N)_{i_1, i_2 \cdots, i_N} = [\mathrm{a}_i^1]_{i_1} [\mathrm{a}_i^2]_{i_2} \cdots [\mathrm{a}_i^N]_{i_N}$.

This definition of the rank of a tensor is an extension of the rank of a matrix, but with different properties. A key difference is that the rank of a tensor is difficult to handle, as there is no straightforward way to determine the rank of a given tensor. In fact, the problem is NP-hard [13], [15], [17]. Fortunately, the n-rank of a tensor $\mathcal{X}$, denoted by n-rank($\mathcal{X}$), is easy to compute, which consists of the matrix ranks of the mode-n matricization of the tensor.

Definition 2.2. *The n-rank of an N-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ is the tuple of the ranks of the mode-n unfolding,*

$$n\text{-}rank(\mathcal{X}) = (rank(X_{(1)}), rank(X_{(2)}), \cdots, rank(X_{(N)})),$$

where $rank(X_{(n)})$ is called the mode-n rank.

Liu et al. [20] and Gandy et al. [10] have proposed to minimize the weighted tensor n-rank model for tensor completion problems as follows:

$$\min_{\mathcal{X}} \sum_{n=1}^{N} w_n \text{rank}(X_{(n)}), \quad \text{s.t.}, \mathcal{X}_\Omega = \mathcal{T}_\Omega, \tag{2.1}$$

where w_n's are pre-specified weights, and $X_{(n)}$ denotes the unfolded matrix along the n-th mode on the tensor $\mathcal{X}$. And the entries of $\mathcal{T}$ in the set Ω are given while the remaining elements are missing. In order to keep things simple, the weighted sum of the ranks of the different unfolded matrices is used to take the place of the n-rank of the involved tensor. In addition, Gandy et al. [10] have presented an unweighted model, i.e., a special case of the above model (2.1), where $w_n = 1$, $n = 1, \ldots, N$.

The non-convex optimization problem (2.1) can be solved by convex relaxation replacing the rank of the matrix with the nuclear norm as follows:

$$\min_{\mathcal{X}} \sum_{n=1}^{N} w_n \|X_{(n)}\|_*, \quad \text{s.t.}, \mathcal{X}_\Omega = \mathcal{T}_\Omega, \tag{2.2}$$

where the nuclear norm of the tensor $\mathcal{X}$ is defined as

$$\|\mathcal{X}\|_* = \sum_{n=1}^{N} w_n \|X_{(n)}\|_*, \tag{2.3}$$

and $\|X_{(n)}\|_*$ denotes the nuclear norm of the matrix $X_{(n)}$, i.e., the sum of its singular values. In the presence of noise, we obtain a corresponding unconstrained formulation:

$$\min_{\mathcal{X}} \sum_{n=1}^{N} w_n \|X_{(n)}\|_* + \frac{\lambda}{2} \|\mathcal{X}_\Omega - \mathcal{T}_\Omega\|_F^2, \tag{2.4}$$

where $\lambda > 0$ is a regularization parameter.

In fact, each mode-n unfolded matrix $X_{(n)}$ shares the same entries and cannot be optimized independently. Recently, some nuclear norm minimization algorithms have been proposed to successfully address the visual data and hyperspectral data recovery problems [20], [10]. However, all those algorithms have to be solved iteratively and involve multiple SVDs at each iteration, thus suffering from high computational cost. Furthermore, many additional variables are introduced to split these interdependent terms in (2.2) and (2.4) such that they can be solved independently. As a result, all those algorithms for the convex combination of multiple nuclear norm minimization problems suffer from high computational cost.

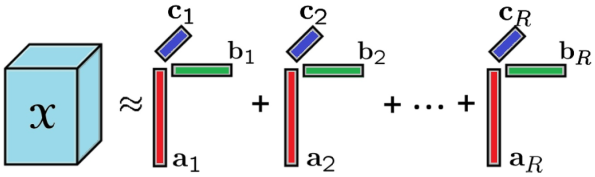


Figure 1: Illustration of an R-component CP model for a three-order tensor.

2.2 Tensor Decompositions for Completion

Next, we will introduce two tensor decomposition models for tensor completion problems. In [1], Acar et al. presented a weighted least squares model for three-order tensor decomposition problems with missing entries:

$$\min_{U_1, \cdots, U_N} \|\mathcal{W} * (\mathcal{T} - U_1 \circ U_2 \cdots \circ U_N)\|_F^2, \tag{2.5}$$

where the symbol $*$ denotes the Hadamard (elementwise) product, $U_1 \circ U_2 \cdots \circ U_N$ is the CP model based decomposition, and $U_n \in \mathbb{R}^{I_n \times R}$ are referred to as the factor matrices which are the combination of the vectors from the rank-one components (e.g., as shown in Fig. 1, $U_1 = [\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_R]$), and R is a positive integer. Moreover, $\mathcal{W}$ is a nonnegative weight tensor with the same size as $\mathcal{T}$,

$$w_{i_1, i_2 \cdots, i_N} = \begin{cases} 1 & \text{if } t_{i_1, i_2 \cdots, i_N} \text{ is known}, \\ 0 & \text{otherwise}. \end{cases}$$

In [9], the Tucker model is formulated as follows:

$$\min_{U_n, \mathcal{C}} \|\mathcal{W} * (\mathcal{T} - \mathcal{C} \times_1 U_1 \times_2 \cdots \times_N U_N)\|_F^2, \tag{2.6}$$

where $U_n \in \mathbb{R}^{I_n \times R_n}$, $\mathcal{C} \in \mathbb{R}^{R_1 \times R_2 \cdots R_N}$ is a core tensor with the given ranks $(R_1, R_2, \cdots, R_N)$, and $\times_n$ denotes the n-mode matrix product (see [15]).

Recently, some extensions of both basic decomposition models and corresponding algorithms were developed for tensor estimation problems. However, to solve the tensor approximation and completion problems, a suitable rank value needs to be given, and it was shown that both the tucker and CP models have a poor performance while real-world data have a high rank [20].

3 Factor Matrix Nuclear Norm Minimization Model

The major bottleneck of existing tensor completion algorithms for solving the optimization problems (2.2) and (2.4) is the high computational cost of multiple SVDs on the large matrices at each iteration. Motivated by this, we formulate a novel factor matrix rank minimization model by deducing the relation between the rank of each factor matrix and the mode-n rank of a tensor. In other words, our model is to minimize the n-rank of factor matrices of the CP decomposition instead of low-n-rank of the involved tensor.

3.1 The Rank of Factor Matrices Let $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ be an N-order low-rank tensor with tensor rank r, then the CP model of the tensor $\mathcal{X}$ is written as follows:

$$\mathcal{X} = U_1 \circ U_2 \cdots \circ U_N = \sum_{i=1}^{r} \mathrm{u}_1^{(i)} \circ \mathrm{u}_2^{(i)} \cdots \circ \mathrm{u}_N^{(i)}, \tag{3.7}$$

where $U_n = [\mathrm{u}_n^{(1)}\, \mathrm{u}_n^{(2)} \cdots \mathrm{u}_n^{(r)}] \in \mathbb{R}^{I_n \times r}$, $n = 1, \cdots, N$ denote the factor matrices of the tensor $\mathcal{X}$.

Theorem 3.1. *Let $X_{(n)}$, $n = 1, \cdots, N$ be the unfolded matrices of the tensor $\mathcal{X}$ with the rank r, and U_n, $n = 1, \cdots, N$ be the factor matrices. Then*

$$rank(X_{(n)}) \leq rank(U_n), \quad n = 1, \cdots, N. \tag{3.8}$$

Proof. Since $\mathcal{X} = U_1 \circ U_2 \cdots \circ U_N$, we have

$$\begin{aligned} X_{(n)} = U_n(U_N \odot \cdots U_{n+1} \odot U_{n-1} \odot \cdots U_1)^T, \\ n = 1, \cdots, N. \end{aligned} \tag{3.9}$$

Thus

$$\begin{aligned} \mathrm{rank}(X_{(n)}) &= \mathrm{rank}(U_n(U_N \odot \cdots U_{n+1} \odot U_{n-1} \odot \cdots U_1)^T) \\ &\leq \mathrm{rank}(U_n), \quad n = 1, \cdots, N. \end{aligned}$$

From the above theorem, it is clear that the factor matrices $U_n \in \mathbb{R}^{I_n \times r}$, $n = 1, \cdots, N$ have a much smaller size than the mode-n unfolded matrices $X_{(n)} \in \mathbb{R}^{I_n \times \Pi_{j \neq n} I_j}$, while the rank of each factor matrix is an upper bound on the rank of the corresponding unfolded matrix. In the following subsection, we will propose a new model that uses the rank of the factor matrices U_n, i.e., rank(U_n), instead of the mode-n rank of the tensor in (2.1), i.e., rank$(X_{(n)})$.

3.2 Our Model Suppose that the unknown tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ is low rank, our rank minimization model based on the CP decomposition for tensor completion can be expressed as follows:

$$\begin{aligned} &\min_{\mathcal{X}, U_n} \sum_{n=1}^{N} w_n \mathrm{rank}(U_n), \\ &\text{s.t.}, \ \mathcal{X} = U_1 \circ \cdots \circ U_N, \ \mathcal{X}_\Omega = \mathcal{T}_\Omega, \end{aligned} \tag{3.10}$$

where $\mathcal{X}$ and $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$, $U \in \mathbb{R}^{I_n \times R}$, and R denotes a upper bound of the tensor rank and is a positive integer. Moreover, the factor matrix rank minimization is a relaxation form of the mode-n rank minimization of the tensor along each mode. In addition, several researchers [14] have provided some tensor rank estimation strategies to compute a good value r for the rank of the tensor. Thus, we only set a relatively large integer R such that $R \geq r$.

Due to the discrete nature of the rank, the above model (3.10) can be relaxed by replacing the rank function with the nuclear norm as follows:

$$\begin{aligned} &\min_{\mathcal{X}, U_n} \sum_{n=1}^{N} w_n \| U_n \|_*, \\ &\text{s.t.}, \ \mathcal{X} = U_1 \circ \cdots \circ U_N, \ \mathcal{X}_\Omega = \mathcal{T}_\Omega. \end{aligned} \tag{3.11}$$

Furthermore, the relaxed version is formulated by relaxing the CP decomposition constraint term in (3.11):

$$\begin{aligned} &\min_{\mathcal{X}, U_n} \sum_{n=1}^{N} w_n \| U_n \|_* + \frac{\lambda}{2} \| \mathcal{X} - U_1 \circ \cdots \circ U_N \|_F^2, \\ &\text{s.t.}, \ \mathcal{X}_\Omega = \mathcal{T}_\Omega. \end{aligned} \tag{3.12}$$

The above model (3.12) is called as a factor matrix nuclear norm minimization (NNCP) approach based on the CP decomposition. We only need to perform SVD in some much smaller scale involved factor matrices, and hence the NNCP algorithm we present in Section 4 is very efficient. Meanwhile, the performance of NNCP is much more robust to the given tensor rank R than the Tucker and the CP methods, which will be verified by our experiments.

4 NNCP Algorithm

Recently, it has been shown in [2], [32], [21] that the alternating direction method of multipliers (ADMM) is very efficient for some convex or non-convex programming problems from various applications. We also refer to [18], [20], [26] for some recently exploited applications of ADMM. In this paper we also propose an ADMM scheme for solving the proposed model (3.12), and the problem (3.12) is reformulated into the following equivalent form:

$$\begin{aligned}&\min_{\mathcal{X},U_n,M_n}\sum_{n=1}^{N}w_n\|M_n\|_*+\frac{\lambda}{2}\|\mathcal{X}-U_1\cdots\circ U_N\|_F^2,\\&\text{s.t.}, \mathcal{X}_\Omega=\mathcal{T}_\Omega,\ M_n=U_n,\ n=1,\cdots,N,\end{aligned}\tag{4.13}$$

where $M_n,\ n=1,\cdots,N$ are auxiliary variables.

4.1 Solving Scheme. The partial augmented Lagrangian function for (4.13) is given by:

$$\begin{aligned}&\mathcal{L}(M_1,\cdots,M_N,\mathcal{X},U_1,\cdots,U_N,Y_1,\cdots,Y_N,\mu)\\&=\sum_{n=1}^{N}(w_n\|M_n\|_*+\langle Y_n,M_n-U_n\rangle+\frac{\mu}{2}\|M_n-U_n\|_F^2)\\&\qquad+\frac{\lambda}{2}\|\mathcal{X}-U_1\circ U_2\cdots\circ U_N\|_F^2,\end{aligned}\tag{4.14}$$

where $Y_n\in\mathbb{R}^{I_n\times R},\ n=1,\cdots,N$ are the Lagrange multipliers, and $\mu>0$ is a penalty parameter.

In the following, we present an ADMM iterative scheme to minimize $\mathcal{L}$ with respect to $(M_1,\cdots M_N,\mathcal{X},U_1,\cdots U_N,Y_1,\cdots Y_N)$. More specifically, to update the variables $M_n,\ n=1,\cdots,N$ with fixing other variables, the optimization problem is formulated concretely as follows:

$$\min_{M_n}w_n\|M_n\|_*+\frac{\mu^k}{2}\|M_n-U_n^k+Y_n^k/\mu^k\|_F^2.\tag{4.15}$$

Following [3], a closed-form solution to the problem (4.15) can be obtained easily as follows:

$$M_n^{k+1}=\text{SVT}_{w_n/\mu^k}(U_n^k-Y_n^k/\mu^k),\tag{4.16}$$

where $\text{SVT}_\delta(A)=U\text{diag}(\{(\sigma-\delta)_+\})V^T$ is a singular value thresholding (SVT) operator, the SVD of A is given by $A=U\text{diag}(\{\sigma_i\}_{1\leq i\leq r})V^T$, $t_+=\max(0,t)$, and $\max(\cdot,\cdot)$ should be understood element-wise. The computation complexity of the SVT operator is $O(I_nR^2)$. Thus, our NNCP method has a significantly lower complexity than other nuclear norm minimization algorithms for both optimization problems (2.2) and (2.4) which require to perform SVDs on the unfolded matrices $X_{(n)}$ with the size of $I_n\times\prod_{i\neq n}I_i,\ n=1,\cdots,N$ at each iteration, with the time cost $O(I_n^2\times\prod_{i\neq n}I_i)$.

To update the variables $(U_1,\cdots,U_N)$, the optimization problem is given by:

$$\begin{aligned}&\min_{U_1,\cdots,U_n}\frac{\lambda}{2}|\mathcal{X}^k-U_1\circ\cdots\circ U_N\|_F^2\\&\qquad+\sum_{n=1}^{N}\frac{\mu^k}{2}\|U_n-M_n^{k+1}-Y_n^k/\mu^k\|_F^2.\end{aligned}\tag{4.17}$$

To solve $U_n,\ n=1,\cdots,N$ with fixing other variables, the problem (4.17) becomes a smooth optimization problem. Let $B_n=(U_N^k\circ\cdots U_{n+1}^k\circ U_{n-1}^{k+1}\circ\cdots U_1^{k+1})$, then the resulting subproblem with respect to U_n is formulated as follows:

$$\min_{U_n}\frac{\lambda}{2}\|U_nB_n-X_{(n)}^k\|_F^2+\frac{\mu^k}{2}\|U_n-M_n^{k+1}-Y_n^k/\mu^k\|_F^2.\tag{4.18}$$

Thus the inexact updates of U_n are given by solving the optimization problem (4.18) as follows [18]:

$$U_n^{k+1}=\frac{1}{\lambda+\mu^k}(\lambda X_{(n)}^kB_n^T+\mu^kM_n^{k+1}+Y_n^k)(\lambda B_nB_n^T+\mu^kI)^{-1}.\tag{4.19}$$

To update the variable $\mathcal{X}$, we have the following subproblem,

$$\begin{aligned}&\min_{\mathcal{X}}\|\mathcal{X}-U_1^{k+1}\circ\cdots\circ U_N^{k+1}\|_F^2,\\&\text{s.t.}, \mathcal{X}_\Omega=\mathcal{T}_\Omega.\end{aligned}\tag{4.20}$$

Furthermore, the optimal solution $\mathcal{X}$ satisfies the following equation by deriving simply the KKT conditions for (4.20):

$$\mathcal{X}_\Omega^{k+1}=\mathcal{T}_\Omega\text{ and }\mathcal{X}_{\Omega^C}^{k+1}=(U_1^{k+1}\circ\cdots\circ U_N^{k+1})_{\Omega^C},\tag{4.21}$$

where Ω^C is the complement of Ω, i.e., the set of indexes of the unobserved entries.

Based on the above analysis, we develop the following ADMM iterative scheme for the tensor completion problem (3.12), as outlined in **Algorithm 1**. Nonetheless, the proposed model (3.12) is non-convex and proving the convergence properties of ADMM in theory is still an open issue [12]. The stability and efficiency of our NNCP algorithm (i.e., **Algorithm 1**) can be validated in the experimental section.

4.2 Analysis of NNCP The proposed NNCP model (3.12) can be reformulated as follows:

$$\min_{U_n}\sum_{n=1}^{N}\frac{w_n}{\lambda}\|U_n\|_*+\frac{1}{2}\|\mathcal{W}*(\mathcal{T}-U_1\cdots\circ U_N)\|_F^2.\tag{4.22}$$

Thus, our proposed model (3.12) is also a nuclear norm regularized least squares problem. When letting $\lambda\to\infty$, the model (4.22) degenerates to the ordinary CP tensor decomposition model (2.5) for tensor completion problems.

In this sense, our NNCP algorithm is also a fast higher-order tensor decomposition method in the presence of missing data and gives a concise representation of the latent structure of the tensor. And the existing tensor completion model (2.5) based on the CP decomposition is a special case of our NNCP method.

Algorithm 1 Solving the model (3.12) via ADMM.

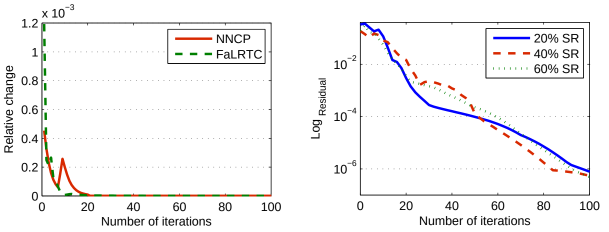
```
Input: $N$-order tensor $\mathcal{T}$, index set $\Omega$, rank $R$, and parameter $\lambda$.
Initialize: $Y_n^0 = 0$, $M_n^0 = 0$, $U_n^0 = \text{rand}(I_n, R)$, $n = 1, \cdots, N$, $\mu^0 = 10^{-6}$, $\mu_{max} = 10^{10}$, $\rho = 1.15$, and tol $= 10^{-5}$.
while not converged do
    for $n = 1 : N$ do
        Update $M_n^{k+1}$ by (4.16);
        Update $U_n^{k+1}$ by (4.19).
    end for
    Update $\mathcal{X}^{k+1}$ by (4.21).
    Update the multiplier $Y_n^{k+1}$ by
        $Y_n^{k+1} = Y_n^k + \mu^k (M_n^{k+1} - U_n^{k+1})$.
    Update the parameter $\mu^{k+1}$ by
        $\mu^{k+1} = \min(\rho\mu^k, \mu_{max})$.
    Check the convergence condition,
    $\max\{\|M_n^{k+1} - U_n^{k+1}\|_F,\ n = 1, \cdots, N\} < \text{tol}$.
end while
Output: $U_n$, $n = 1, \cdots, N$.
```

Next, we discuss the time complexity of our NNCP algorithm. For the tensor completion problem (3.12), the main running time of NNCP is consumed by performing SVD for the SVT operator and some multiplications. The time complexity of performing SVD is $O_1 = O(\sum_{n=1}^{N} I_n R^2)$. The time complexity of some multiplication operators is $O_2 = O(R \sum_{n=1}^{N} (\sum_{i=1}^{n-1} \prod_{j=n-i}^{n-1} I_j + \sum_{i=1}^{N-n-1} \prod_{j=n-i}^{N} I_j))$ and $O_3 = O(NR \prod_{n=1}^{N} I_n)$. Thus, the total time complexity of **Algorithm 1** is $O(T(O_1 + O_2 + O_3))$, where T is the number of iterations.

5 Experiments

In this section, we evaluate the effectiveness and efficiency of our NNCP algorithm for some tensor completion tasks on both synthetic data and real-world data including multi-relational prediction.

5.1 Synthetic Data In this part we generated a low-n-rank tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \cdots I_N}$ which is used as ground truth data. The tensor data follow the Tucker model, i.e., $\mathcal{T} = \mathcal{C} \times_1 U_1 \times_2 \cdots \times_N U_N$, where the core tensor $\mathcal{C} \in \mathbb{R}^{r \times r \cdots \times r}$ and $U_n \in \mathbb{R}^{I_n \times r}$ are generated with i.i.d. standard Gaussian entries. With this construction, the n-rank of $\mathcal{T}$ equals $(r, r, \cdots, r)$ almost surely. The order of the tensors varies from three to five, and r is set to 5. Furthermore, we randomly sample a few entries from $\mathcal{T}$ and recover the whole tensor with various sampling rates (SR) by our NNCP algorithm and three existing state-of-the-art algorithms including weighted Tucker (WTucker)[1] [9], weighted CP (WCP)[2] [1] and FaLRTC[3] [20].



(a) The relative change (b) The log-value of residual

Figure 2: Convergence behaviors of FaLRTC and our NNCP algorithm on the synthetic tensor data of size 80 × 80 × 80.

We set tol $= 10^{-5}$ and maxiter=1000 for all these algorithms. In the implementation of our NNCP algorithm, we set the regularization parameter $\lambda = 10$. For our NNCP algorithm and FaLRTC, the weights w_n are set to $w_n = 1/N, n = 1, \cdots, N$, and the smoothing parameter for FaLRTC is set to $\mu_n = 5w_n/I_n$. The other parameters of FaLRTC are set to their default values. For the three tensor decomposition methods including NNCP, WTucker, and WCP, the tensor rank is set to $R = 10$. The relative square error (RSE) of the recovered tensor $\mathcal{X}$ for all these algorithms is given by RSE $= \|\mathcal{X} - \mathcal{T}\|_F / \|\mathcal{T}\|_F$. The experiments are performed with Matlab 7.11 on an Intel Core 2 Duo (3.0 GHz) PC running Windows XP with 2-GB main memory.

The average results (RSE and time cost) of 10 independent runs are shown in Table 1, where the order of tensor data varies from four to five. From the results shown in Table 1, we can see that our NNCP algorithm usually yields much more accurate solutions using less time, and often outperforms the other three compared algorithms including WTucker, WCP and FaLRTC in terms of RSE and efficiency.

We also study the convergence behaviors of FaLRTC and our NNCP algorithm on the synthetic tensor data of size 80 × 80 × 80 with the given ranks of the tensor $r_1 = r_2 = r_3 = 5$, as shown in Fig. 2, where the ordinate is the log-value of the residual of $\max\{\|M_n^{k+1} - U_n^{k+1}\|_F, n = 1, \cdots, N\}$, or the relative change of $\mathcal{X}^k$, and the abscissa denotes the number of iteration. We can observe that the relative change or the residual of our NNCP algorithm drops much quickly, and converges much fast within 50 iterations.

To further evaluate the robustness of our NNCP

[1] http://www.lair.irb.hr/ikopriva/marko-filipovi.html

[2] http://www.sandia.gov/∼tgkolda/TensorToolbox/

[3] http://pages.cs.wisc.edu/∼ji-liu/

Table 1: The RSE and time cost (seconds) comparison on the synthetic data:
(a) Tensor size: $20 \times 20 \times 20 \times 20 \times 20$

	WTucker		WCP		FaLRTC		NNCP	
SR	RSE	Time	RSE	Time	RSE	Time	RSE	Time
20%	2.26e-01	4071.76	4.48e-01	2634.85	4.65e-01	1198.77	1.58e-01	209.39
50%	6.34e-02	2451.53	1.25e-01	2463.57	1.14e-01	821.46	7.86e-02	224.67
80%	3.17e-02	3034.17	7.26e-02	2396.24	5.77e-02	852.21	2.73e-02	260.25

(b) Tensor size: $30 \times 30 \times 30 \times 30$

	WTucker		WCP		FaLRTC		NNCP	
SR	RSE	Time	RSE	Time	RSE	Time	RSE	Time
20%	1.85e-01	502.93	9.72e-02	708.04	4.48e-01	359.43	9.48e-02	34.59
50%	6.59e-02	439.52	7.13e-02	645.63	1.16e-01	512.85	5.63e-02	36.35
80%	4.84e-02	490.86	5.40e-02	669.63	6.48e-02	473.20	1.85e-02	39.81

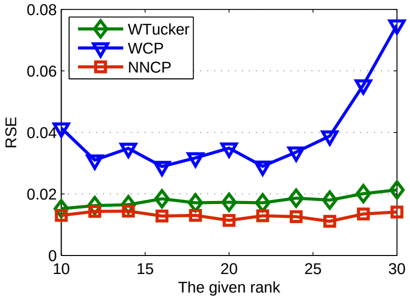


Figure 3: The RSE results of WTucker, WCP and our NNCP algorithm versus the given tensor ranks.

algorithm versus the given tensor rank, we conduct some experiments on the rank-$(5, 5, 5)$ synthetic data of size $80 \times 80 \times 80$, and illustrate the recovery results of WTucker, WCP and our NNCP algorithm with 30% sampling rate where the rank parameter R for these three algorithms is chosen from $\{10, \cdots, 30\}$. The average RSE results of 10 independent runs are shown in Fig. 3, from which we can see that our NNCP algorithm performs much more robust than both WTucker and WCP in terms of the given tensor rank. This also confirms that our NNCP model with nuclear norm regularization can provide a good estimation of a low-rank tensor.

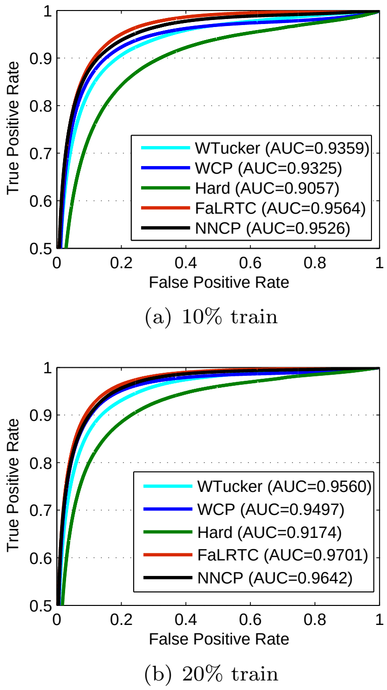
5.2 Large-Scale Link Prediction In this part, we examine our method on real-world network data sets, such as the YouTube data set[4] [30]. YouTube is currently the most popular video sharing web site, which allows users to interact with each other in various forms such as contacts, subscriptions, sharing favorite videos, etc. In total, this data set reaches 848,003 users, with 15,088 users sharing all of the information types, and includes 5-dimensions of interactions: contact network, co-contact network, co-subscription network, co-subscribed network, and favorite network. Additional information about the data can be found in [30]. We ran these experiments on a machine with 6-core Intel Xeon 2.4GHz CPU and 64GB memory.

We address the link prediction problem as a tensor completion problem. For our NNCP algorithm, we set the regularization parameter $\lambda = 10$. The tolerance value of our NNCP method, WTucker, WCP, the hard completion (Hard) method[5] [27] and FaLRTC is fixed at tol$= 10^{-5}$. For FaLRTC, the weights w_n are set to $w_n = 1/3, n = 1, 2, 3$, and the smoothing parameters are set to $\mu_n = 5w_n/I_n$, $n = 1, 2, 3$. Besides, for the hard completion method we let $\tau = 10^4$ and $\lambda_1 = \lambda_2 = \lambda_3 = 1$. For the three tensor decomposition methods including NNCP, WTucker, and WCP, the tensor ranks are set to $R_1 = R_2 = 40$ and $R_3 = 5$.

We compare the link prediction results of NNCP, WTucker, WCP, Hard and FaLRTC on the YouTube data set. Note that we chose the users who have more than 10 interactions as a subset, which consists of 4,117 users and five types of interactions, i.e., $4,117 \times 4,117 \times 5$. We randomly select 10% or 20% entries as the training set, and the remainder as the testing data. We present the average prediction accuracy (the

[4] http://leitang.net/heterogeneous_network.html

[5] https://sites.google.com/site/marcosignoretto/codes



(a) 10% train

(b) 20% train

Figure 4: Average ROC curves showing the performance of link prediction methods with 10% and 20% training data, respectively.

score Area Under the receiver operating characteristic Curve, AUC) over 10 independent runs in Fig. 4, and the average running time (in seconds) of all these five algorithms over 10 independent runs in Fig. 5, where the number of users is gradually increased. From the results shown in Figs. 4 and 5, we can see that two trace norm regularized tensor completion algorithms, NNCP and FaLRTC outperform WTucker, WCP and Hard in terms of prediction accuracy. Our NNCP algorithm can achieve very similar performance with FaLRTC both in the Receiver Operating Characteristic (ROC) curve and in terms of AUC. Moreover, our NNCP algorithm runs significantly faster than FaLRTC, WTucker, WCP and Hard. The runtime of NNCP increases slightly when the number of users increases. This shows that our NNCP method has very good scalability and can address large-scale problems. In contrast, the runtime of WTucker, WCP, FaLRTC and Hard increases dramatically. They could not complete within 48 hours on the two largest data sets with 8,000 and 15,088 users.

6 Conclusions and Future Work

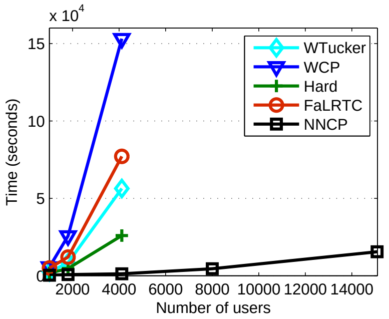


Figure 5: Time cost (seconds) comparison on the YouTube data set. For each dataset, we use 20% for training. Note that the other four methods including WTucker, WCP, Hard and FaLRTC could not run sizes $\{8,000, 15,088\}$ due to runtime exceptions.

We proposed a novel factor matrix n-rank minimization method based on the CP decomposition for low-rank tensor completion and decomposition tasks. We first used a factor matrix rank minimization instead of the low-n-rank minimization of involved tensors. Then, we relaxed the matrix rank into a tractable convex surrogate, and obtained a much smaller scale factor matrix nuclear norm minimization problem. Finally, we developed an efficient ADMM solving scheme to solve the proposed problem. Our convincing experimental results on both synthetic and real-world data verified both the efficiency and effectiveness of our NNCP algorithm.

Moreover, our NNCP algorithm can handle large-scale tensor completion and decomposition problems, and its performance is very robust to the given tensor rank. For future work, we will theoretically analyze the relationship of the optimal solutions to our NNCP model (3.12) and the traditional tensor nuclear norm minimization model (2.4). We are also interested in exploring ways to regularize our proposed model with auxiliary information as in [23], [25].

Acknowledgments

We would like to thank Dr. Ji Liu of University of Wisconsin-Madison, Dr. Marco Signoretto of Katholieke Universiteit Leuven and Dr. Marko Filipovic for sharing their codes. Software based on the Matlab Tensor Toolbox [16] will be made available from http://www.sandia.gov/∼tgkolda/TensorToolbox/index-2.5.html. This work is supported by the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) Project Nos. CUHK 411211 and 411310, and the Chinese University of Hong Kong Direct Grant Nos. 4055015 and 4055017. This last author is partially supported by the National Science Foundation under Grant No. IIS1017415 and by the Army Research Laboratory under Cooperative Agreement

No. W911NF-09-2-0053.

References

[1] E. Acar, D.M. Dunlavy, T.G. Kolda, and M. Morup. Scalable tensor factorizations with missing data. In *SDM*, pp. 701–711, 2010.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, 2011.

[3] J. Cai, E. J. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.*, 20(4):1956–1982, 2010.

[4] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9(6):717–772, 2009.

[5] E. J. Candes, and T. Tao. Decoding by linear programming. *IEEE Trans. Inform. Theory*, 51(2):4203–4215, 2009.

[6] L. De Lathauwer. A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization. *SIAM J. Matrix Anal. Appl.*, 28:642–666, 2006.

[7] L. De Lathauwer and J. Vandewalle. Dimensionality reduction in higher-order signal processing and rank-(r1, r2, . . ., rn) reduction in multilinear algebra. *Lin. Algebra. Appl.*, 391:31–55, 2004.

[8] M. Fazel. Matrix rank minimization with applications. PhD Thesis, Stanford University, California, 2002.

[9] M. Filipovic and A. Jukic. Tucker factorization with missing data with application to low-n-rank tensor completion. *Multidim. Syst. Sign. Process.*, 2014.

[10] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problem*, 27(2), 025010, 2011.

[11] R.A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA working papers in phonetics*, 16:1–84, 1970.

[12] B. He, M. Tao, and X. Yuan. Alternating direction method with Gaussian back substitution for separable convex programming. *SIAM J. Optim.*, 22(2):313–340, 2012.

[13] C. J. Hillar, L. H. Lim. Most tensor problems are NP hard. *J. ACM*, 45(6), 2013.

[14] R. Bro, H. A.L. Kiers. A new efficient method for determining the number of components in PARAFAC models. *J. Chemometrics*, 17:274–286, 2003.

[15] T.G. Kolda and B.W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[16] T.G. Kolda and J. Sun. Scalable tensor decompositions for multi-aspect data mining. in *ICDM*, pp. 363–372, 2008.

[17] L. Lathauwer, B Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

[18] Z. Lin, M. Chen, and L. Wu. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical Report UILU-ENG-09-2215, Univ. Illinois, Urbana-Champaign, Oct. 2009.

[19] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. In *ICCV*, pp. 2114–2121, 2009.

[20] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. *IEEE TPAMI*, 35(1):208–220, 2013.

[21] Y. Liu, L. C. Jiao, and F. Shang. A fast tri-factorization method for low-rank matrix recovery and completion. *pattern recogn.*, 46(1):163–173, 2013.

[22] M. Morup, L.K. Hansen, C.S. Herrmann, J. Parnas, and S.M. Arnfred. Parallel factor analysis as an exploratory tool for wavelet transformed event-related EEG. *NeuroImage*, 29(3):938–947, 2006.

[23] A. Narita, K. Hayashi, R. Tomioka, and H. Kashima. Tensor facotiration using auxiliary information. *DMKD*, 25(2):298–324, 2012.

[24] B. Recht, M. Fazel, and P. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[25] F. Shang, L.C. Jiao, and F. Wang. Graph dual regularization non-negative matrix factorization for co-clustering. *pattern recogn.*, 45(6):2237–2250, 2012.

[26] Y. Shen, Z. Wen, and Y. Zhang. Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optim. Method Softw.*, 29(2):239–263, 2014.

[27] M. Signoretto, Q. T. Dinh, L.D. Lathauwer, and J. Suykens. Learning with tensors: a framework based on covex optimization and spectral regulzation. *Mach. Learn.*, in press, 2014.

[28] M. Signoretto, L.D. Lathauwer, and J. Suykens. Nuclear norms for tensors and their use for convex multilinear estimation. Technical Report 10-186, ESAT-SISTA, K. U. Leuven, 2010.

[29] J. Sun, S. Papadimitriou, C.-Y. Lin, N. Cao, S. Liu, and W. Qian. MultiVis: content-based social network exploration through multi-way visual analysis. In *SDM*, pp. 1063–1074, 2009.

[30] L. Tang, X. Wang and H. Liu. Uncovering groups via heterogeneous interaction analysis. In *ICDM*, pp. 503–512, 2009.

[31] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[32] 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(4):333–361, 2012.

[33] Y. K. Yilmaz, A. T. Cemgil, and U. Simsekli. Generalized coupled tensor factorization. In *NIPS*, pp. 2151–2159, 2011.