This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2017.2760287, IEEE Transactions on Fuzzy Systems

Fuzzy Double Trace Norm Minimization for Recommendation Systems

Fanhua Shang, *Member, IEEE,* Yuanyuan Liu, James Cheng, and Da Yan

***Abstract*—Recovering low-rank matrices from incomplete observations is a fundamental problem with many applications, especially in recommender systems. In theory, under certain conditions, this problem can be solved by convex or non-convex relaxation. However, most existing provable algorithms suffer from superlinear per-iteration cost, which severely limits their applicability to large-scale problems. In this paper, we propose a novel fuzzy double trace norm minimization (DTNM) method for recommender systems. We first present a tractable DTNM model, in which we can integrate both the user social relationship and the user reputation information using a fuzzy weighting way and coupling fuzzy matrix factorization. In essence, our model is a Schatten-$1/2$ quasi-norm minimization problem. Moreover, we develop two efficient augmented Lagrangian algorithms to solve the proposed problems, and prove the convergence of our algorithms. Finally, we investigate the empirical recoverability properties of our model and its advantage over classical trace norm. Extensive experimental results on both synthetic and real-world data sets verified both the efficiency and effectiveness of our method compared with the state-of-the-art algorithms.**

***Index Terms*—Collaborative filtering; double trace norm; fuzzy weighting; contextual information; matrix completion**

I. Introduction

In recent years, many trace norm (also known as the nuclear norm) or Schatten-p $(0 < p < 1)$ quasi-norm minimization methods have been employed in many machine learning and data mining applications, such as low-rank matrix completion (LRMC), matrix classification, multi-task learning and dimensionality reduction [1]. For solving such convex or non-convex optimization problems, those algorithms do not require the rank to be specified and have to be solved iteratively. Naturally, the singular value decomposition (SVD) tends to play a critical computational role in the design of various solvers, e.g., APG [2] and IRucLq [3]. All those algorithms involve SVD and apply a soft-thresholding operator on singular values in each iteration, and thus they suffer from high computational cost $O(mn^2)$ of SVDs [4]–[6]. In particular, when their iterations need to pass through a region where the spectrum is dense, they can become prohibitively expensive to run when the size of the associated matrices grows beyond a few thousand [7], e.g., the MovieLens data sets. Only those singular values that exceed a threshold, and associated singular vectors, contribute to the soft-thresholding operator. Thus, a commonly used strategy is to compute only partial SVD instead of the full one, for example, APG [2] uses the PROPACK package [8]. However, it can compute only a given number of largest singular values, and the soft-thresholding operator requires the principal singular values to be greater than a given threshold [7], [9].

Matrix factorization is arguably the most widely used method for real-world problems including the Netflix Prize, due to its high accuracy, scalability and flexibility to incorporate side-information [10], [11]. If the rank of given matrices is known, a class of matrix factorization algorithms [5], [12]–[14] cast the LRMC problem as a fixed-rank matrix factorization problem. Wen *et al.* [14] proposed a successive over-relaxation algorithm to solve such a problem. In [5] and [13], two improved algorithms were proposed to optimize such problems on Grassmannian manifolds, and improve its convergence using conjugate gradients rather than standard gradient descent. Moreover, Keshavan *et al.* [12] proved that exact recovery can be obtained with high probability by solving a non-convex optimization problem. In all of the models mentioned above, the correct rank needs to be known a priori. Unfortunately, the determination of the reduced rank is also an open problem, especially for the noisy matrix estimation.

Recommender systems, which attempt to tackle the information overload problem by suggesting to users the information that is potentially of interests, attract more and more attention in recent years [15]–[17]. However, traditional recommender systems assume that users are independent and identically distributed (i.i.d.), and purely mine the user-item rating matrix for recommendations. Therefore, they give somewhat unrealistic output. But in real life, for example, when we ask our friends for recommendations of smart phones or movies, we are actually utilizing the social contextual information for recommendations. Social relations provide an independent source of information about users beyond rating information, and can potentially be used to improve the performance of traditional recommender systems [11], [16], [18]–[20].

In this paper, we focus on three major challenges faced by existing recommender systems, namely the computational efficiency, the robustness of the rank parameter, and the incorporation of the social contextual information. In order to address such challenges, we propose a novel fuzzy double trace norm minimization (DTNM) method. We summarize the main contributions of this work as follows. 1) We first formulate the LRMC problem as a tractable DTNM problem, which significantly reduces the computational cost per-iteration from $O(mn^2)$ to $O(mnd)$ ($d \ll m, n$ in general). 2) In fact, our model is a Schatten-$1/2$ quasi-norm minimization problem, and can be extended to incorporate the social con-

F. Shang (Corresponding author), Y. Liu and J. Cheng are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. E-mail: {fhshang, yyliu, jcheng}@cse.cuhk.edu.hk.

D. Yan is with the Department of Computer and Information Sciences, University of Alabama, Birmingham, USA. E-mail: yanda@uab.edu.

Manuscript received February 26, 2016.

textual information such as the user social relationship and user reputation information by using a fuzzy weighting way and coupling fuzzy matrix factorization. 3) We develop two efficient algorithms based on the alternating direction methods of multipliers (ADMM) to solve our challenging non-convex non-smooth problems. 4) Finally, we provide convergence guarantees for our algorithms, and analyze the superiority of our double trace norm over the popular trace norm.

The rest of the paper is organized as follows. We review some related work in Section II. In Section III, we propose a DTNM model and then incorporate both the user social relationship and user reputation information into it. We develop two efficient ADMM algorithms in Section IV. We give theoretical analysis in Section V. We report empirical results in Section VI, and conclude this paper in Section VII.

II. Background and Related Work

The Schatten p-norm ($0<p<\infty$) of a matrix $X\in\mathbb{R}^{m\times n}$ is defined as

$$\|X\|_{S_p}\triangleq[\sum_{i=1}^{\min(m,n)}\sigma_i^p(X)]^{1/p}$$

where $\sigma_i(X)$ denotes the i-th singular value of X. When $p=1$, the Schatten 1-norm is the well-known trace norm, $\|X\|_*$. As non-convex surrogate functions of the matrix rank [3], [21], the Schatten-p quasi-norm with $0<p<1$ is a better approximation than the trace norm, and so is the ℓ_p-quasi-norm a better approximation than the ℓ_1-norm [22], [23].

A. Schatten Norm Minimization

Given a matrix, some of its entries may not be observed due to problems in the acquisition process, e.g., loss of information or high cost of experiments to obtain complete data [24]. To recover a low-rank matrix from a small number of entries, we solve the general Schatten p-norm minimization problem

$$\min_{X\in\mathbb{R}^{m\times n}}\|X\|_{S_p}^p,\quad \text{s.t.}, \mathcal{P}_\Omega(X)=\mathcal{P}_\Omega(D) \tag{1}$$

or the Schatten p-norm regularized least squares problem

$$\min_{X\in\mathbb{R}^{m\times n}}\|X\|_{S_p}^p+\frac{\lambda}{2}\|\mathcal{P}_\Omega(X)-\mathcal{P}_\Omega(D)\|_F^2 \tag{2}$$

where in general $p\in[0,1]$, $\lambda>0$ is a regularization parameter, and $\mathcal{P}_\Omega(D)$ is defined as the projection of D onto the observed entries $\Omega:=\{(i,j)|D_{ij}\text{ is observed}\}$: $\mathcal{P}_\Omega(D)_{ij}=D_{ij}$ if $(i,j)\in\Omega$ and $\mathcal{P}_\Omega(D)_{ij}=0$ otherwise.

Many convex trace norm minimization (TNM) algorithms (e.g., APG [2]), reweighted trace norm (e.g., [25]) or Schatten quasi-norm minimization (SNM) algorithms (e.g., IRucLq [3], [21], [26]) have been proposed to solve the problems (1) and (2) or their special cases, e.g., $p=1$. However, they have to be solved iteratively and involve SVD of very large matrices in each iteration. Therefore, they suffer from high computational cost and are even not applicable for large-scale problems.

B. Matrix Factorization Formulations

Alternatively, fixed-rank matrix factorization for LRMC has received a significant amount of attention [5], [12]–[14], [27]. For example, Wen *et al.* [14] proposed a successive over-relaxation iteration scheme to alternatively solve the following least-squares problem,

$$\min_{U\in\mathbb{R}^{m\times d},V\in\mathbb{R}^{n\times d},X}\|UV^T-X\|_F^2,\ \text{s.t.}, \mathcal{P}_\Omega(X)=\mathcal{P}_\Omega(D). \tag{3}$$

The matrix factorization problem can also be converted into some Riemannian manifold optimization problems, such as [5], [12], [13]. However, in all the algorithms we need to know the exact rank, which is usually difficult to obtain, especially for noisy matrices. Thus, for noisy matrix completion, the importance of regularization in such problems is well known to practitioners. For instance, one important component of many algorithms competing for the Netflix challenge involves minimizing the following objective function [11],

$$\min_{U,V}\frac{\lambda}{2}\|\mathcal{P}_\Omega(UV^T)-\mathcal{P}_\Omega(D)\|_F^2+\frac{1}{2}(\|U\|_F^2+\|V\|_F^2). \tag{4}$$

Some similar bilinear spectral regularized matrix factorization (SRMF) formulations are used in [5], [12], [13]. Srebro *et al.* [28] and Mazumder *et al.* [4] pointed out the equivalence relation between the bilinear spectral regularization and the trace norm as follows.

Lemma 1. *Given a matrix $X\in\mathbb{R}^{m\times n}$ with rank$(X)\le d$, the following equalities hold:*

$$\|X\|_*=\min_{X=U_{m\times d}V_{n\times d}^T}\frac{1}{2}(\|U\|_F^2+\|V\|_F^2)=\min_{X=UV^T}\|U\|_F\|V\|_F.$$

III. Double Trace Norm Minimization

In this section, we propose a novel double trace norm minimization model for LRMC problems, which is a tractable Schatten-$1/2$ quasi-norm minimization problem. Moreover, we present a generic coupled fuzzy matrix factorization model for fuzzy recommender systems, in which we incorporate both the user social relationship and user reputation information.

A. Double Trace Norm

Let d be an upper bound on the rank of a low-rank matrix X, i.e., $d\ge r=\text{rank}(X)$, then X can be decomposed into a product of two much smaller factor matrices, $U\in\mathbb{R}^{m\times d}$ and $V\in\mathbb{R}^{n\times d}$, such that $X=UV^T$. In particular, Keshavan *et al.* [12] and Wen *et al.* [14] presented several matrix rank estimation strategies to compute a good value r' for the rank of the involved matrix. Thus, we only set a relatively large integer d such that $d\ge r'$, e.g., $d=r'+1$. Inspired by the equivalence relation between the trace norm and the bilinear spectral regularization shown in Lemma 1, our double trace norm is naturally defined as follows [29].

Definition 1. *For any $X\in\mathbb{R}^{m\times n}$ with rank$(X)\le d$, it can be factorized into $U\in\mathbb{R}^{m\times d}$ and $V\in\mathbb{R}^{n\times d}$ such that $X=UV^T$. Then the double trace norm of X is defined as*

$$\|X\|_{**}:=\min_{X=UV^T}\left(\frac{\|U\|_*+\|V\|_*}{2}\right)^2.$$

In fact, the double trace norm is not a real norm, because it is non-convex and does not satisfy the triangle inequality of a norm. Similar to the definitions in [6], [30], which cannot be directly used in real-world applications, the double trace norm is the Schatten-1/2 quasi-norm, as stated in the following theorem [29] (whose proof is also different from that in [6]).

Theorem 1. *The double trace norm* $\|\cdot\|_{**}$ *is the Schatten-*$1/2$ *quasi-norm.*

It is easy to verify that the double trace norm possesses the following properties [30].

Property 1. *The double trace norm satisfies the following properties:*

1) $\|X\|_{**} \geq 0$, *with equality iff* $X = \mathbf{0}$*;*
2) $\|X\|_{**}$ *is unitarily invariant, i.e.,* $\|X\|_{**} = \|PXQ^T\|_{**}$*, where both* $P \in \mathbb{R}^{m\times m}$ *and* $Q \in \mathbb{R}^{n\times n}$ *are orthonormal matrices.*

Property 2. *For any matrix* $X \in \mathbb{R}^{m\times n}$ *with rank*$(X) \leq d$*, the following equalities hold:*

$$\|X\|_{**} = \min_{X=U_{m\times d}V^T_{n\times d}} \frac{1}{4}(\|U\|_* + \|V\|_*)^2 = \min_{X=UV^T} \|U\|_*\|V\|_*.$$

The higher efficiency of the $\ell_{1/2}$-quasi-norm than the ℓ_1-norm for sparse vector recovery was demonstrated in signal and image recovery applications [31], [32]. By realizing the intimate relationship between the ℓ_p-quasi-norm and the Schatten-p quasi-norm, the Schatten-$1/2$ quasi-norm generally obtains better empirical performance than the trace norm [25]. Some work [3], [26], [31] has shown that $p = 1/2$ is the best choice. We apply the double trace norm as a surrogate of the rank function, and propose the following formulation,

$$\min_X \|X\|_{**}^{1/2}, \quad \text{s.t., } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(D) \tag{5}$$

and its Lagrangian version

$$\min_{X:\,\text{rank}(X)\leq d} \|X\|_{**}^{1/2} + \frac{\lambda}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(D)\|_F^2. \tag{6}$$

We call (6) as a double trace norm minimization (DTNM) model for social recommendation. To solve (6), we only need to perform SVDs on two much smaller factor matrices $U \in \mathbb{R}^{m\times d}$ and $V \in \mathbb{R}^{n\times d}$ in each iteration. Hence, our DTNM method is very efficient. DTNM is also robust to the given rank d, which will be verified by experiments in Section VI.

B. Fuzzy Weighted Model

In fact, the ratio of observed elements in real-world recommendation system data is very small, e.g., 1.31% in MovieLens10M. Thus, there has been an upsurge of interest in utilizing some side-information about users/items to compensate for the insufficiency of rating information. For example, user reputation plays an important role in recommendation and has attracted various real-world applications [15], [16]. Seno and Lukas [33] found that suggestions from people with high reputations positively affect a consumer's adoption of a brand. Massa [34] found that ratings from users with high reputations are more likely to be trustworthy.

There are many algorithms to calculate the reputations of nodes in social networks according to their connections, and we adopt one of the most popular algorithms, PageRank [35], to compute the user reputation scores in this network. We first perform PageRank to rank users by exploiting the perspective of social relations similar to [17]. We assume that $r_i \in [1, m]$ is the reputation ranking of the i-th user u_i, where $r_i = 1$ denotes that u_i has the highest reputation in the whole social network. Then we define the fuzzy user reputation score w_i as a function h of user reputation ranking r_i as follows:

$$w_i = h(r_i) = \frac{1}{1 + \log(r_i)/\log^2(m)} \tag{7}$$

where the fuzzy reputation score $w_i \in [0, 1]$ and the function h is a decreasing function of r_i, i.e., top-ranked users have high reputation scores similar to leverage scores in [17], [36].

By incorporating the user reputation into (6), we obtain the following fuzzy weighted DTNM problem

$$\min_{X:\,\text{rank}(X)\leq d} \|X\|_{**}^{1/2} + \frac{\lambda}{2}\|T_1 \odot (X - D)\|_F^2 \tag{8}$$

where $\odot$ denotes the Hadamard product, i.e., $[A \odot B]_{ij} = A_{ij}B_{ij}$, and $T_1 \in \mathbb{R}^{m\times n}$ is formally defined as

$$(T_1)_{ij} = \begin{cases} \sqrt{w_i}, & \text{if } (i,j) \in \Omega, \\ 0, & \text{otherwise.} \end{cases}$$

It is clear that (6) can be viewed as a special case of (8), where $w_i = 1$ for all $i = 1, \ldots, m$. That is, a large value of the reputation score w_i, indicating high reputation of the user, will force X_{ij} to tightly fit the rating D_{ij}, while D_{ij} will be loosely approximated by X_{ij} when w_i is small.

C. Coupled Fuzzy Matrix Factorization

Recent research on analyzing social networks has demonstrated that relational patterns of homophily can be exploited to improve predictive models of both link structure and behavior [37], [38]. That is, users with strong ties are more likely to share similar tastes than those with weak ties, and treating all social relations equally is likely to lead to degradation in recommendation performance. Thus, those observations suggest that we should consider heterogeneous strengths when exploiting user social relationship for recommendation.

Let $C = \{c_{ik}\}$ be the social network matrix. For a pair of users, u_i and u_k, the weight $c_{ik} \in (0, 1]$ is the social relation strength associated with a social link from u_i to u_k, and $c_{ik} = 0$, otherwise. In the physical world, c_{ik} can be interpreted as how much the i-th user u_i trusts the k-th user u_k in a social network. Note that $C \in \mathbb{R}^{m\times m}$ is an asymmetric fuzzy matrix, since in a trust-based social network, the user u_i trusting the user u_k does not necessarily indicate that u_k trusts u_i.

From social correlation theories, the user preferences of two socially connected users are correlated. Thus, we propose the following formulation to capture the social relationships:

$$\min_{Z:\,\text{rank}(Z)\leq d} \|Z\|_{**}^{1/2} + \frac{\gamma}{2}\|T_2 \odot (Z - C)\|_F^2 \tag{9}$$

Algorithm 1 Solving DTNM problem (6) via ADMM

Input: $\mathcal{P}_\Omega(D)$, d, λ and ε.

Initialize: $U_0 = U_0' = \mathbf{0}$, $V_0 = V_0' = \mathbf{0}$, $Y_0^i = \mathbf{0}$, and $\mu_0 = 10^{-4}$.

while not converged **do**

Update U_{k+1} and V_{k+1} by (14) and (15).

Update U'_{k+1} and V'_{k+1} by (17) and (18).

Update X_{k+1} by (20).

Update the multipliers by
$Y_{k+1}^1 = Y_k^1 + \mu_k(U_{k+1} - U'_{k+1})$, $Y_{k+1}^2 = Y_k^2 + \mu_k(V_{k+1} - V'_{k+1})$,
$Y_{k+1}^3 = Y_k^3 + \mu_k(X_{k+1} - U_{k+1}V_{k+1}^T)$.

Update μ_{k+1} by $\mu_{k+1} = \rho\mu_k$.

Check the convergence condition,
$\max\{\|U_{k+1} - U'_{k+1}\|_F, \|V_{k+1} - V'_{k+1}\|_F, \|X_{k+1} - U_{k+1}V_{k+1}^T\|_F\} < \varepsilon$.

end while

Output: X_{k+1}, U_{k+1} and V_{k+1}.

where $\gamma > 0$ is a regularization parameter, and $T_2 \in \mathbb{R}^{m\times m}$ is an indicator matrix whose (i,k)-th entry is equal to 1 if user u_i trusts user u_k and is equal to 0 otherwise.

In the above, we introduce our solutions to capture the user reputation and the observed social network relationships mathematically. With these solutions, we propose a generic coupled fuzzy matrix factorization model as follows:

$$\min_{X,Z:\, \substack{\text{rank}(X)\leq d \\ \text{rank}(Z)\leq d}} \|X\|_{**}^{1/2} + \|Z\|_{**}^{1/2} + \frac{\lambda}{2}\|T_1 \odot (X - D)\|_F^2 + \frac{\gamma}{2}\|T_2 \odot (Z - C)\|_F^2. \tag{10}$$

Our model (10) is called a double trace norm minimization (DTNMC) method with coupled fuzzy matrix factorization. In the following, we will propose two efficient augmented Lagrangian algorithms to solve (6), (8) and (10).

IV. Optimization Algorithms

In this section, we develop two efficient algorithms based on the alternating direction method of multipliers (ADMM) to solve (6), (8) and (10). The ADMM was introduced for optimization in the 1970's, and its origins can be traced back to techniques for solving partial differential equations in the 1950's. It has received renewed interests due to the fact that the ADMM is efficient to tackle large scale problems and solve the problems with multiple non-smooth terms in the objective function [39], [40]. Recently, it has been shown in the literature that the ADMM is very efficient for convex or non-convex optimization problems from many real-world applications.

A. DTNM Algorithm

According to the definition of our double trace norm and the constraint in (6), we can write X as the product of two considerably smaller matrices U and V, i.e., $X = UV^T$. Due to the interdependent matrix trace norm terms, our model (6) is difficult to solve. The key motivation of simplifying this original problem is how to split both interdependent terms such that they can be solved independently. Thus, we introduce two additional matrices U' and V' as auxiliary variables, and obtain the following equivalent formulation,

$$\min_{U,V,U',V',X} \frac{1}{2}(\|U'\|_* + \|V'\|_*) + \frac{\lambda}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(D)\|_F^2, \quad \text{s.t., } X = UV^T,\ U = U',\ V = V'. \tag{11}$$

The augmented Lagrangian function of (11) is

$$\begin{aligned}\mathcal{L}_\mu = {} & \frac{1}{2}(\|U'\|_* + \|V'\|_*) + \frac{\lambda}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(D)\|_F^2 \\ & + \langle Y^1, U - U'\rangle + \langle Y^2, V - V'\rangle + \langle Y^3, X - UV^T\rangle \\ & + \frac{\mu}{2}(\|U - U'\|_F^2 + \|V - V'\|_F^2 + \|X - UV^T\|_F^2)\end{aligned}$$

where Y^i $(i = 1,2,3)$ are the matrices of Lagrange multipliers, and $\mu > 0$ is a penalty parameter.

1) Update of U_{k+1} *and* V_{k+1}*:* By fixing the other variables at their latest values, removing and adding some proper terms that do not depend on U and V, the optimization problems with respect to U and V are formulated as follows:

$$\|U - U'_k + Y_k^1/\mu_k\|_F^2 + \|X_k - UV_k^T + Y_k^3/\mu_k\|_F^2, \tag{12}$$

$$\|V - V'_k + Y_k^2/\mu_k\|_F^2 + \|X_k - U_{k+1}V^T + Y_k^3/\mu_k\|_F^2. \tag{13}$$

Since both (12) and (13) are smooth convex optimization problems, their closed-form solutions are given by

$$U_{k+1} = \left[(X_k + Y_k^3/\mu_k)V_k + U'_k - Y_k^1/\mu_k\right](V_k^T V_k + I)^{-1}, \tag{14}$$

$$V_{k+1} = \left[(X_k + Y_k^3/\mu_k)^T U_{k+1} + V'_k - Y_k^2/\mu_k\right](U_{k+1}^T U_{k+1} + I)^{-1}. \tag{15}$$

2) Update of U'_{k+1} *and* V'_{k+1}*:* By keeping all other variables fixed, U'_{k+1} is updated by solving the following problem:

$$\|U'\|_* + \mu_k\|U_{k+1} - U' + Y_k^1/\mu_k\|_F^2. \tag{16}$$

To solve (16), the spectral soft-thresholding operator [1], [9] is considered as a shrinkage operation on the singular values, and is defined as follows:

$$U'_{k+1} = \mathcal{S}_{1/(2\mu_k)}(Q_1) := \widehat{U}\text{diag}(\max\{\widehat{\sigma} - 0.5\mu_k^{-1}, 0\})\widehat{V}^T, \tag{17}$$

where $Q_1 = U_{k+1} + Y_k^1/\mu_k$, $\max\{\cdot,\cdot\}$ should be understood element-wise, $\widehat{U}$, $\widehat{V}$, and $\widehat{\sigma} = (\widehat{\sigma}_1, \widehat{\sigma}_2, \ldots, \widehat{\sigma}_d)^T$ are obtained by SVD of Q_1, i.e., $Q_1 = \widehat{U}\text{diag}(\widehat{\sigma})\widehat{V}^T$. Similarly, the update rule of V'_{k+1} is given by

$$V'_{k+1} = \mathcal{S}_{1/(2\mu_k)}\left(V_{k+1} + Y_k^2/\mu_k\right). \tag{18}$$

3) Update of X_{k+1}*:* By fixing all other variables, the optimal X_{k+1} is the solution to the following problem:

$$\frac{\lambda}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(D)\|_F^2 + \frac{\mu_k}{2}\|X - U_{k+1}V_{k+1}^T + \frac{Y_k^3}{\mu_k}\|_F^2. \tag{19}$$

Since (19) is a smooth convex optimization problem, it is easy to show that the optimal solution to (19) is

$$\begin{aligned}X_{k+1} = {} & \mathcal{P}_\Omega[(\lambda D + \mu_k U_{k+1}V_{k+1}^T - Y_k^3)/(\lambda + \mu_k)] \\ & + \mathcal{P}_\Omega^\perp\left(U_{k+1}V_{k+1}^T - Y_k^3/\mu_k\right)\end{aligned} \tag{20}$$

where $\mathcal{P}_\Omega^\perp$ is the complementary operator of $\mathcal{P}_\Omega$, i.e., $\mathcal{P}_\Omega^\perp(D)_{ij} = 0$ if $(i,j) \in \Omega$, and $\mathcal{P}_\Omega^\perp(D)_{ij} = D_{ij}$ otherwise.

Based on the description above, we develop an efficient ADMM algorithm for solving the double trace norm minimization (DTNM) problem (6), as outlined in **Algorithm 1**. A fixed

μ is commonly used. But there are some schemes of varying the penalty parameter to achieve better convergence [39], [41]. This algorithm can also be accelerated by adaptively changing μ: $\mu_{k+1} = \rho\mu_k$, where $\rho \in (1.0, 1.1]$ in general and μ_0 is a small constant. Moreover, our algorithm can be easily applied to solve the fuzzy weighted DTNM problem (8).

B. DTNMC Algorithm

To efficiently solve our unified model (10), we also introduce three auxiliary variables U', V' and W', and obtain the following equivalent formulation:

$$\begin{aligned} \min \ & \frac{2\|U'\|_* + \|V'\|_* + \|W'\|_*}{2} + \frac{\lambda}{2}\|T_1 \odot (X - D)\|_F^2 \\ & + \frac{\gamma}{2}\|T_2 \odot (Z - C)\|_F^2, \\ \text{s.t.}, \ & X = UV^T, Z = UW^T, U = U', V = V', W = W'. \end{aligned} \tag{21}$$

1) Updating U_{k+1}, V_{k+1} and W_{k+1}: The optimization problems w.r.t. U, V and W are formulated as follows:

$$\begin{aligned} &\|U - U'_k + Y_k^1/\mu_k\|_F^2 + \|X_k - UV_k^T \\ &+ Y_k^4/\mu_k\|_F^2 + \|Z_k - UW_k^T + Y_k^5/\mu_k\|_F^2, \end{aligned} \tag{22}$$

$$\|V - V'_k + Y_k^2/\mu_k\|_F^2 + \|X_k - U_{k+1}V^T + Y_k^4/\mu_k\|_F^2, \tag{23}$$

$$\|W - W'_k + Y_k^3/\mu_k\|_F^2 + \|Z_k - U_{k+1}W^T + Y_k^5/\mu_k\|_F^2. \tag{24}$$

Since (22), (23) and (24) are all smooth convex problems, their closed-form solutions are given by

$$\begin{aligned} U_{k+1} = &[(X_k + Y_k^4/\mu_k)V_k + (Z_k + Y_k^5/\mu_k)W_k \\ &+ U'_k - Y_k^1/\mu_k](V_k^T V_k + W_k^T W_k + I)^{-1}, \end{aligned} \tag{25}$$

$$\begin{aligned} V_{k+1} = &[V'_k - Y_k^2/\mu_k + (X_k + Y_k^4/\mu_k)^T U_{k+1}] \\ &(U_{k+1}^T U_{k+1} + I)^{-1}, \end{aligned} \tag{26}$$

$$\begin{aligned} W_{k+1} = &[W'_k - Y_k^3/\mu_k + (Z_k + Y_k^5/\mu_k)^T U_{k+1}] \\ &(U_{k+1}^T U_{k+1} + I)^{-1}. \end{aligned} \tag{27}$$

2) Updating U'_{k+1}, V'_{k+1} and W'_{k+1}: U'_{k+1} is updated by solving the following problem,

$$\|U'\|_* + \frac{\mu_k}{2}\|U_{k+1} - U' + Y_k^1/\mu_k\|_F^2. \tag{28}$$

Similar to (16), the optimal solution to (28) is given by

$$U'_{k+1} = \mathcal{S}_{1/\mu_k}\left(U_{k+1} + Y_k^1/\mu_k\right). \tag{29}$$

Similarly, V'_{k+1} and W'_{k+1} are updated by

$$\begin{aligned} V'_{k+1} &= \mathcal{S}_{1/(2\mu_k)}\left(V_{k+1} + Y_k^2/\mu_k\right), \\ W'_{k+1} &= \mathcal{S}_{1/(2\mu_k)}\left(W_{k+1} + Y_k^3/\mu_k\right). \end{aligned} \tag{30}$$

Algorithm 2 Solving DTNMC problem (10) via ADMM

Input: $\mathcal{P}_\Omega(D)$, d, λ, γ and ε.

Initialize: $U_0 = U'_0 = Y_0^1 = \mathbf{0}$, $V_0 = V'_0 = Y_0^2 = \mathbf{0}$, $W_0 = W'_0 = Y_0^3 = \mathbf{0}$, and $\mu_0 = 10^{-4}$.

while not converged **do**

  Update U_{k+1}, V_{k+1} and W_{k+1} by (25), (26) and (27).

  Compute U'_{k+1}, V'_{k+1} and W'_{k+1} by (29) and (30).

  Update X_{k+1} and Z_{k+1} by (31) and (32).

  Update Y_{k+1}^i $(i = 1, \dots, 5)$ by

  $Y_{k+1}^1 = Y_k^1 + \mu_k(U_{k+1} - U'_{k+1})$, $Y_{k+1}^2 = Y_k^2 + \mu_k(V_{k+1} - V'_{k+1})$,
  $Y_{k+1}^3 = Y_k^3 + \mu_k(W_{k+1} - W'_{k+1})$, $Y_{k+1}^4 = Y_k^4 + \mu_k(X_{k+1} - U_{k+1}V_{k+1}^T)$,
  $Y_{k+1}^5 = Y_k^5 + \mu_k(Z_{k+1} - U_{k+1}W_{k+1}^T)$.

  Update μ_{k+1} by $\mu_{k+1} = \rho\mu_k$.

  Check the convergence condition

  $\max\{\|X_{k+1} - U_{k+1}V_{k+1}^T\|_F, \|Z_{k+1} - U_{k+1}W_{k+1}^T\|_F\} < \varepsilon$.

end while

Output: X_{k+1}, Z_{k+1}, U_{k+1}, V_{k+1} and W_{k+1}.

---

3) Updating X_{k+1} and Z_{k+1}: The optimal X_{k+1} and Z_{k+1} are the solutions to the following optimization problems:

$$\begin{aligned} &\min_X \lambda\|T_1 \odot (X - D)\|_F^2 + \mu_k\|X - U_{k+1}V_{k+1}^T + Y_k^4/\mu_k\|_F^2, \\ &\min_Z \gamma\|T_2 \odot (Z - C)\|_F^2 + \mu_k\|Z - U_{k+1}W_{k+1}^T + Y_k^5/\mu_k\|_F^2. \end{aligned}$$

Since the above problems are smooth convex problems, it is easy to show that the optimal solution X_{k+1} is given by

$$\begin{aligned} X_{k+1} = &\widehat{W} \odot (\lambda \overline{W} \odot D + \mu_k U_{k+1}V_{k+1}^T - Y_k^4) \\ &+ \mathcal{P}_\Omega^\perp(U_{k+1}V_{k+1}^T - Y_k^4/\mu_k) \end{aligned} \tag{31}$$

where $\widehat{W}$ is the Hadamard inverse, i.e., $\widehat{W}_{ij} = (\lambda(T_1)_{ij}^2 + \mu_k)^{-1}$ if $(T_1)_{ij} \neq 0$, and $\widehat{W}_{ij} = 0$ otherwise, and $\overline{W}_{ij} = (T_1)_{ij}^2$.

Similar to (31), the optimal solution Z_{k+1} is given by

$$\begin{aligned} Z_{k+1} = &T_2 \odot \left(\frac{\gamma C + \mu_k U_{k+1}W_{k+1}^T - Y_k^5}{\gamma + \mu_k}\right) \\ &+ (J - T_2) \odot (U_{k+1}W_{k+1}^T - Y_k^5/\mu_k) \end{aligned} \tag{32}$$

where J is the $m \times m$ matrix with all entries equal to 1.

Based on the description above, we develop an efficient ADMM algorithm for solving our DTNMC problem (10), as outlined in **Algorithm 2**.

V. Theoretical Analysis

In this section, we provide the recovery guarantees of our double trace norm minimization model and the convergence analysis and complexity analysis for our algorithms.

A. Convergence Analysis

For a proper and lower semi-continuous (PLSC) function, denotes as $h : \mathbb{R}^p \to (-\infty, +\infty]$, the domain of h is defined by $\mathrm{dom}\, h := \{x \in \mathbb{R}^p : h(x) < +\infty\}$. Next, we define the critical points of a PLSC function.

Definition 2 ([42]). *Let h be a PLSC function.*

- *The Frèchet sub-differential of h at x is defined as*

$$\widehat{\partial} h(x) = \{u \in \mathbb{R}^p : \lim_{y \neq x} \inf_{y \to x} \frac{h(y) - h(x) - \langle u, y - x \rangle}{\|y - x\|_2} \geq 0\},$$

and $\widehat{\partial}h(x)=\emptyset$ *if* $x\notin \operatorname{dom} h$.

- *The limiting sub-differential of* h *at* x *is defined as*

$$\partial h(x)=\{u\in\mathbb{R}^n: \exists x^k\to x, h(x^k)\to h(x) \text{ and } u^k\in\widehat{\partial}h(x^k)\to u \text{ as } k\to\infty\}.$$

- x *is a critical (or stationary) point of* h *if* $0\in\partial h(x)$.

Note that our problems (6), (8) and (10) are all challenging non-convex and non-smooth problems. Unlike the non-convex and smooth case [43], the local convergence of our ADMM algorithm (i.e., Algorithm 1) is guaranteed as follows.

Theorem 2. *Let* $\{(U_k, V_k, U'_k, V'_k, X_k, \{Y^i_k\})\}$ *be a sequence generated by Algorithm 1, and if the sequence* $\{Y^3_k\}$ *is bounded, then the following holds:*

1) $\{(U_k, V_k)\}$, $\{(U'_k, V'_k)\}$ *and* $\{X_k\}$ *are all Cauchy sequences.*
2) *If* $\lim_{k\to\infty}\|Y^i_{k+1}-Y^i_k\|_F=0$ $(i=1,2,3)$, *then the accumulation point of the sequence* $\{(U_k, V_k, X_k)\}$ *satisfies the KKT conditions for Problem* (6).

The proof of the theorem can be found in the Supplementary Material. Theorem 2 shows that under mild conditions each sequence generated by Algorithm 1 converges to a critical point (or stationary point). Similar to Algorithm 1, the convergence of Algorithm 2 can also be guaranteed.

B. Recovery Guarantees

According to Definition 1, the double trace norm term in (6), (8) and (10) is the square root of $\|X\|_{**}$. To obtain our conclusion, we first give the following lemma [44].

Lemma 2. $\|X\|_F\le\|X\|_*\le\sqrt{\mathit{rank}(X)}\|X\|_F$.

Theorem 3. *For any* $X\in\mathbb{R}^{m\times n}$ *and rank*$(X)=r$, *the following inequalities hold:*

$$\|X\|_*\le\|X\|_{**}\le r\|X\|_*.$$

The proof of Theorem 3 is provided in the Supplementary Material. According to Theorem 3, it is clear that our double trace norm penalty is much tighter than both the bilinear spectral regularization term in (4) and trace norm penalty in (1) and (2), similar to the trace norm vs. the Frobenius norm.

We further analyze the superiority of our double trace norm over the traditional trace norm, and establish the recovery guarantee for the following general model, which is based on the properties of the general linear operator $\mathcal{A}:\mathbb{R}^{m\times m}\to\mathbb{R}^l$, such as the matrix restricted isometry property (RIP):

$$\min_{U,V}\ \|X\|^{1/2}_{**},\ \ \text{s.t.}, \|\mathcal{A}(UV^T)-b\|_2\le\epsilon \tag{33}$$

where $b\in\mathbb{R}^l$ is linear observations and $\epsilon>0$ is a noise level.

Definition 3. *The matrix RIP constant* $\delta_r(\mathcal{A})$ *of the linear operator* $\mathcal{A}$ *is the smallest value such that*

$$(1-\delta_r(\mathcal{A}))\|X\|^2_F\le\|\mathcal{A}(X)\|^2_2\le(1+\delta_r(\mathcal{A}))\|X\|^2_F$$

holds for all matrices with rank$(X)\le r$.

[45]–[48] provided the matrix RIP-based condition for robust and accurate recovery of low-rank matrices from noisy measurements. Similarly, we also provide the theoretical guarantee for our double trace norm model (33).

TABLE I
COMPARISON OF RECOVERY THRESHOLDS δ_{2r+2}.

Rank	5	10	15	20
Trace norm model	0.4752	0.4648	0.4610	0.4590
Double trace norm model	0.5212	0.4886	0.4771	0.4711

Theorem 4. *Assume* $X_0\in\mathbb{R}^{m\times n}$ *is a true matrix with rank*$(X_0)\le r$ *and the corrupted measurements* $\mathcal{A}(X_0)+e=b$, *where* $\|e\|_2\le\epsilon$. *Let* (U^*, V^*) *be a solution to Problem* (33) *and* $X^*=U^*(V^*)^T$. *If*

$$\delta_{2t}<\frac{2(\sqrt{2}-1)(t/r)^{1.5}}{2(\sqrt{2}-1)(t/r)^{1.5}+1} \tag{34}$$

holds for some integers $t\ge r$, *then*

$$\|X_0-X^*\|_{S_{1/2}}\le C_1\|X_0-X_0^{(r)}\|_{S_{1/2}}+D_1r^{1.5}\epsilon,$$
$$\|X_0-X^*\|_F\le C_2t^{-1.5}\|X_0-X_0^{(r)}\|_{S_{1/2}}+D_2\epsilon$$

where $X_0^{(r)}$ *is a matrix obtained by keeping the* r *largest singular values in the SVD of* X_0, $\|\cdot\|_{S_{1/2}}$ *is the Schatten-*$1/2$ *norm, and* C_1, C_2, D_1, D_2 *depend only on* δ_{2t} *and* t/r.

Proof. We use the inequality $\gamma_{2t}\ge(1+\delta_{2t})/(1-\delta_{2t})$ to state our results in terms of δ_{2t}, where γ_{2t} is the asymmetric RIP constant defined in [22]. Our result in Theorem 4 can be obtained by combining Proposition 2 in [48] and Theorem 1. □

Remark 1. *When* $\epsilon=0$ *and rank*$(X_0)\le r$, *Theorem 4 implies that* X_0 *is a unique solution to the noiseless formulation* (33). *Recall that* $\delta_{2r}<1$ *is a sufficient condition for the success of the rank minimization model [47]. Then for the given* r, *the matrix RIP condition* $\delta_{2r+2}<1$ *is sufficient for robust recovery of matrices with rank at most* r *by using our model* (33). *Substituting* t *with* $r+1$ *in* (34), *we further analyze the recovery threshold* δ_{2r+2} *in Theorem 4. Table I shows that the required threshold* δ_{2r+2} *for our model* (33) *is less restrictive than that for the popular trace norm model as in [1], [44].*

C. Error Bound on Matrix Completion

Although the LRMC problem is a practically important application of Problem (33), the projection operator $\mathcal{P}_\Omega$ does not satisfy the standard RIP condition in general [1], [49]. Therefore, we need to provide the recovery guarantee for performance of our algorithm for solving the LRMC problem (6). Without loss of generality, assume that the observed matrix $D\in\mathbb{R}^{m\times n}$ can be decomposed as a true matrix X_0 of rank $r\le d$ and a random Gaussian noise E, i.e., $D=X_0+E$.

Theorem 5. *Let* $(\widehat{U},\widehat{V})$ *be a critical point of Problem* (6) *with given* d, *and* $m\ge n$. *Then there exists an absolute constant* C_3, *such that with probability at least* $1-2\exp(-m)$,

$$\frac{\|X_0-\widehat{U}\widehat{V}^T\|_F}{\sqrt{mn}}\le\frac{\|E\|_F}{\sqrt{mn}}+C_3\delta\left(\frac{md\log(m)}{|\Omega|}\right)^{1/4}+\frac{\sqrt{d}}{2C_4\lambda\sqrt{|\Omega|}},$$

where $\delta=\max_{i,j}|D_{i,j}|$ *and* $C_4=\frac{\|\mathcal{P}_\Omega(D-\hat{U}\hat{V}^T)\hat{V}\|_F}{\|\mathcal{P}_\Omega(D-\hat{U}\hat{V}^T)\|_F}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2017.2760287, IEEE Transactions on Fuzzy Systems

The proof of Theorem 5 is very similar to that of Theorem 8 [30] and is thus omitted. Theorem 5 shows that when the samples size $|\Omega| \gg md\log(m)$, the second and third terms diminish, and the recovery error is essentially bounded by the "average" magnitude of entries of E as in [30]. That is, to perfectly recover a rank-r matrix of size $m \times n$, only $O(md\log(m))$ observed entries are required by our double trace norm minimization model (6), which is significantly lower than $O(mr\log^2(m))$ in standard matrix completion theories [12], [44], [50].

Remark 2. *Since the Schatten quasi-norm minimization problem is non-convex, non-smooth and non-Lipschitz, the recovery guarantees in [45]–[47] are naturally based on the global optimal solution, but finding it is in fact very difficult and not guaranteed. To the best of our knowledge, our recovery guarantee analysis is the first one for the solution generated by associated algorithms, not for the global optima.*

D. Complexity Analysis

For the DTNM problem (6), the per-iteration cost of Algorithm 1 is dominated by performing matrix multiplication operators in (14), (15) and (20), and their complexity is $O(mnd)$. Thus, the per-iteration complexity of Algorithm 1 is $O(mnd)$ (usually $d \ll \min(m,n)$). For the DTNMC problem (10), the complexity of performing SVD and some multiplication operators is $O(mnd+m^2d)$. The complexity analysis shows that our algorithms significantly reduce the computational cost compared with Schatten quasi-norm minimization algorithms [3], [21], [26]. Thus, in practice our method is fast and scales well to handle large-scale problems.

VI. Experimental Evaluation

In this section, we evaluate our DTNM method by comparing with several TNM methods, the SRMF method[1], and one SNM method. Moreover, we also evaluate the effectiveness and efficiency of our DTNM and DTNMC methods for social recommendation tasks on real-world problems. All experiments were performed on an Intel Xeon E7-4830V2 2.20GHz CPU with 64G RAM.

A. Model Comparison for Matrix Completion

Synthetic ground-truth matrices $X_0 \in \mathbb{R}^{m\times n}$ with rank r are generated by the procedure in [6], [30]. The experiments were conducted on noisy matrices with the noise factor, $nf = 0.05$ or 0.1, where the observed subset is corrupted by i.i.d. standard Gaussian random variables as in [6]. Only 1.5% or 3.0% entries are sampled uniformly at random as observed data. The size of matrices is $1,000 \times 1,000$, and their rank is set to 5. We use the relative standard error (RSE $:= \|X - X_0\|_F / \|X_0\|_F$) as the evaluation measure, where X denotes the recovered matrix. The regularization parameter is set to $\lambda = \sqrt{\max(m,n)}$.

We compared DTNM with APG [2], ALT [51], SRMF, and IRucLq [3]. The average RSE results over 20 independent runs of APG, ALT, SRMF, IRucLq ($p \in \{0.1, 0.2, \ldots, 1\}$) and our DTNM method on noisy random matrices are shown in Fig. 1, from which we can observe that:

- As a scalable alternative to the trace norm, SRMF with relatively small ranks (e.g., the rank parameter is set to $\lfloor 1.25r \rfloor$ as in DTNM) often obtains more accurate solutions than the trace norm counterparts, APG and ALT.
- If p is chosen from the range of $\{0.5, 0.6, 0.7\}$, IRucLq usually outperforms APG, ALT and SRMF in terms of RSE. Otherwise, IRucLq sometimes performs much worse than the other methods, especially $p = 0.1$.
- The RSE of our DTNM method under all of these settings is consistently much better than that of the other methods. This result clearly justifies the usefulness of the proposed double trace norm penalty.

We also conducted some experiments on noisy matrices of size 100×100 with $nf = 0.1$, and report the RSE results of all algorithms with different sampling rates (SR) in Fig. 2(a). We can see that the performance of all the methods becomes worse as SR decreases. DTNM outperforms the other methods in all the settings and has much greater advantage over them in cases when SR is relatively small, e.g., 5%. Moreover, we report the running time as the size of noisy random matrices increases, as shown in Fig. 2(b), where SR is set to 2.5%. The running time of IRucLq increases dramatically when the size of matrices increases, and it could not produce results within 48 hours when the size of matrices is $10^4 \times 10^4$. In contrast, DTNM is much faster than the other methods. Especially, DTNM can be $500\times$ faster than IRucLq. This further justifies that DTNM has very good scalability and can address large-scale problems. As APG uses the PROPACK package [8] to compute a partial SVD, it sometimes runs slightly faster than ALT.

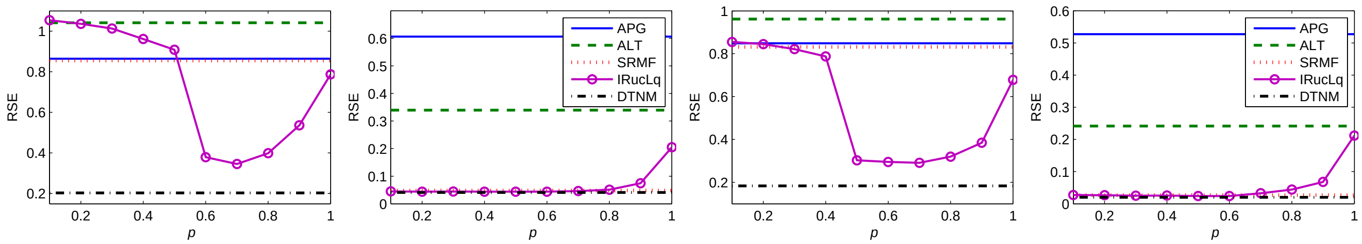
B. Collaborative Filtering

In order to evaluate our method for collaborative filtering tasks, some matrix completion experiments were conducted on the four data sets[2]: MovieLens100K with 100K ratings, MovieLens1M with 1M ratings, MovieLens10M with 10M ratings and MovieLens20M with 20M ratings, as shown in the Supplementary Material. We randomly choose 90% as the training set and the remaining as the testing set, and report the average results obtained over 10 independent runs. Besides those methods used above, we also compared our method with the trace norm solver, IMPUTE [4], one of the fastest methods, LMaFit [14], and two manifold optimization methods: OptSpace [12] and RTRMC [13]. The stopping tolerance for all algorithms is set to $\varepsilon = 10^{-4}$, and the settings for regularization parameters of different algorithms are listed in the Supplementary Material.

We use the root mean squared error (RMSE) as the evaluation measure, and report the average testing RMSE results of all those methods with ranks varying from 5 to 15 in Fig. 3. Moreover, we also report more experimental results of all those

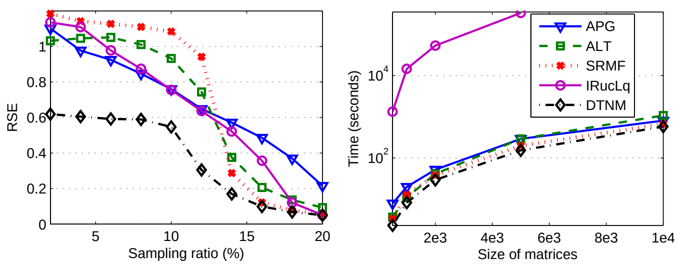
[1]For fair comparison, the SRMF problem (4) is solved by the proposed ADMM algorithm, which, as well as its convergence analysis, is provided in the Supplementary Material.

[2]http://www.grouplens.org/node/73



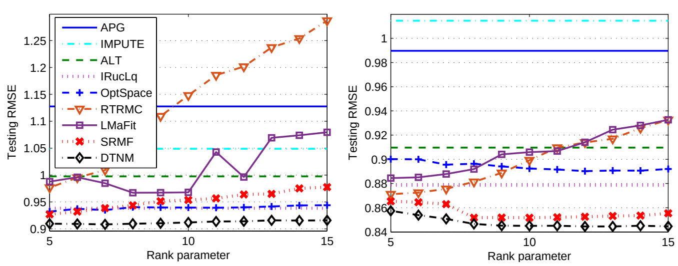
(a) 1.5% SR and $nf = 0.10$ (b) 3.0% SR and $nf = 0.10$ (c) 1.5% SR and $nf = 0.05$ (d) 3.0% SR and $nf = 0.05$

Fig. 1. The RSE results of APG [2], ALT [51], SRMF, IRucLq [3], and our DTNM method on noisy matrices of size $1,000 \times 1,000$.

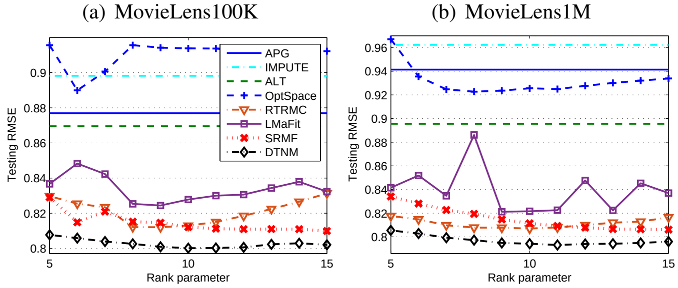


(a) RSE vs. sampling ratio (b) Time vs. sizes of matrices

Fig. 2. The recovery results (RSE and running time (seconds)) of APG [2], ALT [51], SRMF, IRucLq [3], and our DTNM method.



(a) MovieLens100K (b) MovieLens1M



(c) MovieLens10M (d) MovieLens20M

Fig. 3. Evolution of the testing RMSE of different methods with varying rank parameters (from 5 to 15).

methods on the four data sets in the Supplementary Material. From all the results, we can observe that:

1) The matrix factorization methods, RTRMC, LMaFit, SRMF and DTNM, except OptSpace, significantly outperform the trace norm solvers including APG, IMPUTE and ALT in terms of RMSE, especially on the two larger datasets, as shown in Figs. 3(c) and 3(d).
2) In most cases, the sophisticated matrix factorization methods, except OptSpace, outperform LMaFit without any regularization term. This suggests that those regularized models can effectively alleviate over-fitting problems.
3) The testing RMSE of DTNM varies only slightly when the rank parameter increases. In contrast, the testing RMSE of all the other matrix factorization methods changes dramatically. This further means that DTNM performs more robust than them in terms of this parameter.
4) We reported the best results of IRucLq with all choices of p in $\{0.1, \ldots, 1\}$ in Fig. 3. IRucLq performs much better than the three trace norm solvers. However, IRucLq could not run on the two largest datasets due to runtime exceptions, and is more than 50 times slower than DTNM.
5) Our DTNM method under all of the settings consistently outperforms the other methods in terms of RMSE. This confirms that our double trace norm regularized model can provide a good estimation of a low-rank matrix, even though from only a few observations.

C. Fuzzy Recommender System

Finally, we conducted some experiments to investigate the effects of social context, and chose the two real-world data sets[3] used in [17], Epinions and Ciao, to evaluate DTNMC. Some statistics of the two data sets are given in the Supplementary Material. The users in both data sets rated products with scores from 1 to 5, and they also established social relations with others. For each data set, we chose 50%, 70% or 90% as training data and the remaining as testing data.

We compared DTNMC with several state-of-the-art recommender methods: PMF [52], SoRec [15], SoReg [16] and LOCABAL [17], where PMF only utilizes rating information. Both SoRec and SoReg only exploit the social network information, while DTNMC and LOCABAL can exploit both the user reputation and social network information. The experimental results are reported in Table II, from which we observe that DTNMC and LOCABAL consistently outperform the other methods. This further confirms that both the social network and user reputation information can improve the recommender performance. More details on the effects of both types of social information on the performance of DTNMC and LOCABAL are discussed as follows.

We investigate the effects of both the social network (S) and the user reputation (R) information on DTNMC and LOCABAL. Note that DTNMC/S/R (i.e., DTNM) and LOCABAL/S/R denote DTNMC and LOCABAL without both the

[3] http://www.cse.msu.edu/~tangjili/trust.html

TABLE II
COMPARISON OF DIFFERENT METHODS ON EPINIONS AND CIAO.

Databases	Training set	PMF	SoRec	SoReg	LOCABAL	DTNMC
Epinions	50%	1.1736	1.1366	1.1285	1.1083	**1.1037**
	70%	1.1482	1.1137	1.1090	1.0857	**1.0726**
	90%	1.1381	1.1025	1.0967	1.0746	**1.0653**
Ciao	50%	1.1925	1.1579	1.1473	1.1270	**1.1185**
	70%	1.1834	1.1485	1.1272	1.1074	**1.1021**
	90%	1.1796	1.1383	1.1164	1.1002	**1.0912**

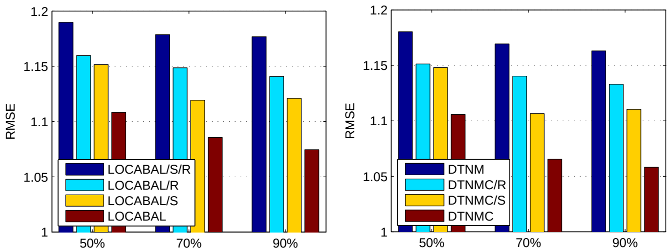


Fig. 4. The impact of the social network and user reputation information on LOCABAL (left) and DTNMC (right).

social network and user reputation information; DTNMC/R and LOCABAL/R denote DTNMC and LOCABAL without user reputation information; DTNMC/S and LOCABAL/S denote DTNMC and LOCABAL without social network information, respectively. Since we have similar observations on the Ciao date set, we only show the experimental results on the Epinions date set in Fig. 4. It is clear that both social network and user reputation information can help improve the accuracy of recommender systems. Moreover, DTNMC consistently outperforms LOCABAL in all these settings, which implies that DTNMC can make better use of both user reputation and social network information than LOCABAL.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel fuzzy double trace norm minimization (DTNM) method for LRMC problems. We first presented a double trace norm minimization model, which is in essence a tractable Schatten-$1/2$ quasi-norm problem. In our model, we updated two much smaller factor matrices to replace the repetitively calculating SVD of a larger matrix as in Schatten norm minimization methods. Therefore, it can be several orders of magnitude faster, and scales well to handle large-scale problems. Then we also extended our model to incorporate the social contextual information, such as the user social relationship and user reputation information, for social recommendation. Finally, we developed two efficient ADMM algorithms for solving the proposed problems. Extensive results on both synthetic and real-world data sets verified that our double trace norm penalty performs better than both bilinear spectral regularization and trace norm penalty.

For future work, we are interested in exploring ways to regularize our model with other auxiliary information as in [7], [53], such as semantic information contained in social network [19] and leverage scores [36]. Moreover, our method can be extended to various robust low-rank tensor recovery and completion problems as in [54].

ACKNOWLEDGMENTS

The authors are supported in part by Grants (CUHK 14206715 & 14222816) from the Hong Kong RGC, ITF 6904079, Grant 3132821 funded by the Research Committee of CUHK. Da Yan is supported by NSF DGE-1723250 and Microsoft Azure Research Award CRM: 0740132.

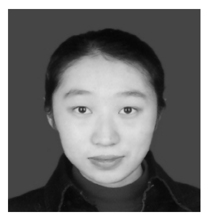
REFERENCES

[1] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.
[2] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems," *Pac. J. Optim.*, vol. 6, pp. 615–640, 2010.
[3] M. Lai, Y. Xu, and W. Yin, "Improved iteratively rewighted least squares for unconstrained smoothed ℓ_p minimization," *SIAM J. Numer. Anal.*, vol. 51, no. 2, pp. 927–957, 2013.
[4] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *J. Mach. Learn. Res.*, vol. 11, pp. 2287–2322, 2010.
[5] T. Ngo and Y. Saad, "Scaled gradients on Grassmann manifolds for matrix completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1421–1429.
[6] F. Shang, Y. Liu, and J. Cheng, "Scalable algorithms for tractable Schatten quasi-norm minimization," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 2016–2022.
[7] Y. Liu, F. Shang, H. Cheng, and J. Cheng, "A Grassmannian manifold algorithm for nuclear norm regularized least squares problems," in *Proc. 30th Conf. Uncert. Artif. Intell.*, 2014, pp. 515–524.
[8] R. Larsen, "PROPACK-Software for large and sparse SVD calculations," *Available from http://sun.stanford.edu/srmunk/PROPACK/*, 2005.
[9] J. Cai, E. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.
[10] Y. Wang and H. Xu, "Stability of matrix factorization for collaborative filtering," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 417–424.
[11] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[12] R. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Trans. Inform. Theory*, vol. 56, no. 6, pp. 2980–2998, Jun. 2010.
[13] N. Boumal and P.-A. Absil, "RTRMC: A Riemannian trust-region method for low-rank matrix completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 406–414.
[14] 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.*, vol. 4, no. 4, pp. 333–361, 2012.
[15] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 17th Int. Conf. Inform. Knowl. Manag.*, 2008, pp. 931–940.
[16] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proc. 4th Int. Conf. Web Search Data Min.*, 2011, pp. 287–296.
[17] J. Tang, X. Hu, H. Gao, and H. Liu, "Exploiting local and global social context for recommendation," in *Proc. 6th Int. Conf. Web Search Data Min.*, 2013, pp. 2712–2718.
[18] S. Purushotham, Y. Liu, and C. J. Kuo, "Collaborative topic regression with social matrix factorization for recommendation systems," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 759–766.
[19] M. Jiang, P. Cui, F. Wang, W. Zhu, and S. Yang, "Scalable recommendation with social contextual information," *IEEE Trans. Knowledge and Data Eng.*, vol. 26, no. 11, pp. 2789–2802, Nov. 2014.
[20] D. Wu, J. Lu, and G. Zhang, "A fuzzy tree matching-based personalized e-learning recommender system," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 2412–2426, Dec. 2015.
[21] F. Nie, H. Huang, and C. Ding, "Low-rank matrix recovery via efficient Schatten p-norm minimization," in *Proc. 26th AAAI Conf. Artif. Intell.*, 2012, pp. 655–661.
[22] S. Foucart and M.-J. Lai, "Sparsest solutions of underdetermined linear systems via ℓ_p-minimization for $0<q\leq 1$," *Appl. Comput. Harmon. A.*, vol. 26, pp. 397–407, 2009.
[23] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Guntuk, "Iteratively reweighted least squares minimization for sparse recovery," *Commun. Pure Appl. Math.*, vol. 63, pp. 1–38, 2010.

[24] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations with missing data," in *Proc. SIAM Int. Conf. Data Min.*, 2010, pp. 701–711.
[25] K. Mohan and M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," *J. Mach. Learn. Res.*, vol. 13, pp. 3441–3473, 2012.
[26] C. Lu, J. Tang, S. Yan, and Z. Lin, "Generalized nonconvex nonsmooth low-rank minimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2014, pp. 4130–4137.
[27] F. Shang, Y. Liu, H. Tong, J. Cheng, and H. Cheng, "Robust bilinear factorization with missing and grossly corrupted observations," *Inform. Sciences*, vol. 307, pp. 53–72, 2015.
[28] N. Srebro, J. Rennie, and T. Jaakkola, "Maximum-margin matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 1329–1336.
[29] F. Shang, J. Cheng, Y. Liu, Z.-Q. Luo, and Z. Lin, "Bilinear factor matrix norm minimization for robust PCA: Algorithms and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017.
[30] F. Shang, Y. Liu, and J. Cheng, "Tractable and scalable Schatten quasi-norm approximations for rank minimization," in *Proc. 19th Int. Conf. Artif. Intell. Statist.*, 2016, pp. 620–629.
[31] Z. Xu, X. Chang, F. Xu, and H. Zhang, "$L_{1/2}$ regularization: A thresholding representation theory and a fast solver," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1013–1027, Jul. 2012.
[32] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1033–1041.
[33] D. Seno and B. Lukas, "The equity effect of product endorsement by celebrities: A conceptual framework from a co-branding perspective," *Eur. J. Marketing*, vol. 41, pp. 121–134, 2007.
[34] R. Song, L. Korba, and G. Yee, *Trust in E-services: Technologies, Practices and Challenges*. Idea Group Publishing, 2007.
[35] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," Stanford Digital Library Technologies Project, Tech. Rep., 1999.
[36] S. Bhojanapalli, P. Jain, and S. Sanghavi, "Tighter low-rank approximation via sampling the leveraged element," in *Proc. 26th ACM-SIAM Symp. Discrete Algorithms*, 2015, pp. 902–920.
[37] M. McPherson, L. Smith-Lovin, and J. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociol.*, vol. 27, no. 1, pp. 415–444, 2001.
[38] J. Tang, H. Gao, and H. Liu, "mTrust: Discerning multi-faceted trust in a connected world," in *Proc. 5th Int. Conf. Web Search Data Min.*, 2012, pp. 93–102.
[39] 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.*, vol. 3, no. 1, pp. 1–122, 2011.
[40] F. Shang, Y. Liu, J. Cheng, and H. Cheng, "Robust principal component analysis with missing data," in *Proc. 23rd Int. Conf. Inform. Knowl. Manag.*, 2014, pp. 1149–1158.
[41] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 612–620.
[42] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Math. Program.*, vol. 146, pp. 459–494, 2014.
[43] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Front. Math. Chin.*, vol. 7, no. 2, pp. 365–384, 2012.
[44] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
[45] A. Rohde and A. B. Tsybakov, "Estimation of high-dimensional low-rank matrices," *Ann. Statist.*, vol. 39, no. 2, pp. 887–930, 2011.
[46] M. Zhang, Z. Huang, and Y. Zhang, "Restricted p-isometry properties of nonconvex matrix recovery," *IEEE Trans. Inform. Theory*, vol. 59, no. 7, pp. 4316–4323, Jul. 2013.
[47] S. Oymak, K. Mohan, M. Fazel, and B. Hassibi, "A simplified approach to recovery conditions for low rank matrices," in *Proc. IEEE Int. Symp. Inform. Theory*, 2011, pp. 2318–2322.
[48] M. Mohammadi, M. Zadeh, and M. Skoglund, "Performance guarantees for Schatten-p quasi-norm minimization in recovery of low-rank matrices," *Signal Proces.*, vol. 114, pp. 225–230, 2015.
[49] P. Jain, R. Meka, and I. Dhillon, "Guaranteed rank minimization via singular value projection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 937–945.
[50] E. Candès and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, Jun. 2010.
[51] C.-J. Hsieh and P. Olsen, "Nuclear norm minimization via active subspace selection," in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. 575–583.
[52] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1257–1264.
[53] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowledge Discov. Data Mining*, 2008, pp. 426–434.
[54] Y. Liu, F. Shang, W. Fan, J. Cheng, and H. Cheng, "Generalized higher-order orthogonal iteration for tensor decomposition and completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1763–1771.

Fanhua Shang (M'14) received the Ph.D. degree in Circuits and Systems from Xidian University, Xi'an, China, in 2012.

He is currently a Research Associate with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. From 2013 to 2015, he was a Post-Doctoral Research Fellow with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. From 2012 to 2013, he was a Post-Doctoral Research Associate with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA. His current research interests include machine learning, data mining, pattern recognition, and computer vision.



Yuanyuan Liu received the Ph.D. degree in Pattern Recognition and Intelligent System from Xidian University, Xi'an, China, in 2013.

She is currently a Post-Doctoral Research Fellow with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. Prior to that, she was a Post-Doctoral Research Fellow with the Department of Systems Engineering and Engineering Management, CUHK. Her current research interests include image processing, pattern recognition, and machine learning.

James Cheng is an Assistant Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. His current research interests include distributed computing systems, large-scale network analysis, temporal networks, and big data.

Da Yan is an Assistant Professor with the Department of Computer Science at the University of Alabama at Birmingham. He is the winner of the 2015 Hong Kong Young Scientist Award, and he also led the BigGraph@CUHK project (see http://www.cse.cuhk.edu.hk/systems/graph). His research interests include Big Data analytics, distributed systems, graph data management, geospatial data management, and data mining.