Feature and Nuclear Norm Minimization for Matrix Completion

Mengyun Yang, Yaohang Li, and Jianxin Wang

Abstract—Matrix completion, whose goal is to recover a matrix from a few entries observed, is a fundamental model behind many applications. Our study shows that, in many applications, the to-be-complete matrix can be represented as the sum of a low-rank matrix and a sparse matrix associating with side information matrices. The low-rank matrix depicts the global patterns while the sparse matrix characterizes the local patterns, which are often described by the side information. Accordingly, to achieve high-quality matrix completion, we propose a Feature and Nuclear Norm Minimization (FNNM) model. The rationale of FNNM is to employ transductive completion to generalize the global pattern and inductive completion to recover the local pattern. Alternative minimization algorithm based on fixed-point iteration is developed to numerically solve the FNNM model. FNNM has demonstrated promising results on a variety of applications, including movie recommendation, drug-target interaction prediction, and multi-label learning, consistently outperforming the state-of-the-art matrix completion algorithms.

Index Terms—Matrix completion, nuclear norm minimization, side information, movie recommendation, drug-target interaction prediction, multi-label learning.

1 Introduction

The matrix completion problem is defined as recovering the missing ones from the observed entries in an incomplete matrix. In recent years, many machine learning applications, including computer vision [1], [2], collaborative filtering [3], [4], recommendation systems [5], [6], [7], multi-class learning [8], [9], [10], and bioinformatics [11], [12], [13], [14], [15], have benefited from finding solutions of the matrix completion problem. In the past decade, many methods have been proposed to solve the matrix completion problem. In general, these methods can be categorized into two classes: transductive matrix completion and inductive matrix completion. Both adopt the low-rank matrix assumption.

Transductive completion assumes that the to-be-completed matrix is of low-rank. Instead of directly optimizing the rank function, which is well-known to be NP-hard, the transductive completion methods [16], [17], [18], [19] are often relaxed to minimizing the nuclear norm, defined as the sum of all singular values of a matrix. Theoretical studies show that nuclear norm is a convex surrogate of the rank function [20], which enables the design of efficient convex matrix completion algorithms, including Singular Value Thresholding (SVT) [16], Accelerated Proximal Gradient (APG) [18], Fixed-Point Continuation (FPC) [17], and many others. [21], [22], [23] show that nuclear norm minimization can recover a missing matrix perfectly from sufficient known entries under some general settings. Serving the purpose of better approximating the rank function, a few non-convex extensions of the nuclear norm minimization, including Truncated Nuclear Norm Regularization (TNNR) [2], Weighted Nuclear Norm Minimization (WNNM) [24], and joint Schatten p-norm and l_p-norm minimization [25], have also been proposed.

On the other hand, the inductive matrix completion methods are designed to take advantage of the large amount of side information about row or column objects often available in applications. Giving a couple of examples, in movie recommendation, user profiles and movie descriptions are provided besides the existing rating matrix; in drug-target interaction prediction, chemical composition of drugs and sequence information of target proteins are also available in addition to the interaction matrix. In particular, the side information can effectively address the well-known "cold-start" problem in transductive completion. Inductive completion models assume a low-rank latent matrix to associate the side information. [11], [26], [27] address the inductive matrix completion problem via non-convex matrix factorization. [28], [29] propose convex inductive matrix completion based on nuclear norm minimization. Maxide [28] is developed to reduce sample complexity and improve scalability of inductive matrix completion. SIMC [29] focuses on the interpretability of the interaction matrix connecting side information.

In many applications, the to-be-complete matrix can be well approximated as the sum of a low-rank matrix and a sparse matrix connecting the side information matrices. Typically, the low-rank matrix represents the global patterns while the sparse matrix describes the local patterns. Let us use the user-movie association matrix in movie recommendation as an example. The low-rank part encodes the general user behaviors, i.e., if the users share similar ratings

- *M. Yang and J. Wang are with the School of Computer Science and Engineering, Central South University, Changsha 410083, China. E-mail: jxwang@mail.csu.edu.cn.*
- *M. Yang is also with the Provincial Key Laboratory of Informational Service for Rural Area of Southwestern Hunan, Shaoyang University, Shaoyang 422000, China. E-mail: mengyunyang@csu.edu.cn.*
- *Y. Li is with the Department of Computer Science, Old Dominion University, Norfolk 23529, USA. E-mail: yaohang@cs.odu.edu.*

Corresponding author: Jianxin Wang.

in the past on the same set of movies, then they will likely rate the other movies similarly. At the same time, each user has his/her personalized preferences and each movie has its own uniqueness, which lead to certain sparsity in mapping users to movies. i.e., classic movies are often favorable to an elderly user while a child likes cartoon movies. This can be represented as sparse associations between user features (such as age) and movie features (such as cartoon and classic), which in turns results in the sparsity of the association matrix. The sparse part is correlated to the side information, which is also adopted in [29] and has demonstrated certain success. This inspires us to design a matrix completion model that can recover the global patterns as well as the local patterns.

In this work, we propose a novel Feature and Nuclear Norm Minimization (FNNM) model, which combines both transductive and inductive matrix completion. The fundamental idea of FNNM is to recover the low-rank part by nuclear norm minimization and the sparse part by sparse inductive matrix completion integrating side information. FNNM is able to deal with the extreme cases when there is scarce observations are available or the side information is unreliable. An efficient alternative minimization algorithm is designed to efficiently solve the FNNM model. We demonstrate the effectiveness of FNNM in a variety of applications including movie recommendation, drug-target interaction prediction, and multi-label learning. The source code and datasets are freely available at https://github.com/BioinformaticsCSU/FNNM.

The remainder of this paper is organized as follows: We will give a brief description of related work in section 2. In section 3, we present our model (FNNM) and optimization scheme. Experimental results on synthetic and real-world datasets are analyzed in section 4. The computational time of different methods are compared in section 5. Finally, we summarize our conclusions in section 6.

2 Related Work

In this section, we describe six state-of-the-art matrix completion models, which will be compared with FNNM. By an abuse of notation, we denote λ as the harmonic parameter balancing different terms in these models, but it is important to note that each model has its own preferred λ value.

Singular Value Thresholding algorithm (SVT) [16] solves the following model

$$\min_X \lambda \|X\|_* + \frac{1}{2}\|X\|_F^2, \ \ subject\ to\ \ \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(A), \quad (1)$$

where $\|\cdot\|_*$ denotes the nuclear norm, which can lead low-rank property, $\|\cdot\|_F$ represents Frobenius norm, A is the to-be-complete matrix, Ω is a set containing the index pairs (i,j) of all known entries in A, and $\mathcal{P}_\Omega(X)$ is the projection operator projecting matrix X onto Ω such that

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

The SVT algorithm considers the global pattern of the matrix A, and recovers the missing entries in the matrix by minimizing the convex nuclear norm.

Fixed-Point Continuation algorithm (FPC) [17] is designed to address the nuclear norm regularized least squares problem:

$$\min_X \lambda \|X\|_* + \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(A)\|_F^2. \quad (2)$$

Compared to SVT, by putting the constraint term as an error term to the optimization objective function, FPC is able to handle noisy matrix completion problem.

Weighted Nuclear Norm Minimization algorithm (WNNM) [24] minimizes the weighted nuclear norm such that

$$\min_X \|X\|_{w,*}, \ \ subject\ to\ \ \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(A), \quad (3)$$

where $\|X\|_{w,*} = \sum_i |w_i \sigma_i(X)|$ is the weighted nuclear norm and w_i is a non-negative weight assigned to singular value $\sigma_i(X)$. The weighted nuclear norm allows flexibility in optimizing specific singular values.

Inductive Matrix Completion model (IMC) [11] is designed to incorporate side information as feature matrices U and V into matrix completion such that

$$\min_{M,N} \lambda(\|M\|_F^2 + \|N\|_F^2) + \frac{1}{2}\left\|\mathcal{P}_\Omega(UMN^TV^T) - \mathcal{P}_\Omega(A)\right\|_F^2. \quad (4)$$

The interaction matrix that connects U and V is factorized as MN^T and is assumed to be low-rank. To obtain a good performance, the feature matrices U and V should be orthogonal and satisfy the following conditions: $Col(A) \subset Col(U), Row(A) \subset Col(V)$, where $Col(\cdot)$ and $Row(\cdot)$ represent the column and row space, respectively.

Instead of factorizing the interaction matrix, *the speedup matrix completion with side information* model (Maxide) [28] employs the nuclear norm to regularize the interaction matrix such that

$$\min_X \lambda \|X\|_* + \frac{1}{2}\left\|\mathcal{P}_\Omega(UXV^T) - \mathcal{P}_\Omega(A)\right\|_F^2. \quad (5)$$

The accelerated gradient descent method [30] is used to achieve fast convergence.

The Sparse Interactive model for Matrix Completion with side information (SIMC) [29] is formulated as follows:

$$\begin{aligned} &\min_{M,N} \lambda_1 \|M\|_* + \lambda_2 \|N\|_1 + \frac{1}{2}\left\|U^TNV - M\right\|_F^2, \\ &subject\ to\ \ \mathcal{P}_\Omega(M) = \mathcal{P}_\Omega(A), \end{aligned} \quad (6)$$

where $\|\cdot\|_1$ denotes an L_1 norm. A linearized alternating direction method of multipliers (LADMM) algorithm [31] is employed to solve the SIMC model.

SVT, FPC, and WNNM are transductive matrix completion methods, which only consider the entries of the incomplete matrix, while IMC, Maxide, and SIMC are inducitve matrix completion methods, where side information is incorporated. In general, the transductive matrix completion methods suffer from the "cold-start" problem, i.e., the matrix completion algorithms have difficulty in handling rows or columns without any previously known entries. With the help of side information, inductive matrix completion can be used to address the "cold-start" problem, but their performance heavily relies on the quality of the side information.

3 FNNM MODEL AND OPTIMIZATION

Assuming that the to-be-complete matrix A is composed of a low-rank part and a sparse part associating with side information, representing the global patterns and the local patterns, respectively, in this work, we propose an FNNM model combining both transductive completion and inductive completion. The fundamental idea of FNNM is to balance the global patterns and the local patterns associating with the side information. More specifically, the FNNM model attempts to construct a complete matrix of $M + UNV^T$ to approximate A, where matrix M aims at maintaining low-rank property while N is a sparse interaction matrix associating the side information matrices U and V. As a result, the FNNM model is formulated as

$$\min_{M,N} \lambda_1 \|M\|_* + \lambda_2 \|N\|_1 + \frac{1}{2} \left\| \mathcal{P}_\Omega(M + UNV^T) - \mathcal{P}_\Omega(A) \right\|_F^2, \tag{7}$$

where $\left\| \mathcal{P}_\Omega(M + UNV^T) - \mathcal{P}_\Omega(A) \right\|_F^2$ is the error term, M is regularized by a nuclear norm to keep low-rank property, N is regularized by the L_1 norm to maintain sparsity, and λ_1 and λ_2 are the harmonic parameters to balance the nuclear norm, L_1 norm, and the error term.

At first glance, FNNM is similar to SIMC with both sparse and low-rank assumption. In fact, they are fundamentally different. SIMC assumes that the completed matrix is low-rank, while at the same time it can be approximated as the product of a sparse matrix with the side information matrices. In contrast, our FNNM model assumes that the completed matrix is not necessarily a low-rank or sparse matrix, but is made up of a low-rank matrix and a sparse matrix associating with feature matrices. Our analysis shown in Figure 1 agrees well with our assumptions in a variety of matrix completion applications. The horizontal axes of the plots in Figure 1 represent the sparsity of the $\hat{N}$, where $\hat{N}$ is obtained by sparsifying $U^T AV$ under a threshold to control its sparsity. The vertical axes represent the nuclear norm of the leftover matrix R when $U^T AV$ is removed from A, i.e., $R = \mathcal{P}_\Omega(A - U^T AV)$. One can find that $\|R\|_*$ drops significantly when the sparsity of $\hat{N}$ reaches $\sim 10\%$; however, $\|R\|_*$ turns stable when $\hat{N}$ becomes denser. This strongly indicates that we can use the sum of a low-rank matrix (or a matrix with small nuclear norm value) and a sparse matrix interacting with side information. It is also interesting to notice that, when λ_2 in SIMC (6) and FNNM (7) tends to be positive infinity and N is forced to be $\mathbf{0}$, SIMC degenerates to the SVT model (1) while FNNM degenerates to the FPC model (2). As discussed in section 2, FPC is capable of handling noisy data by relaxing the constraint term to an error term in the objective function and hence FNNM inherits the robustness of FPC.

On the one hand, the transductive completion methods recover the matrix solely based on observations, which have difficulty in dealing with the "cold-start" problem. On the other hand, the inductive completion methods mainly rely on the side information, which can be misled when the features in the side information are noisy. FNNM has the advantage of balancing transductive and inductive matrix completion methods to generate high-quality completed matrix.

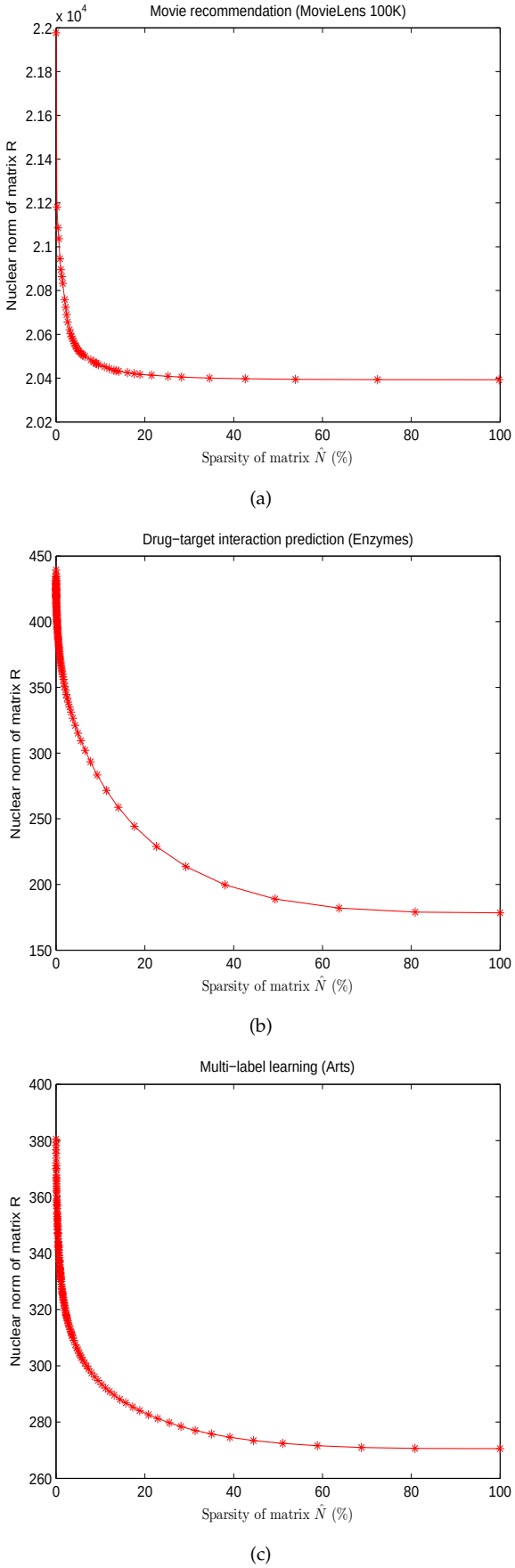


Fig. 1. Sparsity and low-rank properties of to-be-complete matrices in applications of (a) movie recommendation, (b) drug-target interaction prediction, and (c) multi-label learning.

To solve the optimization problem (7) in the FNNM model, we propose an alternative minimization scheme. At each iteration, we fix N and apply a fixed-point iterative method for nuclear norm minimization [17] to obtain M. Then, we fix M alternatively and apply the fixed-point iterative method for L_1 norm minimization [32] to solve N. The alternative minimization scheme includes the steps of computing M_{k+1} and N_{k+1}:

• *Computing* M_{k+1}: Fixing N_k to solve $\min_M \lambda_1 \|M\|_* + \frac{1}{2}\left\|\mathcal{P}_\Omega(M + UN_kV^T) - \mathcal{P}_\Omega(A)\right\|_F^2$. Since this objective function is convex, M^* is the optimal solution if and only if

$$\mathbf{0} \in \lambda_1 \partial \|M^*\|_* + \mathcal{P}_\Omega(M^* + UN_kV^T) - \mathcal{P}_\Omega(A). \quad (8)$$

(8) is equivalent to

$$\mathbf{0} \in \lambda_1\mu_1\partial \|M^*\|_* + M^* - (M^* - \mu_1\mathcal{P}_\Omega(M^* + UN_kV^T - A)), \quad (9)$$

for any $\mu_1 > 0$. We set $E_k = M^* - \mu_1\mathcal{P}_\Omega(M^* + UN_kV^T - A)$ and then (9) is reduced to

$$\mathbf{0} \in \lambda_1\mu_1\partial \|M^*\|_* + M^* - E_k.$$

Clearly, M^* is the optimal solution of $\min_M \lambda_1\mu_1 \|M\|_* + \frac{1}{2}\|M - E_k\|_F^2$. The fixed-point iterative scheme for computing M_{k+1} is derived as

$$\begin{cases} E_k = M_k - \mu_1\mathcal{P}_\Omega(M_k + UN_kV^T - A) \\ M_{k+1} = \mathcal{D}_{\lambda_1\mu_1}(E_k). \end{cases} \quad (10)$$

Here $\mathcal{D}_\tau(X)$ is the singular value shrinkage operator [16], [17] on matrix X defined as

$$\mathcal{D}_\tau(X) = \sum_{i=1}^{\sigma_i \geq \tau} (\sigma_i - \tau) u_i v_i^T,$$

where σ_i is the ith singular values of X larger than threshold τ, while u_i and v_i are the left and right singular vectors corresponding to σ_i, respectively.

• *Computing* N_{k+1}: Fixing M_{k+1} to solve N_{k+1}. N^* is the optimal solution to $\min_N \lambda_2 \|N\|_1 + \frac{1}{2}\left\|\mathcal{P}_\Omega(M_{k+1} + UNV^T) - \mathcal{P}_\Omega(A)\right\|_F^2$, if and only if

$$\mathbf{0} \in \lambda_2\partial \|N^*\|_1 + U^T\mathcal{P}_\Omega(M_{k+1} + UN^*V^T - A)V.$$

The above formula is equivalent to

$$\begin{aligned} \mathbf{0} \in & \lambda_2\mu_2\partial \|N^*\|_1 + N^* - (N^* - \mu_2U^T\mathcal{P}_\Omega(M_{k+1} \\ & + UN^*V^T - A)V) \end{aligned} \quad (11)$$

for any $\mu_2 > 0$. We set $F_k = N^* - \mu_2U^T\mathcal{P}_\Omega(M_{k+1} + UN^*V^T - A)V$ and then (11) becomes

$$\mathbf{0} \in \lambda_2\mu_2\partial \|N^*\|_1 + N^* - F_k.$$

N^* is the optimal solution of

$$\min_N \lambda_2\mu_2 \|N\|_1 + \frac{1}{2}\|N - F_k\|_F^2.$$

As a result, the fixed-point iterative scheme for computing N_{k+1} becomes:

$$\begin{cases} F_k = N_k - \mu_2U^T\mathcal{P}_\Omega(M_{k+1} + UN_kV^T - A)V \\ N_{k+1} = \mathcal{S}_{\lambda_2\mu_2}(F_k). \end{cases} \quad (12)$$

Here $\mathcal{S}_\tau(X)$ is the shrinkage operator [33], [34] on matrix X with respect to threshold τ defined as

$$(\mathcal{S}_\tau(X))_{ij} = max\{|X_{ij}| - \tau, 0\} \cdot sign(X_{ij}),$$

where $sign(\cdot)$ is the sign function.

Putting all pieces together, the whole procedure of optimizing FNNM model is summarized in Algorithm 1. Furthermore, the convergence of this algorithm is justified by analysis in [17] and [33], since model (7) is convex.

Algorithm 1: FNNM Algorithm

Input: The target matrix $\mathcal{P}_\Omega(A)$, feature matrices U, V, parameters λ_1, and λ_2.
Output: M and N.

```
initialize M_1 = 0, N_1 = 0, μ_1 = μ_2 = 0.1, k = 1;
while not converged do
    E_k ← M_k − μ_1 P_Ω(M_k + U N_k V^T − A);
    M_{k+1} ← D_{λ_1 μ_1}(E_k);
    F_k ← N_k − μ_2 U^T P_Ω(M_{k+1} + U N_k V^T − A) V;
    N_{k+1} ← S_{λ_2 μ_2}(F_k);
    k ← k + 1;
end
M ← M_k, N ← N_k;
return M and N.
```

4 Experimental Results

We demonstrate the effectiveness of FNNM for matrix completion on both synthetic experiments and real-world applications, which include movie recommendation, drug-target interaction prediction, and multi-label learning. FNNM is compared with six state-of-the-art matrix completion methods including SVT [16], FPC [17], WNNM [24], IMC [11], Maxide [28], and SIMC [29]. To ensure a fair comparison, the parameters including λ_1, λ_2 in SIMC and FNNM, λ, δ in SVT and FPC, and λ in WNNM, Maxide, and IMC, are set by grid search from $\{10^{-3}, 10^{-2}, \ldots, 10^2, 10^3\}$. For IMC, the best rank of M or N is choosen from $\{0.1, 0.3, 0.5, 0.7, 0.9\} * min(size(A))$. The FNNM algorithm is terminated when the following stopping criterion is satisfied,

$$\frac{\left\|(M_{k+1} + UN_{k+1}V^T) - (M_k + UN_kV^T)\right\|_F}{\|M_k + UN_kV^T\|_F} \leq tol, \quad (13)$$

where tol is a given tolerance set as 10^{-4}. μ_1 and μ_2 in FNNM model are both set to 0.1 in computation on all datasets. All experimental results have been obtained on a Linux server with CPU 2.60 GHz and 1 T memory.

Synthetic experiments. The real-life datasets, in general, is noisy and with low sampling rate. For example, in movie recommendation, the rating matrix is so sparse and low sampling rate since most users watch and rate only a small portion of the movie. Additionally, the rating matrix may contain some artificial noise information, which makes the rank of the matrix raise. Therefore, we attempt to simulate the matrices with noise and low sampling rate in synthetic experiments. We generate two random matrices $P \in \mathbb{R}^{150\times50}$ and $Q \in \mathbb{R}^{200\times50}$ with each component P_{ij}, Q_{ij} drawn independently from $\mathcal{N}(0, 1)$, and then create

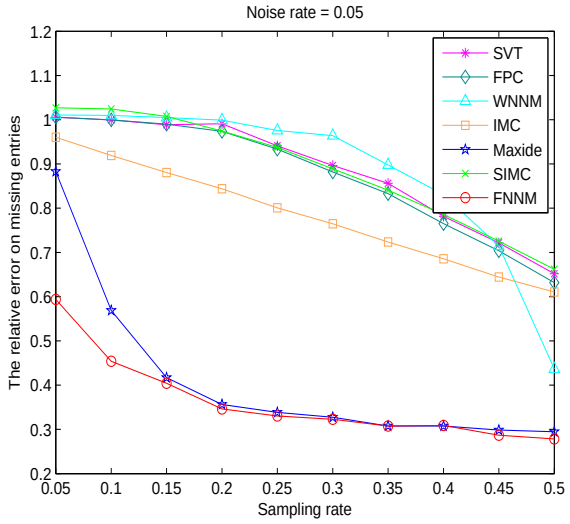
a low-rank matrix $T = PQ^T$. The noise matrix $\mathcal{E}$ is drawn elementwisely from $\mathcal{N}(0, 1)$. In order to fully illustrate the impact of noise in both transductive and inductive matrix completion, we add noise on observation and side information such that $T \leftarrow T + \mathcal{E}$, $P \leftarrow P + \mathcal{E}$, and $Q \leftarrow Q + \mathcal{E}$. The side feature matrices U and V are obtained by applying QR decomposition to noisy P and Q, respectively. The sampling rate of a matrix, s_r, is the ratio of the observed entries with respect to all components in T and the noise rate, n_r, is the proportion of the contaminated entries among all components. To evaluate the performance of all methods under different sampling rates and noise rates, we iteratively set $s_r \in [0.05, 0.5]$ and $n_r \in [0, 0.5]$ with a gradually increasing step of 0.05.

All known entries of matrix T are divided into two parts: one containing the selected observed entries treated as the training set while the other composed of the remaining elements considered as the test set. The relative error on the test set is used to measure the performance of all methods, which is defined as

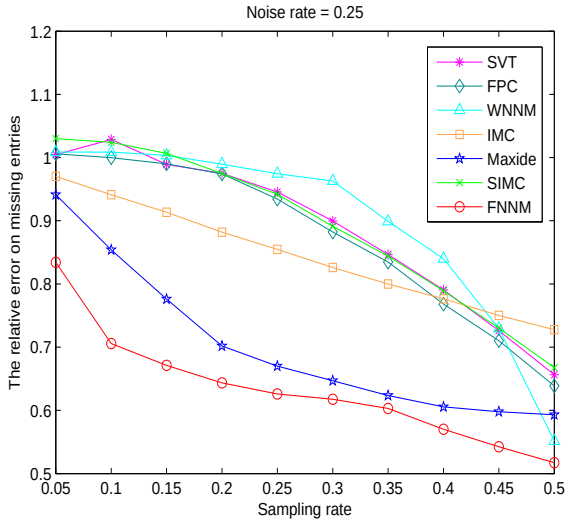
$$Relative\ \ error = \frac{\|\mathcal{P}_{\tilde{\Omega}}(T - B)\|_F}{\|\mathcal{P}_{\tilde{\Omega}}(T)\|_F}, \tag{14}$$

where B is a completed matrix from an observed matrix, and $\tilde{\Omega}$ is the set of indices of the test set. For all models used for comparison, we randomly pick 10% of the known elements from the training set to form a validation set in order to identify the optimal model parameters by grid search. After determining the optimal model parameters, we repeat each trials five times and report the average relative error as the final result.

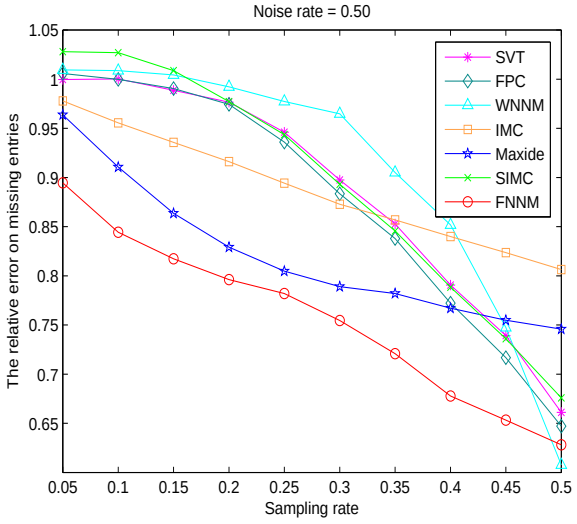
We compare the performance of various matrix completion methods under gradually increasing sampling rates when a specific noise rate is given. We choose three noise rates $\{0.05(low), 0.25(medium), 0.5(high)\}$ to conduct matrix completion, whose results are depicted in Figures 2(a)-(c), respectively. Without surprise, all methods can reduce the relative error gradually by increasing sampling rates. However, it is interesting to find that, when the sampling rate is low ($< 25\%$), the known matrix entries are too scarce to recover the whole matrix and thus the side information plays a more important role in predicting the unknown. As a result, the models (IMC, Maxide, and FNNM) taking advantage of side information outperforms those transductive models such as SVT, WNNM, and FPC. The only exception is SIMC. Although side information is used, SIMC attempts to minimize the difference between the inductive part and the transductive part, which yields similar performance as the other transductive models. Compared to IMC, a non-convex model which may lead to solutions at local minima, the convex models of Maxide and FNNM result in solutions with better accuracy. When the sampling rate is higher and more known entries become available, the transductive models are able to come up with recovered matrices in better quality and therefore their performance starts to catch up and surpass IMC and later Maxide, which use side information only. Due to its nature of balancing the inductive and transductive parts, FNNM yields better accuracy in most cases under different noise rates and sampling rates compared to the other models.



(a)



(b)



(c)

Fig. 2. The performance of all methods in matrix completion under various noise rates. (a) $n_r = 0.05$. (b) $n_r = 0.25$. (c) $n_r = 0.5$.

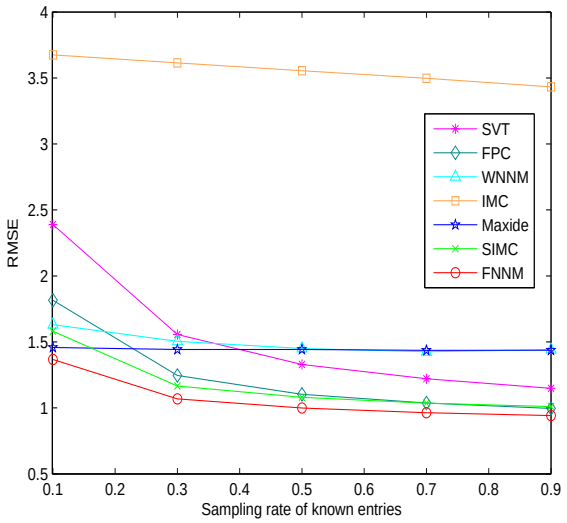


Fig. 3. The comparison of RMSE values of all methods on MovieLens.

Movie recommendation. A MovieLens dataset (100K) is downloaded from [35], which contains $100,000$ ratings (integers ranging from 1 to 5) from 943 users and $1,682$ movies. The dataset contains 23 user features (age, gender, occupation, etc.) and 20 movie features (genre, release date, etc.). We denote s_{kr} as the sampling rate on known ratings and set $s_{kr} \in [0.1, 0.9]$ with an incremental step of 0.2. Then, all known entries of rating matrix are divided into a training set with $100,000 s_{kr}$ ratings and a test set with $100,000(1 - s_{kr})$ ratings. The training set and the test set are not overlapping. Root Mean Square Error (RMSE) on the test set is used to measure the performance of different methods, which is defined as

$$RMSE = \frac{\|\mathcal{P}_{\tilde{\Omega}}(A - B)\|_F}{\sqrt{|\tilde{\Omega}|}}, \tag{15}$$

where A is the original rating matrix, B is the completed matrix, and $\tilde{\Omega}$ is the set of indices in the test set. For each method, we randomly picked 10% of the known elements out of the training set to form a validation set to determine the optimal parameters. We repeat each trial five times and report the average RMSE on the test set as the final result. The comparison of RMSE values of all methods with respect to sampling rates on MovieLens is illustrated in Figure 3. It is interesting to find that, in the movie recommendation problem, the inductive models (IMC and Maxide) using side information only are able to capture the mapping between side information of users and movies with small number of data samples. Increasing the number of data samples does not lead to significant accuracy improvement. In contrast, models involving transductive completion benefit from more data samples. After all, FNNM shows its advantage of integrating inductive and transductive completions, which consistently yields the lowest RMSE values compared to the other models in situations where the sampling rates range from low to high.

Drug-target interaction prediction. Four gold standard datasets (Nuclear Receptors, G-Protein-Coupled Receptors (GPCRs), Ion Channels, and Enzymes) of drug-target interactions are downloaded from [36]. The data statistics for drugs, target proteins, and their interactions are summarized in Table 1. Different from the movie recommendation problem, the datasets in drug-target interaction prediction contain positive samples only. The side information is obtained based on drug similarity and target similarity. Drug similarity is calculated by using SIMCOMP [37] according to the inherent chemical structures in drugs. Target similarity is computed by using a normalized Smith-Waterman score [38] with respect to the protein sequences. We apply Singular Value Decomposition (SVD) to extract the primary drug and target feature vectors from drug similarity matrix (D_s) and target similarity matrix (T_s), respectively. The numbers of drugs' primary feature vectors are determined by using the dominant energy strategy [13]. More specifically, let $U_d \Sigma_d V_d^T$ be the SVD of D_s. Then, the number of drugs' primary feature vectors is determined by

$$L_d = \arg\min_x \{ \frac{\sum_{i=1}^{x} \sigma_i}{\sum_i \sigma_i} \geq 0.9 \}, \tag{16}$$

where σ_i is the ith largest singular value of D_s. Afterwards, $\{U_1, U_2, ..., U_{L_d}\}$ are extracted as the drug features. The target feature vectors are obtained in a similar manners.

Table 2 lists the AUC (Area Under the ROC Curve) values of the seven matrix completion methods in a 10-fold cross-validation. As shown in the Table 1, the sampling rate is small ($< 10\%$) in these four datasets. More severely, some drugs or targets are new without any previously known associations, which presents a "cold-start" challenge to transductive completion methods. As a result, transductive completion models (SVT, FPC, and WNNM) have lower AUC values than those of the inductive models (IMC and Maxide). Similar to the situations in the synthetic experiments when the sampling rate is low, FNNM and Maxide yield the highest and second highest AUC values in all four datasets, respectively. However, in drug-target interaction prediction, one is usually not interested in how the model performs on the negative class; instead, precision and recall of every positive prediction are more important metrics. Therefore, we also evaluate the matrix completion models using the AUPR values (Area Under Precision-Recall curve), which are particularly effective in the case where there are significantly more negative data samples than the positive ones. The AUPR results are listed in Table 3. It is interesting to note that Maxide loses its second position to the other models, indicating that its high AUC values come along with high false negative rates. FNNM still yields the highest AUPR values among all models in all four datasets, demonstrating that it is also an effective method in bioinformatics association prediction applications.

Multi-label learning. We compare the performance of FNNM with six matrix completion models in multi-label learning. We use eleven datasets from "yahoon.com" [39] for web page classification, including 'Arts,' 'Business,' 'Computers,' 'Education,' 'Entertainment,' 'Health,' 'Recreation,' 'Reference,' 'Science,' and 'Social.' For 'Arts' dataset, the number of its instances, dimensions, and lables are $5,000$, 462, and 26, respectively. We randomly pick 10% instances as the test set and use the rest 90% as the training set.

TABLE 1
Statistics for four gold standard datasets of drug-target interaction

Statistics	Enzymes	Ion Channels	GPCRs	Nuclear Receptors
No. of drugs	445	210	223	54
No. of target proteins	664	204	95	26
No. of interactions	2926	1476	635	90
Sampling rate	0.990%	3.445%	2.997%	6.410%

TABLE 2
AUC scores of FNNM and the competing methods in 10-fold cross-validation

Datasets	SVT	FPC	WNNM	IMC	Maxide	SIMC	**FNNM**
Nuclear Receptors	0.6630 (±0.0307)	0.6670 (±0.0163)	0.6672 (±0.0193)	0.8028 (±0.0198)	<u>0.8345</u> (±0.0031)	0.8238 (±0.0073)	**0.8464** (±0.0076)
GPCRs	0.8505 (±0.0012)	0.8524 (±0.0037)	0.8516 (±0.0081)	0.9017 (±0.0032)	<u>0.9116</u> (±0.0036)	0.8087 (±0.0060)	**0.9401** (±0.0028)
Ion Channels	0.9386 (±0.0028)	0.9365 (±0.0015)	0.9376 (±0.0033)	0.9391 (±0.0024)	<u>0.9532</u> (±0.0014)	0.8790 (±0.0033)	**0.9736** (±0.0012)
Enzymes	0.8962 (±0.0017)	0.8953 (±0.0007)	0.8913 (±0.0012)	0.9354 (±0.0011)	<u>0.9446</u> (±0.0014)	0.8818 (±0.0021)	**0.9650** (±0.0008)

Best AUC result in each row is **bold** and second best result is <u>underline</u>.

TABLE 3
AUPR scores of FNNM and the competing methods in 10-fold cross-validation

Datasets	SVT	FPC	WNNM	IMC	Maxide	SIMC	**FNNM**
Nuclear Receptors	0.2737 (±0.0291)	0.2633 (±0.0195)	0.2320 (±0.0086)	<u>0.4374</u> (±0.0176)	0.3157 (±0.0153)	0.3178 (±0.0209)	**0.4445** (±0.0103)
GPCRs	0.5940 (±0.0059)	<u>0.6238</u> (±0.0063)	0.4626 (±0.0150)	0.5292 (±0.0031)	0.3773 (±0.0054)	0.2748 (±0.0107)	**0.6757** (±0.0120)
Ion Channels	<u>0.8478</u> (±0.0027)	0.8472 (±0.0006)	0.7855 (±0.0042)	0.6833 (±0.0032)	0.5325 (±0.0047)	0.3348 (±0.0146)	**0.8732** (±0.0020)
Enzymes	0.7761 (±0.0027)	<u>0.7764</u> (±0.0026)	0.7159 (±0.0045)	0.7626 (±0.0036)	0.6601 (±0.0021)	0.5165 (±0.0022)	**0.8470** (±0.0023)

Best AUPR result in each row is **bold** and second best result is <u>underline</u>.

To conduct partial label assignment in the training set, for each label, we randomly choose $\omega\%$ positive and negative training instances and keep the remaining training instances unknown. The percentage of training instances $\omega\%$ ranges from 10% to 90% with an increasing step size of 20%. We use Average Precision (AP) [28], [40] on all the data as the evaluation metric. Each computation is repeated five times and the average score is reported in Table 4. In the 55 comparison cases, FNNM achieves the best AP for 36 cases and the second best AP for 11 cases, while Maxide and SIMC get the best AP for 14 cases and 1 case, respectively. This also shows the effectiveness of FNNM on multi-label learning.

FNNM is a computational method designed to balance the global pattern and the local pattern. The global pattern depends on the quantity of known entries while the local pattern relies on the quality of side information. Generally, FNNM does not perform well in multi-label learning datasets with extremely low rate of positive samples, due to the fact that the global pattern retrieved from insufficient entries is highly noisy, which contributes little to the predictions. Considering the Business dataset as an example, the positive samples rate is only 0.49% when we set $\omega = 10\%$. This is also confirmed by comparing the transductive approaches (e.g., SVT) with the inductive methods (e.g., Maxide). One can find that Maxide significantly outperforms SVT, since side information used in the inductive methods plays a much more important role for prediction when insufficient entries are known in the association matrix. Therefore, the global pattern contributes very little compared to the local pattern in this situation. As a result, when there are extremely insufficient samples in the association matrix, the inductive methods, such as Maxide, completely relying on the side information, may yield better prediction results than FNNM. Nevertheless, when more samples are available in the association matrix and the extracted global patterns starts to contribute, FNNM, with the advantage of leveraging both global patterns and local patterns, will become more effective, as we show in the results on the other datasets with more samples available in the association matrix.

5 Computation Time Comparisons

We have compared the computational time of different methods on the datasets of MovieLens 100K, Ion Channels, and Arts. Table 5 compares the averaged execution time of various matrix completion algorithms when less than

TABLE 4
AP scores of FNNM and the competing models on incomplete multi-label learning.

Dataset	ω	SVT	FPC	WNNM	IMC	Maxide	SIMC	**FNNM**
Arts	$\omega\% = 10\%$	0.3300 (±0.0294)	0.3378 (±0.0025)	0.3370 (±0.0027)	0.4825 (±0.0032)	**0.5372** (±0.0057)	0.2888 (±0.0333)	0.4991 (±0.0111)
	$\omega\% = 30\%$	0.4723 (±0.0174)	0.4921 (±0.0031)	0.4860 (±0.0035)	0.6066 (±0.0046)	0.6341 (±0.0028)	0.5471 (±0.0205)	**0.6990** (±0.0045)
	$\omega\% = 50\%$	0.6187 (±0.0249)	0.6353 (±0.0021)	0.6236 (±0.0030)	0.6574 (±0.0021)	0.6797 (±0.0015)	0.6663 (±0.0075)	**0.7720** (±0.0013)
	$\omega\% = 70\%$	0.7473 (±0.0110)	0.7645 (±0.0033)	0.7567 (±0.0021)	0.6894 (±0.0016)	0.7082 (±0.0020)	0.7131 (±0.0120)	**0.8531** (±0.0032)
	$\omega\% = 90\%$	0.8709 (±0.0091)	0.8742 (±0.0028)	0.8720 (±0.0013)	0.7123 (±0.0018)	0.7236 (±0.0009)	0.7503 (±0.0082)	**0.9267** (±0.0010)
Business	$\omega\% = 10\%$	0.4986 (±0.0148)	0.8444 (±0.0064)	0.8329 (±0.0033)	0.7429 (±0.0035)	0.8639 (±0.0025)	**0.8736** (±0.0030)	0.8446 (±0.0053)
	$\omega\% = 30\%$	0.5707 (±0.0859)	**0.8898** (±0.0021)	0.8533 (±0.0026)	0.8299 (±0.0030)	0.8744 (±0.0006)	0.9043 (±0.0009)	0.8896 (±0.0026)
	$\omega\% = 50\%$	0.6892 (±0.0329)	**0.9235** (±0.0014)	0.8881 (±0.0044)	0.8501 (±0.0013)	0.8838 (±0.0020)	0.9153 (±0.0264)	0.9223 (±0.0018)
	$\omega\% = 70\%$	0.7740 (±0.0463)	**0.9512** (±0.0020)	0.9246 (±0.0038)	0.8559 (±0.0010)	0.8913 (±0.0009)	0.9447 (±0.0052)	0.9480 (±0.0017)
	$\omega\% = 90\%$	0.9090 (±0.0043)	**0.9712** (±0.0021)	0.9635 (±0.0016)	0.8587 (±0.0005)	0.8975 (±0.0005)	0.9455 (±0.0028)	0.9669 (±0.0014)
Computers	$\omega\% = 10\%$	0.3294 (±0.0552)	0.5421 (±0.0009)	0.5387 (±0.0027)	0.5412 (±0.0061)	**0.6341** (±0.0079)	0.6154 (±0.0161)	0.5431 (±0.0121)
	$\omega\% = 30\%$	0.4502 (±0.0312)	0.6510 (±0.0014)	0.6348 (±0.0024)	0.6745 (±0.0032)	0.7180 (±0.0033)	0.7150 (±0.0048)	**0.7190** (±0.0032)
	$\omega\% = 50\%$	0.5968 (±0.0193)	0.7481 (±0.0017)	0.7308 (±0.0023)	0.7370 (±0.0054)	0.7674 (±0.0029)	0.7813 (±0.0015)	**0.8100** (±0.0014)
	$\omega\% = 70\%$	0.7040 (±0.0327)	0.8331 (±0.0042)	0.8232 (±0.0009)	0.7745 (±0.0026)	0.7940 (±0.0012)	0.8383 (±0.0076)	**0.8828** (±0.0026)
	$\omega\% = 90\%$	0.8671 (±0.0054)	0.9063 (±0.0020)	0.9078 (±0.0019)	0.8012 (±0.0010)	0.8105 (±0.0008)	0.8959 (±0.0036)	**0.9426** (±0.0015)
Education	$\omega\% = 10\%$	0.3291 (±0.0336)	0.3637 (±0.0022)	0.3637 (±0.0014)	0.4923 (±0.0073)	**0.5536** (±0.0057)	0.5223 (±0.0090)	0.5053 (±0.0034)
	$\omega\% = 30\%$	0.4819 (±0.0360)	0.5105 (±0.0028)	0.4973 (±0.0025)	0.6094 (±0.0053)	0.6465 (±0.0034)	0.6320 (±0.0098)	**0.6760** (±0.0047)
	$\omega\% = 50\%$	0.6075 (±0.0148)	0.6424 (±0.0023)	0.6271 (±0.0026)	0.6697 (±0.0020)	0.6964 (±0.0018)	0.7156 (±0.0037)	**0.7718** (±0.0022)
	$\omega\% = 70\%$	0.7529 (±0.0137)	0.7659 (±0.0008)	0.7549 (±0.0017)	0.7070 (±0.0016)	0.7244 (±0.0016)	0.7879 (±0.0113)	**0.8565** (±0.0029)
	$\omega\% = 90\%$	0.8747 (±0.0065)	0.8698 (±0.0014)	0.8731 (±0.0010)	0.7312 (±0.0017)	0.7431 (±0.0013)	0.8466 (±0.0048)	**0.9281** (±0.0022)
Entertainment	$\omega\% = 10\%$	0.3321 (±0.0221)	0.3325 (±0.0023)	0.3304 (±0.0022)	0.5414 (±0.0101)	**0.6210** (±0.0066)	0.5218 (±0.0108)	0.5430 (±0.0067)
	$\omega\% = 30\%$	0.5026 (±0.0315)	0.4777 (±0.0026)	0.4704 (±0.0017)	0.6625 (±0.0040)	0.7002 (±0.0021)	0.6979 (±0.0067)	**0.7086** (±0.0026)
	$\omega\% = 50\%$	0.6178 (±0.0203)	0.6164 (±0.0032)	0.6038 (±0.0020)	0.7233 (±0.0033)	0.7537 (±0.0017)	0.7907 (±0.0065)	**0.8015** (±0.0021)
	$\omega\% = 70\%$	0.7447 (±0.0203)	0.7462 (±0.0022)	0.7392 (±0.0042)	0.7626 (±0.0004)	0.7801 (±0.0014)	0.8541 (±0.0046)	**0.8726** (±0.0028)
	$\omega\% = 90\%$	0.8651 (±0.0065)	0.8690 (±0.0024)	0.8654 (±0.0012)	0.7896 (±0.0022)	0.7976 (±0.0013)	0.9040 (±0.0036)	**0.9383** (±0.0019)
Health	$\omega\% = 10\%$	0.4210 (±0.0351)	0.4495 (±0.0035)	0.4427 (±0.0051)	0.5889 (±0.0028)	**0.7074** (±0.0032)	0.6698 (±0.0069)	0.6099 (±0.0091)
	$\omega\% = 30\%$	0.5259 (±0.0452)	0.5952 (±0.0025)	0.5779 (±0.0018)	0.7260 (±0.0049)	**0.7773** (±0.0030)	0.7654 (±0.0054)	0.7714 (±0.0031)
	$\omega\% = 50\%$	0.6349 (±0.0401)	0.7179 (±0.0020)	0.6983 (±0.0048)	0.7879 (±0.0028)	0.8203 (±0.0011)	0.7736 (±0.1211)	**0.8531** (±0.0011)
	$\omega\% = 70\%$	0.7603 (±0.0220)	0.8190 (±0.0016)	0.8019 (±0.0032)	0.8238 (±0.0014)	0.8419 (±0.0008)	0.8706 (±0.0075)	**0.9072** (±0.0018)
	$\omega\% = 90\%$	0.8666 (±0.0077)	0.8996 (±0.0013)	0.8814 (±0.0052)	0.8445 (±0.0011)	0.8550 (±0.0008)	0.9219 (±0.0030)	**0.9570** (±0.0022)

Recreation	$\omega\% = 10\%$	0.3259 ($\pm$0.0106)	0.3646 ($\pm$0.0027)	0.3670 ($\pm$0.0017)	0.4835 ($\pm$0.0055)	**0.5582** ($\pm$0.0035)	0.4211 ($\pm$0.0274)	0.4930 ($\pm$0.0086)
	$\omega\% = 30\%$	0.4246 ($\pm$0.0136)	0.5048 ($\pm$0.0018)	0.5022 ($\pm$0.0015)	0.6209 ($\pm$0.0050)	0.6582 ($\pm$0.0026)	0.5997 ($\pm$0.0085)	**0.6679** ($\pm$0.0015)
	$\omega\% = 50\%$	0.5775 ($\pm$0.0081)	0.6370 ($\pm$0.0024)	0.6282 ($\pm$0.0008)	0.6873 ($\pm$0.0034)	0.7142 ($\pm$0.0033)	0.7094 ($\pm$0.0069)	**0.7713** ($\pm$0.0025)
	$\omega\% = 70\%$	0.7250 ($\pm$0.0119)	0.7604 ($\pm$0.0008)	0.7574 ($\pm$0.0022)	0.7268 ($\pm$0.0018)	0.7488 ($\pm$0.0022)	0.7822 ($\pm$0.0053)	**0.8549** ($\pm$0.0012)
	$\omega\% = 90\%$	0.8430 ($\pm$0.0031)	0.8753 ($\pm$0.0028)	0.8741 ($\pm$0.0010)	0.7541 ($\pm$0.0011)	0.7675 ($\pm$0.0008)	0.8347 ($\pm$0.0042)	**0.9282** ($\pm$0.0014)
Reference	$\omega\% = 10\%$	0.3280 ($\pm$0.0737)	0.3296 ($\pm$0.0008)	0.3296 ($\pm$0.0002)	0.5313 ($\pm$0.0025)	**0.6522** ($\pm$0.0076)	0.5862 ($\pm$0.0053)	0.5390 ($\pm$0.0071)
	$\omega\% = 30\%$	0.4835 ($\pm$0.0389)	0.4623 ($\pm$0.0016)	0.4606 ($\pm$0.0012)	0.6749 ($\pm$0.0055)	**0.7271** ($\pm$0.0019)	0.6911 ($\pm$0.0033)	0.7144 ($\pm$0.0026)
	$\omega\% = 50\%$	0.6132 ($\pm$0.0208)	0.5952 ($\pm$0.0015)	0.5933 ($\pm$0.0015)	0.7476 ($\pm$0.0026)	0.7833 ($\pm$0.0026)	0.7757 ($\pm$0.0024)	**0.8079** ($\pm$0.0022)
	$\omega\% = 70\%$	0.7448 ($\pm$0.0050)	0.7299 ($\pm$0.0019)	0.7279 ($\pm$0.0017)	0.7896 ($\pm$0.0023)	0.8116 ($\pm$0.0020)	0.8506 ($\pm$0.0015)	**0.8814** ($\pm$0.0028)
	$\omega\% = 90\%$	0.8625 ($\pm$0.0134)	0.8623 ($\pm$0.0009)	0.8599 ($\pm$0.0003)	0.8205 ($\pm$0.0017)	0.8334 ($\pm$0.0014)	0.9205 ($\pm$0.0032)	**0.9427** ($\pm$0.0010)
Science	$\omega\% = 10\%$	0.2729 ($\pm$0.0185)	0.2519 ($\pm$0.0006)	0.2519 ($\pm$0.0015)	0.4400 ($\pm$0.0050)	**0.5343** ($\pm$0.0074)	0.4028 ($\pm$0.0198)	0.4377 ($\pm$0.0063)
	$\omega\% = 30\%$	0.4174 ($\pm$0.0366)	0.4218 ($\pm$0.0023)	0.4188 ($\pm$0.0011)	0.5901 ($\pm$0.0037)	**0.6376** ($\pm$0.0031)	0.5837 ($\pm$0.0148)	0.6368 ($\pm$0.0038)
	$\omega\% = 50\%$	0.5857 ($\pm$0.0194)	0.5780 ($\pm$0.0019)	0.5732 ($\pm$0.0018)	0.6672 ($\pm$0.0041)	0.7016 ($\pm$0.0013)	0.6900 ($\pm$0.0066)	**0.7501** ($\pm$0.0024)
	$\omega\% = 70\%$	0.7286 ($\pm$0.0133)	0.7231 ($\pm$0.0030)	0.7203 ($\pm$0.0025)	0.7159 ($\pm$0.0030)	0.7413 ($\pm$0.0023)	0.7895 ($\pm$0.0067)	**0.8434** ($\pm$0.0023)
	$\omega\% = 90\%$	0.8606 ($\pm$0.0050)	0.8555 ($\pm$0.0007)	0.8541 ($\pm$0.0021)	0.7523 ($\pm$0.0006)	0.7656 ($\pm$0.0016)	0.8583 ($\pm$0.0024)	**0.9230** ($\pm$0.0021)
Social	$\omega\% = 10\%$	0.3682 ($\pm$0.0471)	0.2123 ($\pm$0.0015)	0.2124 ($\pm$0.0011)	0.5548 ($\pm$0.0096)	**0.7299** ($\pm$0.0030)	0.6595 ($\pm$0.0140)	0.5904 ($\pm$0.0066)
	$\omega\% = 30\%$	0.5035 ($\pm$0.0341)	0.3791 ($\pm$0.0021)	0.3763 ($\pm$0.0015)	0.7076 ($\pm$0.0047)	**0.7856** ($\pm$0.0028)	0.7423 ($\pm$0.0062)	0.7581 ($\pm$0.0034)
	$\omega\% = 50\%$	0.6377 ($\pm$0.0197)	0.5420 ($\pm$0.0012)	0.5338 ($\pm$0.0022)	0.7906 ($\pm$0.0031)	0.8361 ($\pm$0.0032)	0.8088 ($\pm$0.0034)	**0.8435** ($\pm$0.0036)
	$\omega\% = 70\%$	0.7560 ($\pm$0.0186)	0.6987 ($\pm$0.0012)	0.6911 ($\pm$0.0014)	0.8400 ($\pm$0.0007)	0.8680 ($\pm$0.0008)	0.8578 ($\pm$0.0125)	**0.9034** ($\pm$0.0012)
	$\omega\% = 90\%$	0.8687 ($\pm$0.0127)	0.8433 ($\pm$0.0012)	0.8405 ($\pm$0.0007)	0.8748 ($\pm$0.0016)	0.8880 ($\pm$0.0009)	0.9022 ($\pm$0.0054)	**0.9551** ($\pm$0.0015)
Society	$\omega\% = 10\%$	0.3203 ($\pm$0.0716)	0.2409 ($\pm$0.0024)	0.2404 ($\pm$0.0016)	0.5034 ($\pm$0.0026)	**0.5661** ($\pm$0.0037)	0.5598 ($\pm$0.0043)	0.5115 ($\pm$0.0042)
	$\omega\% = 30\%$	0.4498 ($\pm$0.0507)	0.4155 ($\pm$0.0030)	0.4003 ($\pm$0.0015)	0.6286 ($\pm$0.0039)	0.6567 ($\pm$0.0025)	0.6607 ($\pm$0.0065)	**0.6778** ($\pm$0.0063)
	$\omega\% = 50\%$	0.6007 ($\pm$0.0402)	0.5742 ($\pm$0.0024)	0.5521 ($\pm$0.0042)	0.6863 ($\pm$0.0027)	0.7120 ($\pm$0.0023)	0.7260 ($\pm$0.0066)	**0.7756** ($\pm$0.0023)
	$\omega\% = 70\%$	0.7225 ($\pm$0.0099)	0.7162 ($\pm$0.0033)	0.7041 ($\pm$0.0015)	0.7283 ($\pm$0.0032)	0.7432 ($\pm$0.0018)	0.7663 ($\pm$0.0088)	**0.8552** ($\pm$0.0036)
	$\omega\% = 90\%$	0.8693 ($\pm$0.0094)	0.8423 ($\pm$0.0019)	0.8467 ($\pm$0.0020)	0.7555 ($\pm$0.0013)	0.7633 ($\pm$0.0014)	0.7984 ($\pm$0.0122)	**0.9299** ($\pm$0.0035)

Best AP result in each row is **bold** and second best result is underline.

1% relative errors are reached. One can find that FNNM has better computational efficiency than SVT, Maxide, and SIMC on small matrices such as Ion Channels and Arts. On a bigger dataset such as MovieLens 100K, FNNM is more computationally efficient compared to SIMC and FPC. In fact, the computational time of a matrix completion algorithm depends on the nature of the application matrices, the model hyperparameters, the computer architectures, as well as the anticipated precision.

6 Conclusions

In this study, we propose an FNNM model for matrix completion, assuming that the to-be-complete matrix can be well approximated as the sum of a low-rank matrix and a sparse matrix associating with the side information matrices. The low-rank matrix represents the global patterns while the sparse matrix describes the local patterns mapping the side information. FNNM model is designed to balance the transductive matrix completion for global patterns and inductive matrix completion for local patterns, which can be efficiently solved by an alternative minimization scheme using fixed-point iterations. Compared with several state-

TABLE 5
The averaged execution time of various matrix completion algorithms to reach 1% relative error

Time(s)	SVT	FPC	WNMM	IMC	Maxide	SIMC	FNNM
MovieLens 100K	1.86	24.23	15.59	0.08	1.05	26.07	18.07
Ion Channels	0.61	0.66	0.44	0.13	3.73	15.48	0.22
Arts	1.43	0.02	0.16	0.19	1.12	26.81	0.74

of-the-art matrix completion models, FNNM consistently demonstrates better performance, measured by different evaluation metrics, in synthetic experiments as well as real-world applications. However, there are two limitations in FNNM. First, it is not convenient to determine its parameters, where one is not able to choose the optimal parameters in advance, nor can one determine them adaptively. Second, FNNM relies on SVD algorithm to carry out its optimization. Although there are fast SVD approximation algorithms [41] available, handling extremely large-scale matrices is still time-consuming.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 61972423, 61732009, 61622213, 61772552 and 61420106009) and the Graduate Research Innovation Project of Hunan (Grant No. CX20190125).

REFERENCES

[1] P. Chen and D. Suter, "Recovering the missing components in a large noisy low-rank matrix: Application to SFM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1051–1063, 2004.

[2] D. Zhang, Y. Hu, J. Ye, X. Li, and X. He, "Matrix completion by truncated nuclear norm regularization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2192–2199, 2012.

[3] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proceedings of the 22nd International Conference on Machine Learning*, pp. 713–719, 2005.

[4] R. Salakhutdinov and N. Srebro, "Collaborative filtering in a non-uniform world: Learning with the weighted trace norm," in *Advances in Neural Information Processing Systems*, pp. 2056–2064, 2010.

[5] A. Ramlatchan, M. Yang, Q. Liu, M. Li, J. Wang, and Y. Li, "A survey of matrix completion methods for recommendation systems," *Big Data Mining and Analytics*, vol. 1, no. 4, pp. 308–323, 2018.

[6] V. Sindhwani, S. S. Bucak, J. Hu, and A. Mojsilovic, "One-class matrix completion with low-density factorizations," in *IEEE International Conference on Data Mining*, pp. 1055–1060, 2010.

[7] M. Wang, W. Fu, S. Hao, D. Tao, and X. Wu, "Scalable semi-supervised learning by efficient anchor graph regularization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1864–1877, 2016.

[8] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Machine Learning*, vol. 73, no. 3, pp. 243–272, 2008.

[9] G. Obozinski, B. Taskar, and M. I. Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems," *Statistics and Computing*, vol. 20, no. 2, pp. 231–252, 2010.

[10] M. Wang, W. Fu, S. Hao, H. Liu, and X. Wu, "Learning on big graph: Label inference and regularization with anchor hierarchy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1101–1114, 2017.

[11] N. Natarajan and I. S. Dhillon, "Inductive matrix completion for predicting gene-disease associations," *Bioinformatics*, vol. 30, no. 12, pp. 60–68, 2014.

[12] H. Luo, M. Li, S. Wang, Q. Liu, Y. Li, and J. Wang, "Computational drug repositioning using low-rank matrix approximation and randomized algorithms," *Bioinformatics*, vol. 34, no. 11, pp. 1904–1912, 2018.

[13] C. Lu, M. Yang, F. Luo, F. X. Wu, M. Li, Y. Pan, Y. Li, and J. Wang, "Prediction of lncRNA-disease associations based on inductive matrix completion," *Bioinformatics*, vol. 34, no. 19, pp. 3357–3364, 2018.

[14] Z. Cai, T. Zhang, and X. Wan, "A computational framework for influenza antigenic cartography," *Plos Computational Biology*, vol. 6, no. 10, p. e1000949, 2010.

[15] Z. Cai, T. Zhang, and X. Wan, "Concepts and applications for influenza antigenic cartography," *Influenza and Other Respiratory Viruses*, vol. 5, no. S1, pp. 202–229, 2011.

[16] J. F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2008.

[17] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, no. 1-2, pp. 321–353, 2011.

[18] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 3, pp. 615–640, 2010.

[19] G. Liu, Q. Liu, X. Yuan, and M. Wang, "Matrix completion with deterministic sampling: Theories and methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[20] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[21] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, p. 717, 2009.

[22] E. J. Candès and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[23] B. Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.

[24] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2862–2869, 2014.

[25] F. Nie, H. Wang, H. Huang, and C. Ding, "Joint Schatten p-norm and l_p-norm robust matrix comletion for missing value recovery," *Knowledge and Information Systems*, vol. 42, no. 3, pp. 525–544, 2015.

[26] P. Jain and I. S. Dhillon, "Provable inductive matrix completion," *arXiv preprint arXiv:1306.0626*, 2013.

[27] A. K. Menon, K. P. Chitrapura, S. Garg, D. Agarwal, and N. Kota, "Response prediction using collaborative filtering with hierarchies and side-information," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 141–149, 2011.

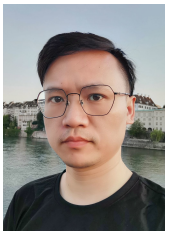
[28] M. Xu, R. Jin, and Z.-H. Zhou, "Speedup matrix completion with side information: Application to multi-label learning," in *Advances in neural information processing systems*, pp. 2301–2309, 2013.

[29] J. Lu, G. Liang, J. Sun, and J. Bi, "A sparse interactive model for matrix completion with side information," in *Advances in Neural Information Processing Systems*, pp. 4071–4079, 2016.

[30] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," in *Technical report, University of Washington, Seattle*, 2008.

[31] J. Yang and X. Yuan, "Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization," *Mathematics of Computation*, vol. 82, no. 281, pp. 301–329, 2012.

[32] E. T. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for l_1-minimization: Methodology and convergence," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1107–1130, 2008.
[33] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34–81, 2009.
[34] J. Yang, W. Yin, Y. Zhang, and Y. Wang, "A fast algorithm for edge-preserving variational multichannel image restoration," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 569–592, 2009.
[35] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, p. 19, 2016.
[36] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, "Prediction of drug-target interaction networks from the integration of chemical and genomic spaces," *Bioinformatics*, vol. 24, no. 13, pp. i232–i240, 2008.
[37] H. Masahiro, O. Yasushi, G. Susumu, and K. Minoru, "Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways," *Journal of the American Chemical Society*, vol. 125, no. 39, pp. 11853–65, 2003.
[38] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981.
[39] N. Ueda and K. Saito, "Parametric mixture models for multi-labeled text," in *Advances in Neural Information Processing Systems*, pp. 737–744, 2003.
[40] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819–1837, 2014.
[41] W. Yu, Y. Gu, J. Li, S. Liu, and Y. Li, "Single-pass PCA of large high-dimensional data," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, 2017.



Mengyun Yang received the MS degree in computational mathematics from Hunan Normal University in 2012. He is a lecturer in Shaoyang University and working toward the PhD degree in the School of Computer Science and Engineering, Central South University, Hunan, China. His current research interests include machine learning and bioinformatics.

Yaohang Li received the BS degree from South China University of Technology in 1997, and the MS and PhD degrees in computer science from Florida State University, Tallahassee, FL, USA, in 2000 and 2003, respectively. He is an associate professor in computer science at Old Dominion University, Norfolk, VA, USA. His research interests are in protein structure modeling, computational biology, bioinformatics, Monte Carlo methods, big data algorithms, and parallel and distributive computing.

Jianxin Wang received the BEng and MEng degrees in computer engineering from Central South University, China, in 1992 and 1996, respectively, and the PhD degree in computer science from Central South University, China, in 2001. He is the dean and a professor in the School of Computer Science and Engineering, Central South University, Changsha, Hunan, China. He is a senior member of the IEEE. His current research interests include algorithm analysis and optimization, parameraized algorithm, bioinformatics, and computer network.