Understanding the Sparsity: Augmented Matrix Factorization with Sampled Constraints on Unobservables

Yongfeng Zhang[†], Min Zhang[†], Yi Zhang[‡], Yiqun Liu[†], Shaoping Ma[†]
[†]State Key Laboratory of Intelligent Technology and Systems
[†]Department of Computer Science & Technology, Tsinghua University, Beijing, 100084, China
[‡]School of Engineering, University of California, Santa Cruz, CA 95060, USA
zhangyf07@gmail.com, {z-m,yiqunliu,msp}@tsinghua.edu.cn, yiz@soe.ucsc.edu

ABSTRACT

An important problem of matrix completion/approximation based on Matrix Factorization (MF) algorithms is the existence of *multiple global optima*; this problem is especially serious when the matrix is *sparse*, which is common in real-world applications such as personalized recommender systems. In this work, we clarify *data sparsity* by bounding the solution space of MF algorithms. We present the conditions that an MF algorithm should satisfy for reliable completion of the unobservables, and we further propose to augment current MF algorithms with extra constraints constructed by compressive sampling on the unobserved values, which is well-motivated by the theoretical analysis. Model learning and optimal solution searching is conducted in a properly reduced solution space to achieve more accurate and efficient rating prediction performances. We implemented the proposed algorithms in the Map-Reduce framework, and comprehensive experimental results on Yelp and Dianping datasets verified the effectiveness and efficiency of the augmented matrix factorization algorithms.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Filtering; F.2.1 [**Numerical Algorithms and Problems**]: Computation on Matrices; G1.6 [**Mathematics of Computing**]: Optimization

Keywords

Matrix Factorization; Collaborative Filtering; Recommender Systems; Compressed Sensing; Optimization

1. INTRODUCTION

Matrix Factorization (MF) techniques have achieved significant success in many real-world applications, such as Collaborative Filtering (CF)-based recommender systems, where MF is conducted on partially observed user-item rating matrices, and the results are thus used to predict the unobserved ratings (i.e., the missing entries). A number of commonly known MF algorithms have been proposed and extensively investigated, for example, the Singular Value Decomposition (SVD), Nonnegative Matrix Factorization (NMF), Probabilistic Matrix Factorization (PMF), etc.

The important advantages of fast iterations and flexible modeling and being amenable to parallelization make MF widely used in real-world systems. However, despite such empirical success, MF approaches have mostly been used as a heuristic with little solid theoretical analysis other than the guarantees of convergence to the local minima [10]. In fact, the performance of most MF algorithms relies heavily on the sparsity and the underlying structure of the matrices.

The existence of multiple global/local optima leads to serious problems when MF is used to predict the unobservables. Consider conducting regularized NMF on a simple Block Diagonal Form (BDF) [28, 29, 27] structured matrix $X = \left[\begin{smallmatrix}X_{11} & \\ & X_{22}\end{smallmatrix}\right]$, where the observed values are restricted in the diagonal blocks X_{11} and X_{22}, and the ratings in off-diagonals X_{12} and X_{21} are all unobserved. Suppose the NMF algorithm gives the global optimal solution that X is factorized as $X \approx UV' = \left[\begin{smallmatrix}U_1\\U_2\end{smallmatrix}\right]\left[\begin{smallmatrix}V_1' V_2'\end{smallmatrix}\right]$, where the rows of U and V are split in the same pattern according to X; then we have $X_{11} \approx U_1V_1'$ and $X_{22} \approx U_2V_2'$. Now we rearrange the columns in U_1 and V_1 in the same order, which gives $\hat{U}_1$ and $\hat{V}_1$. One can see that we have $\hat{U}_1\hat{V}_1' = U_1V_1'$ and that the newly constructed factorization $X \approx \left[\begin{smallmatrix}\hat{U}_1\\U_2\end{smallmatrix}\right]\left[\begin{smallmatrix}\hat{V}_1' V_2'\end{smallmatrix}\right]$ is also a global optimal solution, as it gives the same predicted ratings on the observed values and the same penalty on the regularization terms. However, it could give completely different predictions on the off-diagonal areas, as $\hat{U}_1V_2'$ and $U_2\hat{V}_1'$ would not be equal to U_1V_2' and U_2V_1', after the columns of U_1 and V_1 have been rearranged.

This example demonstrates the *multiple global optima* problem, where an MF algorithm "fails" on BDF matrices, as it gives completely different predictions on the unobserved values, while achieving the same minima on the objective loss function. This contradicts the reason for using MF to predict unobservables because we assume the preferences of the users to be "predictable" from their historical choices. As we will show later, this problem arises not only on BDF matrices but also on many matrices where the observed values are sparse or improperly distributed; thus, the performance of the algorithm cannot be properly constrained.

This problem is rarely noticed or investigated in the research community, most likely because most of the public benchmark datasets, such as MovieLens, Netflix and Yahoo! Music, have been preprocessed by removing those users who have rated less than a specific number of items. Such preprocessing makes a dataset biased from the original rating

CIKM'14, November 3–7, 2014, Shanghai, China.

ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2661976.

distribution, and thus the insufficiency of the ratings does not pose as severe a problem as it could be.

Recently, the release of the more "practical" Yelp rating dataset[1] exposed the problem directly. By permuting the rows and columns of the rating matrix, it can be rearranged into a BDF structure with 53 diagonal blocks, with a dominating block and 52 scattered blocks, which is shown in Fig.1(a). The corresponding bipartite graph of the matrix is highly disconnected, and these scattered blocks correspond to the 52 connected components, as shown in Fig.1(b). According to the analysis above, the presence of one single scattered block increases the global optimal solution space by $O(r!)$, where r is the number of latent factors used in an MF algorithm, which is typically assigned between 50 to 100. The algorithm could converge to any of the global optimal solutions, although they provide very different predictions on the unobserved values.

The essential reason that MF algorithms fail in such cases is that they only make constraints on the observed values in a matrix, without any constraints on the predictions of the unobserved values. In this work, we indicate with theoretical analysis that, for an MF algorithm to avoid the multiple global optima problem, and thus to recover the unobserved values properly, the following two basic conditions should be satisfied:

1. The number of constraints should be up to the order of $O(r(m+n)\log(mn))$, where r is the number of latent factors, and m and n are the number of rows and columns of a matrix.
2. The distribution of the constraints should be *nearly isometric*, which means that they should obey certain large deviation inequalities.

A single observed value can be viewed as a constraint in MF algorithms; however, the number of observed values could be far from the above requirement, and they would not necessarily be nearly isometrically distributed. In this work, we treat the MF as a subspace fitting problem and analyze the difference between the solution space and the ground truth. We propose to augment MF algorithms with extra constraints constructed from the unobserved values, which are selected according to some specific distributions. In this way, our MF model satisfies the above two conditions, and the algorithm can find a proper solution in a reduced solution space. Experimental results verify the effect of our method in improving the prediction accuracy, stability, convergence rate and computational efficiency.

The paper is structured as follows: In section 2, we introduce some of the related work; In sections 3 and 4 we give some preliminaries and conduct theoretical analysis of the solution space, which form the basis of this work, and, afterwards, we present our method and algorithms; the experimental results are shown in section 5; finally, the work is discussed in section 6 and concluded in section 7.

2. RELATED WORK

Latent factor models based on Matrix Factorization (MF) techniques have long been an important research direction in Collaborative Filtering (CF)-based recommendations [13, 26]. Recently, the MF approaches have gained great popularity, as they usually outperform traditional methods, and have achieved state-of-the-art performance [24]. A variety of MF algorithms have been investigated in different CF settings, for example, Principle Component Analysis (PCA) [1], Singular Value Decomposition (SVD) [13], Non-negative Matrix Factorization (NMF) [14], Max-Margin Matrix Factorization (MMMF) [23, 17], and Probabilistic Matrix Factorization (PMF) [19, 18].

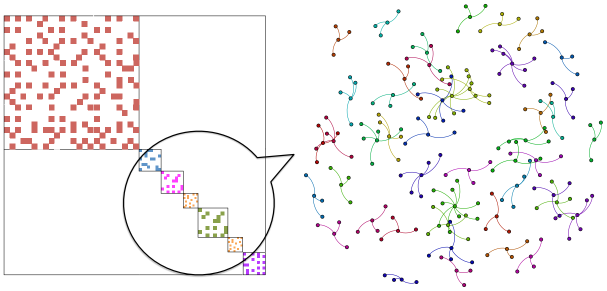

(a) Yelp dataset Matrix (b) Yelp dataset Graph

Figure 1: Structures of Yelp dataset. In the left is the exampled structure of the rating matrix, and in the right is the real structure of the scattered blocks.

However, despite such empirical success, MF approaches have mostly been used as heuristics and have little solid theoretical analysis other than the guarantees of convergence to local minima. The most recent work concerning the theoretical properties of MF algorithms is given in [10, 25], which investigate the optimization algorithms for MF and their stability with adversarial noise in terms of prediction accuracy. However, they do not touch upon the topic of how to overcome the multiple global optimal problem.

This problem is closely related to the research of matrix Compressed Sensing (CS) [8, 4], which can be viewed as a generalized form of matrix completion or matrix factorization in that a constraint is not restricted to a single observed value but instead the linear equations of the observations. According to the mathematical relationships, the CS problem is formulated as a rank minimization problem in [3] and further formulated as a convex optimization problem based on nuclear norm minimization [6, 5]. Later, [21] investigated the uniqueness of low-rank matrix completion problems with the basic tools of rigidity theory. However, the success of CS relies on the assumption that the constraints are sufficient and isometrically distributed, which can hardly be satisfied in the real-world scenarios of CF.

In the effort to tackle this problem, recent work has focused on the idea of reformulating current MF algorithms to fit the real distributions of data. [20] attempted to conduct CF on non-uniformly sampled matrices using a properly weighted version of nuclear-norm regularizers, and [15] proposed a graph theoretic approach for matrix completion under more realistic power-law distributed samples. [12] explored the relationships between matrix factorization and combinatorial algebraic theory. To speed up the process of rank minimization, [16] proposed the Singular Value Projection (SVP) algorithm for matrix completion with affine constraints. However, these approaches make tight assumptions on the distributions of data, which restricts their application in practical systems and scenarios. Instead of the traditional approach of reformulating the algorithms, we attempt to resample the data to alleviate the problem of multiple global optima, which brings about the advantages of both higher prediction accuracy and the ability to conveniently integrate the approach into many MF algorithms.

[1] http://www.kaggle.com/c/yelp-recsys-2013/

3. PRELIMINARIES

Definition 1. *Matrix Factorization (MF) algorithms can be generally defined by the constrained optimization problem:*

$$\begin{aligned} &\min_{X=UV'} \mathcal{R}(U,V) \\ &s.t.\ \mathcal{A}(X)=b \end{aligned} \tag{1}$$

where $X \in \mathbb{R}^{m\times n}$ is the approximation matrix, $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$ are the decision variables, and $\mathcal{R}$ is the regularization term. $\mathcal{A}: \mathbb{R}^{m\times n} \to \mathbb{R}^p$ is a linear map defining a set of p linear constraints on X, where each of the constraints is represented as $b_i = \langle X, A_i\rangle = \mathrm{tr}(A_i'X)$, $1 \le i \le p$, and $b \in \mathbb{R}^p$is the corresponding vector of the b_i's.

This definition of MF is equal to the frequently used regularized optimization form in terms of Lagrange multipliers. Let Λ be the Lagrange multiplier for the linear constraint $\mathcal{A}(X)-b=0$, then Eq.(1) could be reformulated as:

$$\min_{X=UV'} \mathcal{R}(U,V)+\Lambda\|\mathcal{A}(X)-b\|_2^2 = \min_{X=UV'} \|\mathcal{A}(X)-b\|_2^2+\lambda\mathcal{R}(U,V) \tag{2}$$

where $\lambda = 1/\Lambda$ is the regularization coefficient. Eq.(2) is a special case of the unified view of MF given in [22]:

$$\min_{X=f(UV')} \mathcal{D}_W(X,\tilde{X}) + \mathcal{R}(U,V) \tag{3}$$

where $\mathcal{D}_W(X,\tilde{X})$ is the *loss* between the approximation X and the original matrix $\tilde{X}$. However, Eq.(1) and Eq.(2) are general enough to represent the objective function of most MF algorithms. For example, SVD takes the objective function $\|W \odot (X-\tilde{X})\|_F^2 + \lambda(\|U\|_F^2+\|V\|_F^2)$, where $\|\cdot\|_F$ is the Frobenius norm, and W is an indication matrix such that $W_{ij}=1$ if $\tilde{X}_{ij}$ is observed, and 0 otherwise.

Proposition 1. *Let $X \in \mathbb{R}^{m\times n}$ and* $\mathrm{rank}(X) \le r$*, then there exist matrices $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$, such that $X = UV'$.*

Proof. This can be easily proven by the singular value decomposition of X, which is $X = M\Sigma N' = (M\sqrt{\Sigma})(N\sqrt{\Sigma})'$, and by setting $U = M\sqrt{\Sigma}$ and $V = N\sqrt{\Sigma}$. □

Proposition 1 guarantees that we can always find an accurate factorization for X using r factors if the rank of X is no more than r, which allows us to bypass the explicit factorization of X and use the following definition to conduct matrix factorization.

Definition 2. *Low Rank Matrix Factorization (LRMF):*

$$\begin{aligned} &\min \mathcal{R}(X) \\ &s.t.\ \mathcal{A}(X)=b,\ \ \mathrm{rank}(X)\le r. \end{aligned} \tag{4}$$

The most frequently used choice for the regularization term in Eq.(1) is the Frobenius norm regularizer, where $\mathcal{R}(U,V) = \|U\|_F^2 + \|V\|_F^2$, and in Eq.(4) is the nuclear norm $\mathcal{R}(X) = \|X\|_*$, where nuclear norm $\|X\|_*$ is defined as the sum of the singular values of X. The following proposition guarantees their equivalence in terms of low rank matrix factorization.

Proposition 2. *[23] Consider the factorization of X in an unlimited dimension factorization space; then the nuclear norm of X can be represented as:*

$$\|X\|_* = \min_{X=UV'} \|U\|_F^2\|V\|_F^2 = \min_{X=UV'} \tfrac{1}{2}(\|U\|_F^2+\|V\|_F^2) \tag{5}$$

In this work, we leverage the Low Rank Matrix Factorization (LRMF) in Definition 2, as well as the minimization of nuclear norm $\|X\|_*$ primarily to analyze the properties of the solution space of Eq.(4), based on which we propose our augmented matrix factorization framework.

4. METHODOLOGY

It is important to notice that most MF algorithms for collaborative filtering only consider the observed values and require the predictions to be close to the corresponding observations, which means that the measurement matrix $A_i = e_je_k' \in \mathbb{R}^{m\times n}$ in Eq.(1) and Eq.(4) has a non-zero value at only a single element corresponding to one of the observed values, and thus the number of constraints p is equal to the number of observations.

As we will show in the following parts, such a measurement set $\mathcal{A}$ is usually insufficient to guarantee a global optimal solution. However, the measurement set $\mathcal{A}$ could have not been restricted to such single-valued constraints. In this section, we analyze the solution space of the LRMF problem and further propose the Augmented Matrix Factorization (AMF) framework for more accurate rating prediction.

4.1 Solution Space Analysis

We bound the solution space of the LRMF problem with the nuclear norm regularizer by analyzing the properties of the linear map $\mathcal{A}$.

Consider solving the constrained optimization problem of LRMF in Eq.(4) with the nuclear norm regularization $\mathcal{R}(X) = \|X\|_*$ using the method of Lagrange multipliers; this yields the global optimal solution X^*, such that $\mathcal{A}(X^*) = b^*$. We then define the adjoint problem as follows:

$$\min \|X\|_* \quad s.t.\ \mathcal{A}(X)=b^*,\ \ \mathrm{rank}(X)\le r \tag{6}$$

We have X^* as one of the exact global optimal solutions for Eq.(6). Suppose the global optimal solutions of this adjoint problem is generally denoted by X, and the residual matrix between X and the known optimal solution X^* is denoted as $R = X^* - X$. We then investigate this residual error under the linear map $\mathcal{A}$ by analyzing $\|\mathcal{A}(R)\|_2^2$, whose ideal value would be zero because $\mathcal{A}(X^*-X) = \mathcal{A}(X^*) - \mathcal{A}(X) = b^* - b^* = 0$. More specifically, we investigate the appropriate properties that $\mathcal{A}$ should satisfy, which guarantees we will search for the global optimal solution in a properly reduced solution space.

4.1.1 Restricted Isometry Property

In this section, we describe the characteristics of an important class of linear maps, which are those that satisfy the *restricted isometry property*, and we bound $\|\mathcal{A}(R)\|_2^2$ under the condition that the linear map $\mathcal{A}$ satisfies this property.

Definition 3. *[10] A linear map $\mathcal{A}: \mathbb{R}^{m\times n} \to \mathbb{R}^p$ is said to satisfy the Restricted Isometry Property (RIP), with the RIP constant δ_r, if, for all of the matrices $X \in \mathbb{R}^{m\times n}$, s.t.* $\mathrm{rank}(X) \le r$*, the following holds:*

$$(1-\delta_r)\|X\|_F^2 \le \|\mathcal{A}(X)\|_2^2 \le (1+\delta_r)\|X\|_F^2 \tag{7}$$

Lemma 1. *According to the definition of the Restricted Isometry Property, we have $\delta_r \le \delta_{r'}$ for $r \le r'$.*

This definition of the RIP is a generalization to matrices from sparse vectors in [9]. The following theorem shows that the solution space of Eq.(6) can be properly bounded given that $\mathcal{A}$ satisfies the RIP.

Theorem 1. *Suppose that $\mathcal{A} : \mathbb{R}^{m\times n} \rightarrow \mathbb{R}^p$ is isometrically restricted, and $R = X^* - X$ is the residual matrix in Eq.(6); then there exists a matrix R_0 whose rank satisfies $\operatorname{rank}(R_0) \leq 2r$, such that $\|\mathcal{A}(R)\|_2^2 = O(r\delta_{2r}\|R_0\|_F^2)$.*

Theorem 1 indicates that, when the linear map $\mathcal{A}$ is isometrically restricted, the solution space of Eq.(6) reduces along with the deceasing of the RIP constant δ. Specially, when δ is small enough to be close to zero, there would be one single global optimal solution for the adjoint problem. The proof of Theorem 1 requires the following lemmas.

Lemma 2. *Let A, B be matrices of the same dimensions. If $AB' = 0$ and $A'B = 0$, then $\|A + B\|_* = \|A\|_* + \|B\|_*$.*

Lemma 3. *[3] For any matrices $A, B \in \mathbb{R}^{m\times n}$ of the same dimensions, there exist matrices B_1 and B_2, such that: (1) $B = B_1 + B_2$; (2) $\operatorname{rank}(B_1) \leq 2\operatorname{rank}(A)$; (3) $AB_2' = 0$ and $A'B_2 = 0$; (4) $\langle B_1, B_2\rangle = 0$.*

Proof. Let the singular value decomposition of A be $A = U\left[\begin{smallmatrix}\Sigma & 0\\ 0 & 0\end{smallmatrix}\right]V'$, and let $\hat{B} = U'BV$. Partition $\hat{B}$ according to the SVD of A as $\hat{B} = \begin{bmatrix}\hat{B}_{11} & \hat{B}_{12}\\ \hat{B}_{21} & \hat{B}_{22}\end{bmatrix}$, and define $B_1 = U\begin{bmatrix}\hat{B}_{11} & \hat{B}_{12}\\ \hat{B}_{21} & 0\end{bmatrix}V'$ and $B_2 = U\begin{bmatrix}0 & 0\\ 0 & \hat{B}_{22}\end{bmatrix}V'$; then B_1 and B_2 satisfy the above four conditions. □

Lemma 4. *For any matrix X such that $\operatorname{rank}(X) \leq r$, we have $\|X\|_F \leq \|X\|_* \leq \sqrt{r}\|X\|_F$.*

Proof of Theorem 1. By applying Lemma 3 to the matrices X and R, there exist matrices R_0 and R_c such that $R = R_0 + R_c$, $\operatorname{rank}(R_0) \leq 2r$, and $XR_c' = 0$ and $X'R_c = 0$. By the optimality of X^*, we have $\|X\|_* \geq \|X^*\|_*$, and it follows that:

$$\begin{aligned}\|X\|_* \geq \|X + R\|_* &\overset{\zeta_1}{\geq} \|X + R_c\|_* - \|R_0\|_*\\ &\overset{\zeta_2}{=} \|X\|_* + \|R_c\|_* - \|R_0\|_*\end{aligned} \tag{8}$$

where ζ_1 follows from the triangle inequality and ζ_2 follows from Lemma 2. It can be further derived from Eq.(8) that $\|R_0\|_* \geq \|R_c\|_*$.

Let $R_c = U\operatorname{diag}(\sigma)V'$ be the SVD of R_c, where the singular values in the diagonal matrix $\operatorname{diag}(\sigma)$ are sorted in descending order. Now we partition R_c into a sum of matrices $R_1, R_2, \cdots$. For each $i \geq 1$, define the index set $\mathcal{I}_i = \{2r(i-1)+1, \cdots, 2ri\}$, and define $R_i = U_{\mathcal{I}_i}\operatorname{diag}(\sigma_{\mathcal{I}_i})V_{\mathcal{I}_i}'$.

By this construction method, we have $\operatorname{rank}(R_i) \leq 2r$ for $i \geq 1$, and that:

$$\sigma_k \leq \frac{1}{2r}\sum_{j\in\mathcal{I}_i}\sigma_j = \frac{1}{2r}\|R_i\|_*, \quad \forall\, i \geq 1,\ k \in \mathcal{I}_{i+1} \tag{9}$$

which further implies that:

$$\|R_{i+1}\|_F^2 = \sum_{k\in\mathcal{I}_{i+1}}\sigma_k^2 \leq \frac{1}{2r}\|R_i\|_*^2, \quad \forall\, i \geq 1 \tag{10}$$

With Eq.(10) and the relationship $\|R_0\|_* \geq \|R_c\|_*$, we can sum up the following bound:

$$\begin{aligned}&\sum_{j\geq 2}\|R_j\|_F^2 \leq \frac{1}{2r}\sum_{j\geq 1}\|R_j\|_*^2 \leq \frac{1}{2r}\Big(\sum_{j\geq 1}\|R_j\|_*\Big)^2\\ &= \frac{1}{2r}\|R_c\|_*^2 \leq \frac{1}{2r}\|R_0\|_*^2 \leq \|R_0\|_F^2\end{aligned} \tag{11}$$

where the last step follows from Lemma 4 and the fact that $\operatorname{rank}(R_0) \leq 2r$. As for $\|R_1\|_F^2$, we have the following:

$$\|R_1\|_F \leq \|R_1\|_* \leq \|R_c\|_* \leq \|R_0\|_* \leq \sqrt{2r}\|R_0\|_F \tag{12}$$

Now we can wrap up the proof to give the following bound for $\|\mathcal{A}(R)\|_2^2$:

$$\begin{aligned}\|\mathcal{A}(R)\|_2^2 &= \|\mathcal{A}(R_0) + \mathcal{A}(R_1) + \sum\nolimits_{j\geq 2}\mathcal{A}(R_j)\|_2^2\\ &\overset{\zeta_1}{\leq} \|\mathcal{A}(R_0)\|_2^2 + \|\mathcal{A}(R_1)\|_2^2 + \sum\nolimits_{j\geq 2}\|\mathcal{A}(R_j)\|_2^2\\ &\overset{\zeta_2}{\leq} (1+\delta_{2r})(\|R_0\|_F^2 + \|R_1\|_F^2 + \sum\nolimits_{j\geq 2}\|R_j\|_F^2)\\ &\overset{\zeta_3}{\leq} 2(1+r)(1+\delta_{2r})\|R_0\|_F^2 = O(r\delta_{2r}\|R_0\|_F^2)\end{aligned} \tag{13}$$

where ζ_1 follows from the triangle inequality, ζ_2 follows from the RIP of $\mathcal{A}$ and the fact that $\operatorname{rank}(R_i) \leq 2r$ for $i \geq 0$, and ζ_3 follows from Eq.(11) and Eq.(12). □

4.1.2 Nearly Isometric Property

In real-world applications, such as CF-based recommender systems, the linear map $\mathcal{A}$ is usually viewed to be generated from some (perhaps unknown) random distributions. For example, it is observed that most real-world datasets in practical recommender systems usually exhibit power-law distributed samples [15, 20]. Unfortunately, the linear map $\mathcal{A}$ corresponding to the original power-law distributed observations hardly ever satisfies the RIP property, which, according to Theorem 1, makes the solution space of MF algorithms poorly bounded and further leads to the multiple global optimal problem.

Consequently, we focus on the task of augmenting the original linear map $\mathcal{A}$ by resampling the unobserved entries in order to add extra constraints on the MF algorithm and bound the solution space properly. As a result, we need to investigate the necessary properties that $\mathcal{A}$ should satisfy under the circumstance of conducting entry sampling. The following definition describes the isometric property in terms of distributions in a probabilistic view.

Definition 4. *Let $\mathcal{A}$ be a random variable that takes values in linear maps from $\mathbb{R}^{m\times n}$ to $\mathbb{R}^p$. We say that $\mathcal{A}$ satisfies the Nearly Isometric Property (NIP) if, for all $X \in \mathbb{R}^{m\times n}$,*

$$\mathbf{E}\left[\|\mathcal{A}(X)\|_2^2\right] = \|X\|_F^2 \tag{14}$$

and, for all $0 < \varepsilon < 1$,

$$\mathbf{P}\left(|\|\mathcal{A}(X)\|_2^2 - \|X\|_F^2| \geq \varepsilon\|X\|_F^2\right) \leq 2\exp\left(-\frac{p}{2}\left(\frac{\varepsilon^2}{2} - \frac{\varepsilon^3}{3}\right)\right) \tag{15}$$

In order for a random linear map to be nearly isometric, Definition 4 requires two essential ingredients. First, it must be isometric in expectation, which is indicated by Eq.(14); and second, the probability of large distortions of length must be exponentially small, as implied by Eq.(15).

Several frequently used linear maps satisfy the NIP in real applications. For easier explanation of such types of linear maps, we introduce the following matrix representation of a linear map:

Definition 5. *Given $X \in \mathbb{R}^{m\times n}$, linear map $\mathcal{A} : \mathbb{R}^{m\times n} \rightarrow \mathbb{R}^p$, and a set of p constraints $\mathcal{A}(X) = b$, where each constraint is of the form $b_i = \langle X, A_i\rangle = \operatorname{tr}(A_i'X)$. Let $d = mn$*

and $x = \mathrm{vec}(X)$ be the vectorization of X. Then, we define the constraint matrix $A = [\mathrm{vec}(A_1)\,\mathrm{vec}(A_2)\cdots\mathrm{vec}(A_p)]' \in \mathbb{R}^{p\times d}$ so that the linear constraints $\mathcal{A}(X) = b$ can be equally reformulated as $Ax = b$.

Generally, it is proven in [7, 11] that, whenever the entries A_{ij} are independently and identically distributed with zero mean and finite fourth moment, the maximum singular value of the constraint matrix A is almost surely $1 + \sqrt{d/p}$ for d that is sufficiently large, and thus the matrix A and the corresponding linear map $\mathcal{A}$ satisfy the NIP.

Specifically, we give some frequently used examples of distributions that satisfy the NIP, which are considered in the experiments primarily for augmenting MF algorithms. The linear map $\mathcal{A}$ satisfies the NIP when A_{ij} is sampled independently and identically from the Gaussian distribution $A_{ij} \sim \mathcal{N}(0, \frac{1}{p})$. The entries sampled from a symmetric Bernoulli distribution also meet the requirements: $\mathbf{P}(A_{ij} = \sqrt{\frac{1}{p}}) = \mathbf{P}(A_{ij} = -\sqrt{\frac{1}{p}}) = \frac{1}{2}$; a generalization of this distribution that takes zeros into account is also frequently used due to its convenience in reducing the number of parameters, which is $\mathbf{P}(A_{ij} = \sqrt{\frac{1}{2p\eta}}) = \mathbf{P}(A_{ij} = -\sqrt{\frac{1}{2p\eta}}) = \eta$, and $\mathbf{P}(A_{ij} = 0) = 1 - 2\eta(0 < \eta < \frac{1}{2})$ [3, 2].

The following theorem characterizes the family of linear maps that satisfy the NIP regarding the upper bound of the isometric constant.

Theorem 2. *[3] Given fixed $0 < \delta < 1$, and let $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ be a linear map that satisfies the NIP; then, for every $1 \le r \le \min(m, n)$, there exist positive constants c_0 and c_1, depending only on δ, such that, with probability of at least $1 - \exp(-c_1 p)$, we have $\delta_r \le \delta$ whenever $p \ge c_0 r(m + n)\log(mn)$.*

The readers might refer to [3] for a proof of this theorem. The theorem indicates that, given $\mathcal{A}$ is nearly isometrically distributed, the upper bound of the isometric constant δ_r can be closely bounded almost surely, as long as the number of linear constraints p of $\mathcal{A}$ is sufficiently large at the order of $O(r(m + n)\log(mn))$. In consideration of Theorem 1, we further arrive at the following corollary.

Corollary 1. *Suppose that the maximum rank r allowed in the LRMF problem defined by Eq.(4) and Eq.(6) satisfies $r \le \min(m, n)/2$, and that $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ is a linear map with the NIP. Then, $\|\mathcal{A}(R)\|_2^2$ reduces almost surely with the increase of p, whenever $p = O(r(m + n)\log(mn))$, where $R = X^* - X$ is the residual matrix.*

Proof. According to the definition of LRMF problem, we have $\mathrm{rank}(X) \le r$ and $\mathrm{rank}(X^*) \le r$; as a result, $\mathrm{rank}(R) = \mathrm{rank}(X^* - X) \le 2r \le \min(m, n)$, and the corollary can now be deduced from Theorem 2. □

This corollary indicates that, under the assumption that the linear constraints are isometrically distributed, the solution space of the LRMF problem reduces with the increase of the number of linear constraints p, as long as $r \le \min(m, n)/2$, which means that the rank requirement is not "too" large. In fact, this requirement is easily satisfied in practical applications of CF, especially in the scenario of LRMF, because the rank of a matrix is usually far less than the number of rows or columns. This is because the preferences of users are usually determined by a limited number of latent factors, *e.g.*, in most MF algorithms the number of factors is set to be a few tens, while the number of rows or columns of the matrix could be in the millions.

4.2 Case Study

According to the theoretical analysis, we see that, for an MF algorithm to find the appropriate global optimal solution in a properly reduced solution space, the following two basic conditions are needed:

- The *number* of constraints should be up to the order of $O(r(m + n)\log(mn))$.
- The *distribution* of the constraints should satisfy the nearly isometric property.

An MF algorithm might fail if either of the two conditions is not satisfied. For a better understanding of the conditions, we present the following two case studies utilizing the previously noted Yelp rating dataset and BDF structured matrices.

Case Study I. The Yelp Rating Dataset.

This dataset violates the first condition. In fact, the Yelp dataset is different from many previous public datasets, and it is in better conformity with practical situations because the previous datasets usually eliminate those users with less than 20 ratings, which relieves them of the multiple global optimal problem to a great extent. The following table lists the statistics of some frequently used datasets.

Table 1: Statistics of some frequently used datasets, where m, n, p are the number of rows, columns and observed values, and $\hat{p}=r(m+n)\log(mn)$ is the required number of constraints, where we use r=10 to keep the magnitude, as r is usually set to be a few tens in practical applications.

Dataset	m	n	p	$\hat{p}$	$p/\hat{p}$
ML-100k	943	1,682	100,000	162,759	0.61
ML-1m	6,040	3,952	1,000,209	737,195	1.36
Netflix	480,189	17,770	100,480,507	49,452,804	2.03
Yahoo	1,000,990	624,961	256,804,235	191,801,776	1.34
Yelp	51,296	12,742	229,907	5,645,155	0.04

We see that the number of constraints in the MovieLens, Netflix and Yahoo! Music datasets are all approximately to the order of the corresponding requirements, while, for the Yelp dataset, they differ by two orders of magnitude. Experimental findings are consistent with the theoretical analysis: previous work conducted on traditional datasets [13, 14, 23, 19] reported significant improvement in terms of prediction accuracy that MF approaches could achieve, while the experimental results show that many MF algorithms perform even worse than a simple global averaging strategy on the Yelp dataset challenge.

Case Study II. The BDF Structured Matrices.

This type of matrix violates the second condition. Note that the number of constraints in a BDF structured matrix could be extremely high. Suppose a BDF matrix has k diagonal blocks, each of which is an $n \times n$ fully filled sub-matrix with n^2 observed values; then we have:

$$\frac{p}{\hat{p}} = \frac{kn^2}{r(2kn)\log(k^2n^2)} = \frac{n}{4r\log(kn)} \gg 1 \qquad (16)$$

However, such a matrix would still suffer from the multiple global optimal problem, as shown by the example in the previous sections of this paper. The underlying reason is that the observed values are all restricted to the diagonal blocks, which makes the constraints biased from the requirement of nearly isometric distributions.

4.3 Augmented Matrix Factorization

The two basic conditions inspire us to augment MF algorithms by adding extra constraints to the linear map $\mathcal{A}$ if the requirement on the number of constraints is not satisfied, and, at the same time, we attempt to make it nearly isometrically distributed, which allows us to reduce the solution space of the LRMF problem. An MF algorithm is then conducted in this reduced solution space for a global optimal solution. Algorithm 1 shows the procedure of augmenting the constraints in the LRMF problem, followed with more detailed explanations and analyses.

```
Algorithm 1: AUGMENT($\mathcal{A}, b, m, n, p, r, c_0, \eta$)
    Input: $\mathcal{A} := \{A_i\}_{i=1}^p, b, m, n, p, r, c_0, \eta$
    Output: $\mathcal{A} := \{A_i\}_{i=1}^p, b, p$
$A \leftarrow [\text{vec}(A_1)\,\text{vec}(A_2)\cdots\text{vec}(A_p)]'$;
$A_0 \leftarrow A,\ b_0 \leftarrow b$;
$\hat{p} \leftarrow c_0 r(m+n)\log(mn),\ d \leftarrow mn$;
if $p < \hat{p}$ then
    $\Delta p \leftarrow \hat{p} - p$;
    $A \leftarrow [A'\ \mathbf{0}_{d\times\Delta p}]',\ b \leftarrow [b',\ \mathbf{0}_{1\times\Delta p}]'$;
end
$p \leftarrow \max(p, \hat{p})$;
for $i \leftarrow 1$ to $p$ do
    for $j \leftarrow 1$ to $d$ do
        if $A_{ij} = 0$ then
            $\omega \leftarrow \text{random}(0,1)$;
            if $\omega < \eta$ then $A_{ij} \leftarrow \sqrt{1/2p\eta}$;
            if $\omega > 1-\eta$ then $A_{ij} \leftarrow -\sqrt{1/2p\eta}$;
        end
    end
end
$b \leftarrow AA_0^+ b_0$;   //$A_0^+$ is the pseudoinverse of $A_0$
$\{A_i\}_{i=1}^p \leftarrow \{\text{vec}^{-1}(A'_{i\cdot})\}_{i=1}^p$; //$A_{i\cdot}$ is the $i^{th}$ row of $A$
return $\mathcal{A} := \{A_i\}_{i=1}^p, b, p$;
```

In this algorithm, we first check whether the original number of constraints p has reached the requirement $\hat{p}$, and if not, the constraint matrix A and vector b are augmented by appending some zero vectors or values, respectively, to gain the required number of constraints.

These augmented constraints, as well as the original ones, are then resampled in the second stage to meet the requirement of nearly isometric distributions. Note that we choose to use the generalized form of the symmetric Bernoulli distribution for constraint resampling because it keeps A_{ij} set to zero with high probability, compared with the Gaussian distribution and symmetric Bernoulli distribution, which benefits the computational time in practical applications. However, it is important to note that the selection of distribution for resampling is not restricted to the generalized form of the symmetric Bernoulli distribution, and any distribution satisfying the NIP can be integrated into the framework of Algorithm 1.

In the last stage, the constraint vector b is reconstructed to match up with the augmented constraint matrix A by multiplying A with the estimated vector $\hat{x} = A_0^+ b_0$, where A_0 and b_0 are the constraint matrix and constraint vector before augmentation, respectively. This estimation given by the pseudoinverse of A_0 is chosen based on the least square property, namely, we have $\|A_0 x - b_0\|_2 \geq \|A_0\hat{x} - b_0\|_2$ for all x. For an arbitrary matrix A_0, the pseudoinverse A_0^+ can be calculated from the singular value decomposition of A_0, which is shown as follows:

$$A_0 = U\Sigma V' \ \Rightarrow\ A_0^+ = V\Sigma^{-1}U' \ \Rightarrow\ \hat{x} = V\Sigma^{-1}U' b_0 \quad (17)$$

The pseudoinverse is chosen also for the purpose of keeping the generality of the algorithm because we do not apply any restriction to the form of the constraint matrix A_0 in the original LRMF problem. However, this also brings about the problem of computational efficiency because the computation of the SVD of a high dimensional matrix could be notably expensive in practical applications.

Fortunately, it is important to note that, in most MF settings, each of the measurement matrices A_i in the linear map $\mathcal{A}$ has only a single non-zero, whose value is one, at the very position corresponding to an observed value in X. More formally, we have $A_i(p,q) = 1$ for the i-th observed value in X, where $X(p,q) \neq 0$. As a result, A_0 is a matrix where there is at most one non-zero in each row and each column, which gives us that $A_0^+ = A_0'$, and we thus have:

$$A_0^+ = A_0' \ \Rightarrow\ \hat{x} = A_0' b \quad (18)$$

Furthermore, we achieve $\|A_0\hat{x} - b_0\|_2^2 = 0$ in such cases. As a result, we need not compute the SVD of the original constraint matrix A_0 in practice, and the estimated vector $\hat{x}$ is achieved by simple matrix-vector multiplications.

When Gaussian fast sampling is used based on the central limit theorem, the computational complexity of Algorithm 1 is quasilinear: $O(p\log(d)) = O(r(m+n)\log^2(mn))$.

4.4 Algorithms for the LRMF Problem

There exist several possible methods for optimizing the LRMF problem in Eq.(1) and Eq.(4), including Semi-Definite Programming (SDP), the interior point methods, projected subgradient methods and low rank parameterization [3].

The SDP solvers and the interior point methods can achieve quite accurate solutions, even to the machine precision. However, they are also rather computationally expensive and could hardly be applied to the large-scale real-world datasets, where the number of rows and/or columns of a matrix could be in the millions. The projected subgradient method is also expensive as it involves the computation of the SVD of a high dimensional matrix as a core stage.

In this work, we adopt the approach of low rank parameterization based on the method of Lagrange multipliers for model learning after the constraints have been augmented, which is shown in Eq.(2) and is a standard method for solving equality constrained optimization problems. According to Proposition 2, this approach applies both to the case when the nuclear norm $\mathcal{R}(X) = \|X\|_*$ is used for regularization and the case when the Frobenius norm $\mathcal{R}(U,V) = \|U\|_F^2 + \|V\|_F^2$ is used [3].

Let $X = UV'$ be the low rank parameterization of X, where $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$ are the low rank parameters, and let $\mathcal{L}(U,V) = \|U\|_F^2 + \|V\|_F^2 + \Lambda\|\mathcal{A}(UV') - b\|_2^2$ be the Lagrangian function. Then the partial deviations are:

$$\begin{aligned}\nabla_U &= U + \Lambda\left(\sum\nolimits_{i=1}^{p}\left(\mathrm{tr}(A_i'UV') - b_i\right)A_i\right)V\\ \nabla_V &= V + \Lambda\left(\sum\nolimits_{i=1}^{p}\left(\mathrm{tr}(A_i'UV') - b_i\right)A_i'\right)U\end{aligned} \tag{19}$$

In the method of multipliers, we minimize the Lagrangian function by updating the decision variables U and V alternately. The minimization of the Lagrangian function in terms of U and V can be conducted using any local search technique. In this work, we adopt the linear search method for the convenience of implementation, and the updating rules for U and V are:

$$U \leftarrow U + \gamma_U \nabla_U, \quad V \leftarrow V + \gamma_V \nabla_V \tag{20}$$

where the corresponding step sizes for U and V are:

$$\begin{aligned}\gamma_U &= -\frac{\mathrm{tr}(\nabla_U' U) + \Lambda\sum_{i=1}^{p}\mathrm{tr}(A_i'\nabla_U V')(\mathrm{tr}(A_i'UV') - b_i)}{\mathrm{tr}(\nabla_U'\nabla_U) + \Lambda\sum_{i=1}^{p}\mathrm{tr}^2(A_i'\nabla_U V')}\\ \gamma_V &= -\frac{\mathrm{tr}(\nabla_V' V) + \Lambda\sum_{i=1}^{p}\mathrm{tr}(A_i'U\nabla_V')(\mathrm{tr}(A_i'UV') - b_i)}{\mathrm{tr}(\nabla_V'\nabla_V) + \Lambda\sum_{i=1}^{p}\mathrm{tr}^2(A_i'U\nabla_V')}\end{aligned} \tag{21}$$

The readers could refer to the supplementary materials[2] for the detailed derivation of Eq.(19)∼(21). The following algorithm shows the procedure of solving the LRMF problem after augmentation, where the parameters N and θ are used to determine when to terminate the algorithm.

```
Algorithm 2: AugmentMF($\mathcal{A}, b, m, n, p, r, c_0, \eta, \Lambda, N, \theta$)
  Input: $\mathcal{A} := \{A_i\}_{i=1}^p, b, m, n, p, r, c_0, \eta, \Lambda, N, \theta$
  Output: $X$
$\mathcal{A} := \{A_i\}_{i=1}^p, b, p \leftarrow$Augment($\mathcal{A}, b, m, n, p, r, c_0, \eta$);
$U \leftarrow \mathbb{R}^{m\times r}$, $V \leftarrow \mathbb{R}^{n\times r}$; //Initialize randomly
$X \leftarrow UV'$, $t \leftarrow 0$;
repeat
    $X^\dagger \leftarrow X$, $t \leftarrow t + 1$;
    Compute $\nabla_U, \gamma_U$ as in Eq.(19) and Eq.(21);
    $U \leftarrow U + \gamma_U \nabla_U$;
    Compute $\nabla_V, \gamma_V$ as in Eq.(19) and Eq.(21);
    $V \leftarrow V + \gamma_V \nabla_V$;
    $X \leftarrow UV'$;
until $\|X - X^\dagger\|_F^2 < \theta$ or $t > N$;
return $X$;
```

The computational complexity of Algorithm 2 under the generalized symmetric Bernoulli distribution is $O(pr(m + n)) = O(r^2(m + n)^2 \log(mn))$, which is unfortunately not quasilinear like that of Algorithm 1. However, the observation that both the gradients in Eq.(19) and the step sizes in Eq.(21) are summed from the p measurement matrices $\{A_i\}_{i=1}^p$ makes it easy to fit into the Map-Reduce parallelization framework. In this work, when the augmentation factor c_0 and the number of latent factors r are given, we use $\lceil c_0 r\rceil$ mapping tasks in the Map-Reduce framework, which makes the computational time comparable to that of solving the original optimization problem without augmentation.

5. EXPERIMENTS

In this section, we conduct extensive experiments to investigate the performance of the proposed Augmented Matrix Factorization (AMF) framework in terms of several important evaluation aspects. We mainly focus on the following two research questions:

1. What is the performance of the framework in terms of rating prediction on highly sparse matrices?
2. What is the performance on the matrices that are highly biased from the nearly isometric distributions?

One can see that these two research questions correspond to the two conditions required of a matrix for reliable rating prediction, which are the number and the distribution of the constraints. Through the experimentations, we would like to investigate the performance of the framework in these two different CF settings.

5.1 Datasets Description

We chose the Yelp[3] and Dianping[4] user-item rating matrix datasets for the experiments. The Yelp rating dataset is from the Yelp dataset challenge, as has been indicated in the previous sections, and the Dianping dataset is collected from the website. Some statistical information about the datasets is shown in the table below:

Table 2: Statistics of the two datasets.

Dataset	m(*#users*)	n(*#items*)	p(*#ratings*)	*density*
Yelp	45,981	11,537	229,907	0.00043
Dianping	8,361	11,392	210,382	0.00221

The Yelp dataset consists of the user ratings of businesses that are mostly located in the city of Phoenix in the US, while the Dianping dataset consists of the ratings on restaurants located in three of the main cities in China: Beijing, Shanghai and Guangzhou.

In the Dianping dataset, the inter-city ratings are far sparser than the inner-city ratings, *e.g.*, users from Beijing would more likely make ratings on Beijing restaurants, while it is rare for them to rate the restaurants in Shanghai or Guangzhou. This makes the Dianping dataset approximately BDF structured, where each of the diagonal matrices represents a city and they are denser than the off-diagonals. We use the Yelp dataset primarily to verify the performance on matrices that violate the first condition and use the Dianping dataset for the second condition.

5.2 Experimental Setup

The test set of the Yelp dataset is not publicly available; as a result, we conduct ten-fold cross-validation on the training set for model learning and evaluation. In the Dianping dataset, we use all of the inner-city ratings and randomly select 20% of the inter-city ratings each time for training, and use the remaining 80% of the inter-city ratings for testing. We also conduct the procedure of training and evaluation 10 times on the Dianping dataset.

The experiments were conducted on a 3.1GHz Linux server with 64 cores and 64GB RAM. Three popular and state-of-the-art MF algorithms were chosen for performance comparisons, which are NMF in [14], PMF in [18] and fast MMMF in [17]. For easy comparison with the previous work, we use Root Mean Square Error (RMSE) as the evaluation metric. We set $\Lambda = 50$ for regularization and use the parameters $\theta = 0.01$ and $N = 100$ in Algorithm 2 to ensure convergence. A series of experiments were conducted to verify the performance of the AMF algorithm in terms of prediction accuracy, stability, convergence rate and computational efficiency.

[2] http://yongfeng.me/attach/amf-supplementary.pdf

[3] http://www.yelp.com/dataset_challenge

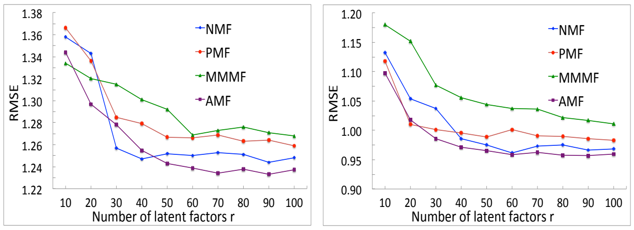
[4] http://www.dianping.com

5.3 Prediction Accuracy

We investigate the prediction accuracy in terms of three important parameters in the AMF algorithm: the number of latent factors r, the augmentation factor c_0, and the sampling rate η in the generalized Bernoulli distribution.

5.3.1 *Number of Latent Factors*

We first investigate the prediction accuracy regarding the number of latent factors r in the low rank parameters U and V. We set $c_0 = 1$ and $\eta = 0.01$ in the AMF algorithm, and the regularization coefficient λ is set to 0.06 in the NMF algorithm of [14]. For PMF in [18], λ_U and λ_V are both set to be 0.005, and the regularization constant $C = 1.5$ in the fast MMMF algorithm of [17]. These hyper parameters are chosen according to grid search-based cross-validation to achieve the best performance for each of the algorithms. We then tune the parameter r in the range of $10 \sim 100$ with a tuning step of 10, and the experimental results are shown in the figures below.



(a) Yelp Dataset (b) Dianping Dataset

Figure 2: RMSE vs different choices of the number of latent factors r in matrix factorization algorithms.

The experimental results show that when the number of latent factors r is sufficient, better performance in terms of RMSE can be achieved in the AMF framework, compared with all of the other three frequently used MF algorithms. The best prediction accuracy and the corresponding r for each of the four algorithms are shown in Table 3.

Table 3: The best prediction accuracy achieved by each MF algorithm on each dataset.

Dataset	NMF		PMF		MMMF		AMF	
	r	RMSE	r	RMSE	r	RMSE	r	RMSE
Yelp	90	1.244	100	1.259	100	1.268	90	**1.233**
Dianping	60	0.961	100	0.983	100	1.011	80	**0.958**

We see that the prediction accuracy tends to be stable with the increase of r. This is because the underlying factors affecting users' decisions are limited, which gives us relatively stable performance when the latent factors used are sufficient. In the following experiments, we set $r = 60$ for all four algorithms on both of the datasets. Applying more factors is allowed, but it is sufficient to achieve stable and satisfactory accuracies according to the experiments.

5.3.2 *Augmentation Factor*

To investigate the relationship of prediction accuracy with the augmentation factor c_0, we fix the parameters $\eta = 0.01$ and $r = 60$. For the Yelp dataset, we tune c_0 in the range of $0.5 \sim 1.5$ with a tuning step of 0.1, and, for the Dianping dataset, c_0 is tuned from 0.5 to 4 with a tuning step of 0.5. Different ranges and tuning steps are used because we find that the optimal augmentation factor c_0 is different on different datasets. The experimental results are plotted in Figure 3.

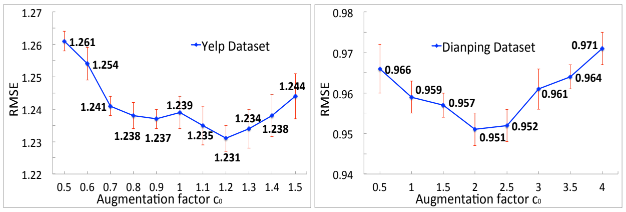


Figure 3: RMSE vs different choices of augmentation factor c_0 in the AMF algorithm.

We see that the AMF framework helps to gain better prediction accuracy when appropriate augmentation factors are given, yet might bring negative effects if c_0 is not appropriately set. It is shown that RMSE decreases with the increase of c_0 at the beginning, until an optimal selection of the augmentation factor, and then tends to increase along with c_0. This is mainly because that sampling noise might be introduced when too many augmented extra constraints are involved. Though these constraints help in reducing the solution space of the LRMF problem and further lead to more stable convergence in fewer iterations, they might guide the problem into a deflected optimal solution.

The best prediction accuracy on Yelp is RMSE = 1.231, with the corresponding augmentation factor $c_0 = 1.2$, and, on Dianping the best performance RMSE = 0.951 is achieved when $c_0 = 2$. In the following experiments, we use the best selection of c_0 for both of the two datasets, correspondingly.

5.3.3 *Sampling Rate*

We investigate the impact of the sampling rate η in the generalized symmetric Bernoulli distribution. In this experiment, we also set $r = 60$ on both datasets and use $c_0 = 1.2$ for Yelp, while using $c_0 = 2$ for Dianping. We tune the parameter η in the range of $10^{-4} \sim 10^{-1}$ with a tuning step of timing 10, and in the range of $0.1 \sim 0.5$ with a step of 0.1. The experimental results are shown in Figure 4 below.

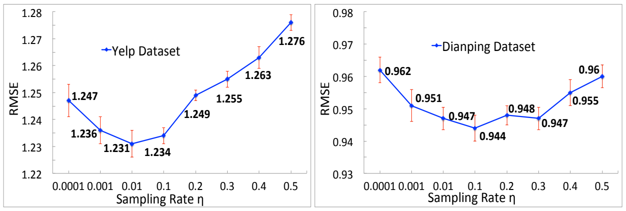


Figure 4: RMSE vs different sampling rates η.

Similarly, it is observed that an appropriate sampling rate is required to achieve the optimal prediction accuracy, and that, whenever η is too small or too large, negative effects might be introduced. Moreover, the prediction accuracy might even be worse than NMF, PMF and MMMF when the sampling rate η is set too high on the Yelp dataset.

To understand the underlying reason for this observation, we go back to the intuitional effect of η in Algorithm 1. One can see that a higher η makes the linear constraints denser, and thus each of the linear constraints, including the augmented ones, involves more parameters in the estimated matrix X. In the extreme case where $\eta = 0.5$, each of the linear constraints attempts to restrict each of the estimations in X. As a result, there, in fact, would be no augmentation to take advantage of if η is too small, while the augmented constraints would counteract or eliminate with each other and even act as noise constraints if η is too large.

5.4 Stability

To verify the effect on solution space reduction of the AMF framework, we investigate the AMF algorithm, as well as the three competing algorithms, in terms of the stability of the final optimal solutions that they converge to. The parameters (r, c_0, η) are assigned as $(60, 1.2, 0.01)$ on Yelp and $(60, 2, 0.1)$ on Dianping, and the parameters for NMF, PMF and MMMF algorithms are the same as those in Section 5.3.1. We calculate the standard deviation σ and the coefficient of variation c_v of the 10 RMSE evaluation results on each of the datasets and for each of the algorithms. The experimental results are shown in Table 4.

Table 4: The standard deviations σ and coefficient of variations c_v of the evaluation results on RMSE.

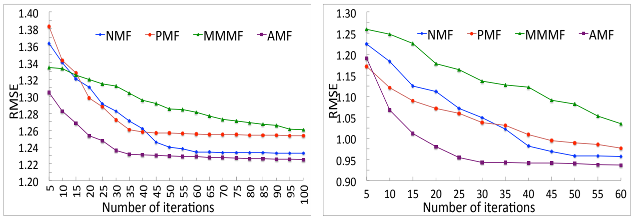
$\sigma, c_v (\times 10^{-2})$	NMF		PMF		MMMF		AMF	
	σ	c_v	σ	c_v	σ	c_v	σ	c_v
Yelp	1.30	1.04	1.72	1.36	0.94	0.74	**0.049**	**0.040**
Dianping	1.86	1.93	2.14	2.13	1.43	1.38	**0.093**	**0.098**

We see that the standard deviation and coefficient of variation in the AMF algorithm are at least an order of magnitude smaller than those of the NMF, PMF and MMMF algorithms. This observation implies that the optimal solutions achieved in the AMF algorithm are more stable and further verifies the fact that the augmented constraints help in reducing the solution space of the LRMF problem, which is in accordance with the theoretical analysis in Section 4.

The experimental results also show that the variation of RMSE is more obvious on the Dianping dataset than on the Yelp dataset. This observation is not surprising because the inter-city ratings falling into the off-diagonal areas are hardly effectively constrained, which further expands the solution space of the LRMF problem. As a result, different initializations of the optimization procedure might result in different optimal solutions. This experimental result is consistent with our solution space analysis on the BDF structured matrices in the previous sections.

5.5 Convergence Rate

In this section, we experiment on the convergence rate of the AMF algorithm in order to further investigate the effect of conducting augmentation on the constraints in the LRMF problem. We use the same parameter assignments as those in Section 5.4 and record the RMSE on the training set in the model learning process for every 5 iterations, which is called an epoch, for both the AMF algorithm and the competing algorithms. The experiments were conducted 10 times, and the average RMSE is calculated on each epoch. The results are plotted in Figure 5.



(a) Yelp Dataset

(b) Dianping Dataset

Figure 5: RMSE on training set vs the number of iterations in the model learning process.

We see that the training loss tends to be stable after approximately 30 ∼ 40 iterations in the AMF algorithm, while it takes 50 or more iterations for NMF and PMF. As for the MMMF algorithm, training loss tends to decrease consistently in the tuning range. This experimental result implies the fast convergence rate of the AMF framework, where the augmented constraints help in guiding the optimization algorithm to converge to an optimal solution.

The underlying reason for this observation can be explained in relation to two aspects. First, the solution space itself has been reduced by incorporating the augmented constraints, and, second, the extra constraints help in calculating a more accurate and rigorous descent gradient in Eq.(19) for model learning in each iteration.

5.6 Computational Efficiency

As the constraints are augmented and extra constraints are introduced in the AMF framework, the computational time is increased remarkably in the model learning process. However, the independence of the constraints makes it easy to conduct optimization in a simple Map-Reduce framework, which makes the computational time comparable to that of the NMF, PMF and MMMF algorithms. In this section, we report the computational efficiency of the AMF algorithm.

We use $\lceil c_0 r \rceil$ mapping tasks in the Map step to compute the gradients ∇_U, ∇_V in Eq.(19) and the step sizes γ_U, γ_V in Eq.(21), and then update U and V in the Reduce step. We still choose the optimal settings of the parameters (r, η), where $(60, 0.01)$ is used for Yelp and $(60, 0.1)$ is used for Dianping. We then tune the parameter c_0 to simulate the augmentation process and record the computational time. Parameters for the NMF, PMF and MMMF algorithms are also the same as those in Section 5.3.1. Experimental results are shown in Figure 6.

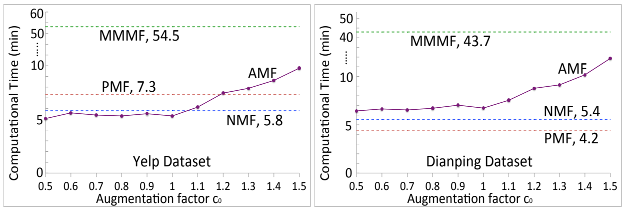


Figure 6: Computational time in minutes vs the augmentation factor c_0 on both datasets.

The results show that the computational time of the AMF algorithm is comparable to the NMF and PMF algorithm under the Map-Reduce framework with $\lceil c_0 r \rceil$ mapping tasks, where each task processes approximately $(m + n) \log(mn)$ constraints, which is numerically comparable to the original number of observed values on both datasets.

One might notice that the computational time begins to rise when $c_0 = 1.1$. This is because the experiments were conducted on a machine with 64 cores and $r = 60$ is set for model learning; thus, when $\lceil c_0 r \rceil$ is greater than 64, the mapping tasks cannot be truly and efficiently distributed among the cores. Nevertheless, the relatively stable computational time when $c_0 \leq 1$ implies that the nature of parallelization of the AMF framework makes it scale efficiently with the processing power in practical applications.

6. DISCUSSIONS

In this section, we discuss aspects of the presented work and note some of the future directions.

The relationship between augmentation and regularization, especially the parallel between augmentation and the Bayesian interpretation of regularization, is of important theoretical interest in conducting matrix completion. For example, the frequently used Frobenius norm regularizer is, actually, guiding the model learning process around a central point under the assumption of Gaussian distribution, while, in the augmentation framework, one can integrate many distributions satisfying the NIP, where the Gaussian distribution is only a special case.

The augmentation framework might also be closely related to other methods like bagging, stacking, or more generally, ensemble learning. In our understanding, the augmentation framework is better than a simple bagging method in that one can appropriately control the resampling procedure to make the constraints satisfy the NIP. However, a deeper relationship therein under the background of matrix completion may bring brand new insights into this well studied problem.

Although the constraints are restricted to linear measurements in this work, they could be much more flexible in that multiple assumptions can be incorporated into the model by means of adding rather "direct" constraints constructed from the observed or unobserved values, instead of only incorporating a single and simple Gaussian distribution assumption in the regularization approach. In addition, we can even integrate various types of external information beyond numerical ratings into the augmented constraints.

We will investigate the deep relationship between augmentation, regularization and ensemble learning both theoretically and practically, as well as take more general non-linear constraints into consideration in the further work.

7. CONCLUSIONS

The problem of data sparsity leads to multiple global optima in matrix factorization algorithms, which further leads to unreliable predictions. In this paper, we investigated the data sparsity with solution space analysis of low rank matrix factorization algorithms, under the conditions of restricted and nearly isometric properties of the linear maps. We found that two basic requirements should be satisfied for reliable completion of matrices in real-world applications, which was verified by the case studies on several frequently used public datasets. Based on these theoretical analyses, we further designed the augmented matrix factorization framework to improve the performance of low-rank matrix factorization. Extensive experimental studies demonstrated the new AMF framework in terms of prediction accuracy, stability, convergence rate and computational efficiency.

Acknowledgement

The authors thank the reviewers for the constructive suggestions. This work was supported by Tsinghua-Samsung Joint Lab under Chinese Natural Science Foundation (60903107, 61073071) and National High Technology Research and Development (863) Program (2011AA01A205), and the third author is sponsored by the National Science Foundation (IIS-0713111). Part of the work was conducted at the Tsinghua-NUS NExT Search Centre, which is supported by the Singapore National Research Foundation & Interactive Digital Media R&D Program Office, MDA under research grant (WBS:R-252-300-001-490).

8. REFERENCES

[1] H. Abdi and L. J. Williams. Principal Component Analysis. *WIREs Comp. Stat.*, 2:433–459, 2010.

[2] D. Achlioptas. Database-Friendly Random Projections: Johnson-Lindenstrauss with Binary Coins. *J. Comput. and System Sci.*, 66:671–687, 2003.

[3] B. Recht, M. Fazel, P. Parrilo. Guaranteed minimum rank solutions of linear matrix equations via nuclear norm minimization. *SIAM*, 52(3):471–501, 2009.

[4] E. Candes, J. Romberg, and T. Tao. Stable Signal Recovery from Incomplete and Inaccurate Measurements. *Commu. on Pure and Appl. Math*, 59(8):1207–1223, 2006.

[5] E. Candes and T. Tao. The Power of Convex Relaxation: Near-Optimal Matrix Completion. *IEEE Trans. on Information Theory*, 56(5):2053–2080, 2010.

[6] E. J. Candes and B. Recht. Exact Matrix Completion via Convex Optimization. *Foundations of Computational Mathematics*, 9:717–772, 2010.

[7] S. Dasgupta and A. Gupta. An Elementary Proof of a Theorem of Johnson and Lindenstrauss. *Jour. Random Structures & Algorithms*, 22(1):60–65, 2003.

[8] D. L. Donoho. Compressed Sensing. *IEEE Trans. on Information Theory*, 52(4):1289–1306, 2006.

[9] E. Candes, T. Tao. Decoding by Linear Programming. *IEEE trans. on info. theory*, 52:4203–4215, 2005.

[10] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank Matrix Completion using Alternating Minimization. *STOC*, 2013.

[11] N. E. Karoui. Recent Results about the Largest Eigenvalue of Random Covariance Matrices and Statistical Application. *Acta Phys. Polo.*, 36(9), 2005.

[12] F. Kiraly and R. Tomioka. A Combinatorial Algebraic Approach for the Identifiability of Low-Rank Matrix Completion. *In Proceedings of ICML*, 2012.

[13] Y. Koren, R. Bell, and C. Volinsky. Matrix Factorization Techniques for Recommender Systems. *Computer*, 2009.

[14] D. D. Lee and H. S. Seung. Algorithms for Non-negative Matrix Factorization. *NIPS*, 2001.

[15] R. Meka, P. Jain, and I. S. Dhillon. Matrix Completion from Power-Law Distributed Samples. *NIPS*, 2009.

[16] P. Jain, R. Meka, I. Dhillon. Guaranteed rank minimization via singular value projection. *NIPS*, 2010.

[17] J. Rennie and N. Srebro. Fast Maximum Margin Matrix Factorization for Collaborative Prediction. *ICML*, 2005.

[18] R. Salakhutdinov and A. Mnih. Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo. *In Proceedings of ICML*, 2008.

[19] R. Salakhutdinov and A. Mnih. Probabilistic Matrix Factorization. *In Proceedings of NIPS*, 2008.

[20] R. Salakhutdinov and N. Srebro. Collaborative Filtring in a Non-Uniform World: Learning with the Weighted Trace Norm. *In Proceedings of NIPS*, 2010.

[21] A. Singer and M. Cucuringu. Uniqueness of Low-Rank Matrix Completion by Rigidity Theory. *SIAM. J. Matrix Anal. and Appl.*, 31:1621–1641, 2010.

[22] A. P. Singh and G. J. Gordon. Relational Learning via Collective Matrix Factorization. *KDD*, 2008.

[23] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-Margin Matrix Factorization. *NIPS*, 2005.

[24] X. Su and T. Khoshgoftaar. A Survey of Collaborative Filtering Techniques. *Advances in AI.*, 2009.

[25] Y. Wang and H. Xu. Stability of Matrix Factorization for Collaborative Filtering. *ICML*, 2012.

[26] Y. Zhang. Browser-oriented universal cross-site recommendation and explanation based on user browsing logs. *RecSys*, 2014.

[27] Y. Zhang, M. Zhang, Y. Liu, and S. Ma. A General Collaborative Filtering Framework based on Matrix Bordered Block Diagonal Forms. *ACM Hypertext*, 2013.

[28] Y. Zhang, M. Zhang, Y. Liu, and S. Ma. Improve Collaborative Filtering Through Bordered Block Diagonal Form Matrices. *SIGIR*, 2013.

[29] Y. Zhang, M. Zhang, Y. Liu, S. Ma, and S. Feng. Localized Matrix Factorization for Recommendation based on Matrix Block Diagonal Forms. *WWW*, 2013.