Journal of Machine Learning Research 19 (2018) 1-35 Submitted 03/17; Revised 05/18; Published 12/18

Using Side Information to Reliably Learn Low-Rank Matrices from Missing and Corrupted Observations

Kai-Yang Chiang KYCHIANG@CS.UTEXAS.EDU
Inderjit S. Dhillon INDERJIT@CS.UTEXAS.EDU
Department of Computer Science
University of Texas at Austin
Austin, TX 78701, USA
Cho-Jui Hsieh CHOHSIEH@UCDAVIS.EDU
Department of Statistics and Computer Science
University of California at Davis
Davis, CA 95616, USA

Editor: Sanjiv Kumar

Abstract

Learning a low-rank matrix from missing and corrupted observations is a fundamental problem in many machine learning applications. However, the role of *side information* in low-rank matrix learning has received little attention, and most current approaches are either ad-hoc or only applicable in certain restrictive cases. In this paper, we propose a general model that exploits side information to better learn low-rank matrices from missing and corrupted observations, and show that the proposed model can be further applied to several popular scenarios such as matrix completion and robust PCA. Furthermore, we study the effect of side information on sample complexity and show that by using our model, the efficiency for learning can be improved given sufficiently informative side information. This result thus provides theoretical insight into the usefulness of side information in our model. Finally, we conduct comprehensive experiments in three real-world applications—relationship prediction, semi-supervised clustering and noisy image classification, showing that our proposed model is able to properly exploit side information for more effective learning both in theory and practice.

Keywords: Side information, low-rank matrix learning, learning from missing and corrupted observations, matrix completion, robust PCA

1. Introduction

Learning a low-rank matrix from noisy, high-dimensional complex data is an important research challenge in modern machine learning. In particular, in the recent big data era, assuming that the observations come from a model with implicit low-rank structure is one of the most prevailing approaches to avoid the curse of dimensionality. While various low-rank matrix learning problems arise from different contexts and domains, the primary challenge is rather similar: namely to reliably learn a low-rank matrix L_0 based only on missing and corrupted observations from L_0. This generic framework includes many well-known machine learning problems such as matrix completion (Candès and Tao, 2009), robust PCA (Wright et al., 2009) and matrix sensing (Zhong et al., 2015), and is shown to be

useful in many important real-world applications including recommender systems (Koren et al., 2009), social network analysis (Hsieh et al., 2012) and image processing (Wright et al., 2009).

Among research related to low-rank matrix learning, one promising direction is to further exploit *side information*, or *features*, to help the learning process. [1] The notion of side information appears naturally in many applications. For example, in the famous Netflix problem where the goal is movie recommendation based on users' ratings, a popular approach is to assume that the given user-movie rating pairs are sampled from a low-rank matrix (Koren et al., 2009). However, besides rating history, profiles of users and/or genres of movies may also be provided, and one can possibly leverage such side information for better recommendation. Since such additional features are available in many applications, designing a model to better incorporate features into low-rank matrix learning problems becomes an important issue with both theoretical and practical interests.

Motivated by the above realization, we study the effect of side information on learning low-rank matrices from missing and corrupted observations in this paper. Our general problem setting can be formally described as follows. Let $L_0 \in \mathbb{R}^{n_1 \times n_2}$ be the low-rank modeling matrix, yet due to various reasons we can only observe a matrix $R \in \mathbb{R}^{n_1 \times n_2}$ which contains missing and/or corrupted observations of L_0. In addition, suppose we are also given additional feature matrices $X \in \mathbb{R}^{n_1 \times d_1}$ and/or $Y \in \mathbb{R}^{n_2 \times d_2}$ as side information, where each row $\mathbf{x}_i \in \mathbb{R}^{d_1}$ (or $\mathbf{y}_i \in \mathbb{R}^{d_2}$) denotes a feature representation of the i-th row (or column) entity of X (or Y). Then, instead of just using R to recover L_0, our hope is to leverage side information X and Y to learn L_0 more effectively. Below, we further list some important applications where the side information naturally comes in as the form of X and/or Y in this framework:

- *Collaborative filtering.* Collaborative filtering is one of the most popular machine learning applications in industry where we aim to predict the preferences of users to any products based on limited rating history (e.g. the Netflix problem we mentioned previously). A traditional approach is to complete the partial user-product rating matrix R via matrix completion. However, one could also collect per-user features x_i and per-product features y_j as possible information to leverage, and the assembled feature representation for users and products becomes X and Y in this framework.

- *Link prediction.* The link prediction problem in online social network analysis is to predict and recommend the implicit friendships of users given the current network snapshot. One approach is to think of the network snapshot as a user-to-user relationship matrix R, and thus any missing relationships in the snapshot can be inferred by conducting matrix completion on R (Liben-Nowell and Kleinberg, 2007). Similarly, if user-specific information (like user profile) is collected, these user features can be deemed as both X and Y.

- *Image denoising.* Another low-rank matrix learning application is image denoising. It is known that same types of images (e.g. images of human face, digits, or images with same scene) often share a common low-rank structure, and learning that low-dimensional space can be useful for many applications such as image recognition

1. We will use terms 'side information' and 'features' interchangeably throughout the paper.

and background subtraction. Yet in the realistic setting, images may be corrupted by sparse noise such as shadowing or brightness saturation, making the learning of that low-dimensional space much more difficult. A popular approach, known as robust PCA, is to construct an observed matrix R where each column is a vector representation of an image, and further learn the underlying low-rank subspace by separating it from the sparse noise in R. In Section 4, we will show that if features of clean images X and/or label-relevant features Y are also given, one can learn the underlying low-dimensional subspace more accurately.

Organization of the paper. To study the effect of side information in low-rank matrix learning with missing and corrupted observations, we focus on answering the following important questions in a systematical manner:

- What type of side information can benefit learning?
- What model should we use for incorporating side information?
- How can we further quantify the merits of side information in learning?

Regarding the first question, in Section 2, we start with the case of *"perfect" side information* (defined in equation 2) as an idealized case where the given features are fully informative, and further generalize to the case of *noisy side information* where the given features are only partially correlated to L_0. We will see that while information from perfect features is extremely useful, certain noisy features can also be quite effective to benefit learning.

The model for incorporating side information can also be constructed subsequently once the type of side information is identified. Precisely, in Section 2, we argue that for perfect features, one can directly transform the low-rank modeling matrix into a bilinear form with respect to features X and Y. However, the validity of such an embedding becomes questionable if features are noisy. Therefore, for noisy features, we propose to break the low-rank matrix into two parts—one that captures information from features and one that captures information outside the feature space—resulting in a general model (problem 4) that learns the low-rank matrix by jointly balancing information from noisy features and observations. In addition, we discuss the connections between our model and several well-known models, such as low-rank matrix completion and robust PCA. We also show that our proposed model can be efficiently solved by well-established optimization procedures.

Furthermore, in Section 3, we provide a theoretical analysis to justify the merits of side information in the proposed model (4). To start with, in Section 3.1, we quantify the quality of features and the noise level of corruption using Rademacher model complexity in the generalization analysis. As a result, a tighter error bound can be derived given better quality of features and/or lower noise level in observations. We further derive sample complexity guarantees for the case of matrix completion in Section 3.2 and for the case where observations are both missing and corrupted in Section 3.3. For the case of matrix completion, our sample complexity result suggests that the proposed model requires asymptotically fewer observations to recover the low-rank matrix compared to standard matrix completion, as long as the given features are sufficiently informative. This result substantially generalizes the previous study of side information in matrix completion in Jain and Dhillon (2013) which only guarantees improved complexity given perfect features. On the other hand, for

the case where observations are both missing and corrupted, our resulting sample complexity guarantee implies that better quality of side information is useful for learning missing entries of the low-rank matrix provided that the corruption is not too severe. These results thus justify the usefulness of side information in the proposed model in theory.

Finally, in Section 4, we verify the effectiveness of the proposed model experimentally on various synthetic data sets, and additionally apply it to three machine learning applications—relationship prediction, semi-supervised clustering and noisy image classification. We show that each of them can be tackled by learning a low-rank modeling matrix from missing or corrupted observations given certain additional features, and therefore, by employing our model to exploit side information, we can achieve better performance in these applications compared to other state-of-the-art methods. These results demonstrate that our proposed model indeed exploits side information for various low-rank matrix learning problems.

Here are the key contributions of this paper:

- We study the effect of side information and provide a general treatment to incorporate side information for learning low-rank matrices from missing and corrupted observations.

- In particular, given perfect side information, we propose to transform the estimated low-rank matrix to a bilinear form with respect to features. Moreover, given noisy side information, we propose to further break the low-rank matrix into a part capturing feature information plus a part capturing information outside the feature space, and therefore, learning can be conducted efficiently by balancing information between features and observations.

- We theoretically justify the usefulness of side information in the proposed model in various scenarios by first quantifying the effectiveness of features and then showing that the sample complexity can be asymptotically improved provided sufficiently informative features.

- We provide comprehensive experimental results to confirm that the proposed model properly embeds both perfect and noisy side information for learning low-rank matrices more effectively compared to other state-of-the-art approaches.

Parts of this paper have previously appeared in Chiang et al. (2015) and Chiang et al. (2016), in which we exclusively studied the effect of noisy side information in matrix completion and the effect of perfect side information in robust PCA, respectively. In this paper, we consider a much more general setting and propose a general model to exploit side information for a broader class of low-rank matrix learning problems. In particular, given this general model, we can further exploit noisy side information for the robust PCA problem and for the case where observations are *both* missing and corrupted as we will discuss in Section 2.3. We also provide much more comprehensive theoretical and experimental results to demonstrate the effectiveness of the proposed treatment.

2. Exploiting Side Information for Learning Low-Rank Matrices

In this section, we discuss how to incorporate side information for learning low-rank matrices from missing and corrupted observations. We first introduce the problem formulation in Section 2.1. We then start with exploiting perfect, noiseless side information in Section 2.2 and introduce the proposed model which can further exploit noisy side information in Section 2.3. We finally describe the optimization for solving the proposed model in Section 2.4.

2.1. Learning from Missing and Corrupted Observations

The problem of learning a low-rank matrix from missing and corrupted observations can be formally stated as follows. Let $L_0 \in \mathbb{R}^{n_1 \times n_2}$ be the underlying rank-r matrix where $r \ll \min(n_1, n_2)$ so that L_0 is low-rank, and S_0 be a noise matrix whose support (denoted as Ω) and magnitude is unknown but the structure is known to be *sparse*. Furthermore, let Ω_{obs} be a set of observed entries with cardinality m, and $\mathcal{P}_{\Omega_{obs}}$ be the orthogonal projection operator defined by:

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

Then, given the observed data matrix R which is in the form of:

$$R = \mathcal{P}_{\Omega_{obs}}(L_0 + S_0) = \mathcal{P}_{\Omega_{obs}}(L_0) + S_0',$$

the goal is to accurately estimate the underlying matrix L_0 given R. Without loss of generality, we assume that S_0 is supported on Ω_{obs}, i.e. $\Omega \subseteq \Omega_{obs}$ and $S_0' = S_0$. Note that this problem can be viewed as an extension of the matrix completion problem, which only assumes the given observations to be undersampled yet noiseless (Ω is the empty set).

An intuitive way to approach this problem is to estimate the low-rank matrix based on the given structural information of the problem. Specifically, Candès et al. (2011) proposed to solve this problem via the following convex program:

$$\min_{L,S} \ \|L\|_* + \lambda \|S\|_1 \quad \text{s.t.} \quad L_{ij} + S_{ij} = R_{ij}, \ \forall \ (i,j) \in \Omega_{obs}, \tag{1}$$

where $\|L\|_*$ is the nuclear norm of L defined by the sum of singular values of L, and $\|S\|_1 := \sum_{i,j} |S_{ij}|$ is the element-wise one norm of S. These two regularizations are known to be useful for enforcing low rank structure and sparse structure, respectively.

Although problem (1) has been shown to enjoy theoretical and empirical success (Candès et al., 2011), it cannot directly leverage side information for recovery if it is provided. A tailored model is thus required to resolve this issue.

2.2. Idealized Case: Perfect Side Information

Suppose in addition to the data matrix R, we are also given features of row and column entities $X \in \mathbb{R}^{n_1 \times d_1}$ and $Y \in \mathbb{R}^{n_2 \times d_2}$, $d_1 < n_1$ and $d_2 < n_2$ as side information. Then, the goal of low-rank matrix learning with side information is to exploit X and Y in addition to the observations R to better estimate L_0. A concrete example is the Netflix problem where R corresponds to the partial user-movie rating matrix and, X and Y correspond to user

and movie features; the hope is to further leverage additional features X and Y along with rating history R to better predict the unknown user-movie ratings.

In principle, not all types of side information will be useful. For instance, if the given X and Y are simply two random matrices, then there is no information gain from the provided side information, and therefore, *any* method incorporating such X and Y is expected to perform the same as methods only using structural information. That being said, to explore the advantage of side information, a condition on side information to ensure its informativeness is required. To begin with, we consider an ideal scenario where the side information is "perfect" in the sense that it implicitly describes the full latent space of L_0.

Definition 1 (Perfect side information) *The side information X and Y is called perfect side information, or noiseless side information, w.r.t. L_0 if X and Y satisfy:*

$$col(X) \supseteq col(L_0), \quad col(Y) \supseteq col(L_0^T), \tag{2}$$

where $col(X)$ and $col(Y)$ denotes the column space of X and Y.

Then, consider $L_0 = U\Sigma V^T$ to be the SVD of L_0, a set of perfect side information will also satisfy $\text{col}(X) \supseteq \text{col}(U)$ and $\text{col}(Y) \supseteq \text{col}(V)$, which further indicates that there exists a matrix $M_0 \in \mathbb{R}^{d_1 \times d_2}$ such that $L_0 = XM_0Y^T$. This fact leads us to expressing the target low-rank matrix as a bilinear form with respect to features X and Y, and as a result, one can cast problem (1) with features as:

$$\min_{M,S} \ \|M\|_* + \lambda\|S\|_1 \quad \text{s.t. } \mathbf{x}_i^T M \mathbf{y}_j + S_{ij} = R_{ij}, \ \forall \ (i,j) \in \Omega_{obs}, \tag{3}$$

in which the problem is reduced to learning a smaller $d_1 \times d_2$ low-rank matrix M. The bilinear embedding with respect to perfect features for the low-rank matrix has already been proposed in matrix completion. Indeed, by casting $L = XMY^T$ as matrix completion, one can obtain a so-called "inductive matrix completion" (IMC) model which is able to learn the underlying matrix with much fewer samples given perfect side information (Jain and Dhillon, 2013; Xu et al., 2013; Zhong et al., 2015). We will discuss the improved sample complexity result of IMC in detail in Section 3.2.

However, an obvious weakness of the bilinear embedding in problem (3) is that it assumes the given side information to be perfect. Unfortunately, in real applications, most given features X and Y will not be perfect, and could be in fact noisy or only weakly correlated to the latent space of L_0. In such cases, L_0 can no longer be expressed as XMY^T and thus the translated objective (3) becomes questionable to use. This weakness will also be empirically shown in Section 4 in which we observe that the recovered matrix XM^*Y^T of problem (3) will diverge from L_0 given noisy side information in experiments. Nevertheless, it is arguable that certain noisy features should still be helpful for learning L_0. For example, given the SVD of $L_0 = U\Sigma V^T$, a small perturbation of a single entry of U (or V) makes the perturbed U, V to be imperfect features, yet such U and V should still be very informative. This observation thus motivates us to design a more general model to exploit noisy side information.

2.3. The Proposed Model: Exploiting Noisy Side Information

We now introduce an improved model to further exploit imperfect, noisy side information. The key idea of our model is to balance both feature information and observations when learning the low-rank matrix. Specifically, we propose to learn L_0 jointly in two parts, one part captures information from the feature space as XMY^T, and the other part N captures the information outside the feature space. Thus, even if the given features are noisy and fail to cover the full latent space of L_0, we can still capture missing information using N learned from pure observations.

However, there is an identifiability issue if we simply learn L_0 with the expression $XMY^T + N$, since there are infinitely many solutions of (M, N) that satisfy $XMY^T + N = L_0$. Although in theory they all perfectly recover the underlying matrix, some of the solutions shall be more preferred than others if we further consider the efficiency of learning. Intuitively, since the underlying L_0 is low-rank, a natural thought is to prefer both XMY^T and N to be low-rank so that the L_0 can be recovered with fewer parameters. This preference leads us to pursue a low-rank M as well, which conceptually means that only a small subspace of X and a subspace of Y are expected to be effective in jointly forming a low-rank estimate XMY^T. Pursuing low-rank solutions of M and N enables us to accurately estimate L_0 with fewer samples because fewer parameters need to be learned compared to other solutions. This advantage will be formally justified later in Section 3.

Therefore, putting this all together, to incorporate noisy side information and learn the low-rank matrix L_0 from missing and corrupted observations, we propose to solve the following problem:

$$\min_{M,N,S} \sum_{(i,j)\in\Omega_{obs}} \ell((XMY^T + N + S)_{ij}, R_{ij}) + \lambda_M\|M\|_* + \lambda_N\|N\|_* + \lambda_S\|S\|_1 \qquad (4)$$

with some convex surrogate loss ℓ, and the underlying matrix L_0 can be estimated by $XM^*Y^T + N^*$, where (M^*, N^*, S^*) is the optimal solution of problem (4). Note that to force M and N to be low-rank, in the proposed objective we add nuclear norm regularization on *both* variables M and N. It is known that nuclear norm regularization is one of the most popular heuristic to pursue low-rank structure as it is the tightest convex relaxation of the rank function (Fazel et al., 2001). In particular, given a low-rank matrix $\text{rank}(R) \le r$ and $\max_{ij}|R_{ij}| \le C_L$, we always have:

$$\|R\|_* \le \sqrt{r}\|R\|_F \le C_L\sqrt{rn_1n_2},$$

and thus, a nuclear norm regularized constraint $\|R\|_* \le t$ can be thought of as a relaxed condition of $\text{rank}(R) \le r$ and $\max_{ij}|R_{ij}| \le t/\sqrt{rn_1n_2}$.

The proposed problem (4) is also a general formulation to better exploit side information for learning low-rank matrices from missing and corrupted observations. This fact can be seen by considering the following equivalent form of problem (4) which converts the loss term to hard constraints:

$$\min_{M,N,S} \alpha\|M\|_* + \beta\|N\|_* + \lambda\|S\|_1 \quad \text{s.t. } (XMY^T + N + S)_{ij} = R_{ij}, \forall(i,j) \in \Omega_{obs}. \qquad (5)$$

Then, it is easy to see that by setting $\alpha = \infty$ or $\beta = \infty$, problem (5) will become problem (1) or problem (3), which learns the low-rank matrix from missing and corrupted observations

either without any side information or using perfect side information, respectively. This suggests that our model (4) is more general as it can exploit both perfect and noisy side information in learning.

The parameters λ_M, λ_N and λ_S of the model are crucial for controlling the contributions from features, observations and corruption. Intuitively, λ_S controls the ratio of corrupted observations. The relative weight between λ_M and λ_N further controls the contributions from XMY^T and N in forming the low-rank estimate. Therefore, with an appropriate ratio between λ_M, λ_N, the proposed model can leverage a (informative) part of the features XMY^T, yet also be robust to feature noise by learning the remaining part N from pure observations. Below, we further discuss the connections between our model (4) and other well-known models for solving various low-rank matrix learning problems.

2.3.1. Connections to models for matrix completion

First, consider the matrix completion case where the partially observed entries are not corrupted. Then, λ_S can be set to ∞ to force $S^* = 0$, and therefore, our proposed problem (4) reduces to the following objective:

$$\min_{M,N} \sum_{(i,j)\in\Omega_{obs}} \ell((XMY^T + N)_{ij}, R_{ij}) + \lambda_M \|M\|_* + \lambda_N \|N\|_*, \tag{6}$$

which is a general model for solving matrix completion problem. For example, when $\lambda_M = \infty$, M^* will be forced to 0 so features are disregarded, and problem (6) becomes a standard matrix completion objective. On the other hand, when $\lambda_N = \infty$, N^* will be forced to 0 and problem (6) becomes the IMC model (Jain and Dhillon, 2013; Xu et al., 2013) where the estimation of the low-rank matrix is completely from XM^*Y^T. However, problem (6) is more general than both problems, since by appropriately setting the weights of λ_M and λ_N, it can better estimate the low-rank matrix jointly from (noisy) features XM^*Y^T and pure observations N^*. Therefore, problem (6) can be thought of as an improved model which exploits noisy side information in matrix completion problem. We thus refer to problem (6) as "IMC with Noisy Features" (IMCNF) and will justify its effectiveness for matrix completion in Section 4.

2.3.2. Connections to models for robust PCA

Another special case is to consider the well-known "robust PCA" setting, in which Ω_{obs} is assumed to be the set of all $n_1 \times n_2$ entries, i.e. observations are full without any missing entries but few of them are corrupted. In this scenario, our proposed problem (4) can be used for solving robust PCA problem with side information by again converting the loss term to hard constraints:

$$\min_{M,N,S} \ \alpha\|M\|_* + \beta\|N\|_* + \lambda\|S\|_1 \quad \text{s.t. } XMY^T + N + S = R. \tag{7}$$

Problem (7) can be further reduced to several robust PCA models. For example, if $\alpha = \infty$, problem (7) will be equivalent to the well-known PCP method (Candès et al., 2011) which solves robust PCA problem purely using a structural prior. On the other hand, suppose side information is perfect, then one can set $\beta = \infty$ in (7) to derive the following "PCP

Model	Corresponding setting in our proposed model (4)
problem (1) (Candès et al., 2011)	$\lambda_M = \infty$
problem (3)	$\lambda_N = \infty$
MC	$\lambda_S = \infty, \lambda_M = \infty$
IMC (Jain et al., 2013)	$\lambda_S = \infty, \lambda_N = \infty$
IMCNF	$\lambda_S = \infty$
LRR (Liu et al., 2013)	Ω_{obs} = all entries, $\lambda_N = \infty$, $Y = I$
PCP (Candès et al., 2011)	Ω_{obs} = all entries, $\lambda_M = \infty$
PCPF	Ω_{obs} = all entries, $\lambda_N = \infty$
PCPNF	Ω_{obs} = all entries

Table 1: Settings of several low-rank matrix learning models in the form of our proposed problem (4).

with (perfect) Features" (PCPF) objective:

$$\min_{M,S} \ \alpha\|M\|_* + \lambda\|S\|_1 \quad \text{s.t. } XMY^T + S = R, \tag{8}$$

in which L_0 can be directly estimated by the bilinear embedding XM^*Y^T as discussed in Section 2.2. However, problem (7) is more general than both PCP and PCPF as it can exploit noisy side information for recovery. We thus refer to (7) as "PCP with Noisy Features" (PCPNF) and will examine its effectiveness to leverage noisy side information in robust PCA in Section 4.

Table 1 summarizes several well-known low-rank matrix learning models in terms of the proposed model (4). [2] From the above discussion, it shall be convincing that problem (4) is a general treatment for solving various matrix learning problems with side information. In particular, we have provided sufficient intuitions on how parameters λ_M, λ_N and λ_S play important roles in learning under various circumstances. In Section 3, we will further analytically show that by properly setting these parameters based on the quality of features and noise level of corruption, the proposed model is able to achieve more efficient learning. As a remark, in practical applications, feature quality and noise level may not be known a priori. Therefore, in this case, we recommend to set these parameters via validation, i.e. choosing parameters such that the learned low-rank model best estimates the entries in the validation set.

2.4. Optimization

We propose an alternative minimization scheme to solve the proposed problem (4). The algorithm is shown in Algorithm 1 in which we alternatively update one of the variables (M, N or S) by fixing the others in each iteration, [3] and update of each variable can thus be done via solving a single variable minimization (sub)problem. This algorithm can be viewed as applying a block coordinate descent algorithm on a convex and continuous function, and in

2. Some models are originally proposed in hard-constrained forms, yet their equivalent forms in soft constraints become instances of our proposed problem (4).
3. For simplicity, we choose $\ell(t, y) = (t - y)^2$ to be the squared loss.

Algorithm 1: Alternative Minimization for Problem (4) with Squared Loss

Input: R: observed matrix, X, Y: feature matrices, t_{max}: max iteration
Output: L^*: estimated low-rank matrix
$M \leftarrow 0, \quad N \leftarrow 0, \quad S \leftarrow 0, \quad t \leftarrow 0$
do
  $M \leftarrow \arg\min_M \sum_{(i,j)\in\Omega_{obs}} ((XMY^T)_{ij} - (R - N - S)_{ij})^2 + \lambda_M \|M\|_*$.
  $N \leftarrow \arg\min_N \sum_{(i,j)\in\Omega_{obs}} (N_{ij} - (R - XMY^T - S)_{ij})^2 + \lambda_N \|N\|_*$.
  $S \leftarrow \arg\min_S \sum_{(i,j)\in\Omega_{obs}} (S_{ij} - (R - XMY^T - N)_{ij})^2 + \lambda_S \|S\|_1$.
  $t \leftarrow t + 1$.
while *not converged* **and** $t < t_{\max}$
$L^* \leftarrow XMY^T + N$

such case the cyclic block coordinate descent algorithm is guaranteed to converge to global minimums (see Tseng, 2001). The condition required in Tseng (2001) is that the level set has to be compact, which is satisfied when $\lambda_M, \lambda_N, \lambda_S > 0$.

We now briefly discuss the optimization for solving three subproblems in Algorithm 1. Let $\mathcal{S}_x(A) := \text{sign}(A) \circ \max(|A| - x, 0)$ be the soft thresholding operator on elements of A, where $\circ$ denotes the element-wise product. Similarly, let $\mathcal{D}_x(A)$ be the thresholding operator on singular values of A, i.e. $\mathcal{D}_x(A) := U_A S_x(\Sigma_A) V_A^T$ where $U_A \Sigma_A V_A^T$ is the SVD of A. Then, when fixing N and S, the minimization problem over M becomes a standard IMC objective with observed matrix to be $R' := R - N - S$. We then solve for M using typical proximal gradient descent update $M \leftarrow \mathcal{D}_{\lambda_M}(M - \eta X^T(R' - XMY^T)Y)$, where η is the learning rate. Notice that in our setting, feature dimensions (d_1, d_2) are much smaller than number of entities (n_1, n_2). Therefore, it is relatively inexpensive to compute a full SVD for a $d_1 \times d_2$ matrix in each proximal step.

On the other hand, when fixing M and S, the subproblem of solving over N becomes standard matrix completion problem where the observed matrix is $R - XMY^T - S$. In principle, any algorithm for matrix completion with nuclear norm regularization can be used to solve this subproblem (e.g. the singular value thresholding algorithm (Cai et al., 2010) using proximal gradient descent). In our experiment, we apply the active subspace selection algorithm (Hsieh and Olsan, 2014) to solve the matrix completion problem more efficiently.

Finally, the solution of minimizing over S given fixed M, N can be written in a simple closed form, $\mathcal{S}_{\lambda_S}(\mathcal{P}_{\Omega_{obs}}(R - XMY^T - N))$. The resulting S^*, therefore, will be always supported on Ω_{obs}.

3. Theoretical Analysis on the Effect of Side Information

In this section, we provide a theoretical analysis to justify the usefulness of side information in our model (4). We will focus on the *sample complexity* analysis of the model, in which we aim to show that by exploiting side information, learning can be accomplished with fewer number of (possibly corrupted) observations. The high-level idea of the analysis is to consider the generalization error of the estimated entries, which is associated to both

number of samples and a model complexity term. We further show that model complexity can be related to the quality of features and the noise level of sparse error, and as a result, better feature quality will lead to a smaller generalization error and also a better sample complexity guarantee, provided a small enough noise level. To concentrate on the whole picture of the analysis, we leave detailed proofs of theorems, corollaries and lemmas in Appendix A.

3.1. Generalization Bound of the Proposed Model

To begin with, we consider the equivalent hard-constrained form of problem (4):

$$\min_{M,N,S} \sum_{(i,j)\in\Omega_{obs}} \ell((XMY^T + N + S)_{ij}, R_{ij}), \quad s.t.\ \|M\|_* \le \mathcal{M}, \|N\|_* \le \mathcal{N}, \|S\|_1 \le \mathcal{S}. \quad (9)$$

In the analysis, we assume that each entry $(i,j) \in \Omega_{obs}$ is sampled i.i.d. from an *unknown* distribution $\mathcal{D}$ with index set $\{(i_\alpha, j_\alpha)\}_{\alpha=1}^m$, [4] and each entry of L_0 is upper bounded by a constant C_L (so $\|L_0\|_* = O(\sqrt{n_1 n_2})$). Such a circumstance is consistent with real scenarios such as Netflix problem where users can rate movies with scale up to 5. Let $\theta := (M, N, S)$ be any feasible solution and $\Theta := \{(M,N,S) \mid \|M\|_* \le \mathcal{M}, \|N\|_* \le \mathcal{N}, \|S\|_1 \le \mathcal{S}\}$ be the set of feasible solutions. Also, let $f_\theta \in [n_1] \times [n_2] \to \mathbb{R}$, $f_\theta(i,j) := \mathbf{x}_i^T M \mathbf{y}_j + \mathbf{e}_i^T N \mathbf{e}_j + \mathbf{e}_i^T S \mathbf{e}_j$ be the estimation function (parameterized by θ) where $\mathbf{e}_t$ is the unit vector on the t-th axis, and let $F_\Theta := \{f_\theta \mid \theta \in \Theta\}$ be the set of feasible functions. We are interested in both expected and empirical "ℓ-risk" quantities, $R_\ell(f)$ and $\hat{R}_\ell(f)$, defined by:

$$R_\ell(f) := \mathbb{E}_{(i,j)\sim\mathcal{D}}\big[\ell(f(i,j), \mathbf{e}_i^T(L_0 + S_0)\mathbf{e}_j)\big], \qquad \hat{R}_\ell(f) := \frac{1}{m} \sum_{(i,j)\in\Omega_{obs}} \ell(f(i,j), R_{ij}).$$

Under this context, our model (problem 9) is to solve for θ^* that parameterizes $f^* = \arg\min_{f\in F_\Theta} \hat{R}_\ell(f)$. Classic generalization error bounds have shown that the expected risk $R_\ell(f)$ can be controlled by $\hat{R}_\ell(f)$ along with a measurement on the complexity of the model. The following lemma is a typical result to bound $R_\ell(f)$:

Lemma 2 (Bound on Expected ℓ-risk, Bartlett and Mendelson, 2003) *Let ℓ be a Lipschitz loss function and is bounded by $\mathcal{B}$ with respect to its first argument, and δ be a constant where $0 < \delta < 1$. Let $\mathfrak{R}(F_\Theta)$ be the Rademacher model complexity of the function class F_Θ (w.r.t. Ω_{obs}) defined by:*

$$\mathfrak{R}(F_\Theta) := \mathbb{E}_\sigma\big[\sup_{f\in F_\Theta} \frac{1}{m} \sum_{\alpha=1}^m \sigma_\alpha \ell(f(i_\alpha, j_\alpha), R_{i_\alpha j_\alpha})\big],$$

where each σ_α takes values $\{\pm 1\}$ with equal probability. Then with probability at least $1-\delta$, for all $f \in F_\Theta$ we have:

$$R_\ell(f) \le \hat{R}_\ell(f) + 2\mathbb{E}_{\Omega_{obs}}\big[\mathfrak{R}(F_\Theta)\big] + \mathcal{B}\sqrt{\frac{\log\frac{1}{\delta}}{2m}}.$$

4. In other words, we consider the observations to be sampled under a sampling with replacement model which is similar to Recht (2011); Shamir and Shalev-Shwartz (2014). There are also studies that consider other sampling procedures such as Bernoulli model (Candès and Tao, 2009; Candès and Recht, 2012).

Therefore, to guarantee a small enough R_ℓ, not only $\hat{R}_\ell$, but also the Rademacher model complexity $\mathbb{E}_{\Omega_{obs}}\big[\mathfrak{R}(F_\Theta)\big]$ has to be carefully controlled. We further introduce a key lemma to show that the model complexity is related to both the feature quality and the sparse noise level, where better quality of features and lower noise level will lead to a smaller model complexity. The intuition of the goodness of feature quality can be motivated as follows. Consider any imperfect side information which violates (2). One can imagine such a feature set is perturbed by some misleading noise which is not correlated to the true latent space. However, features should still be effective if noise does not weaken the true latent space information too much. Thus, if a large portion of true latent space lies on the informative part of the feature spaces X and Y, they should still be somewhat informative and helpful for recovering the matrix L_0.

More formally, for F_Θ in problem (9), its model complexity $\mathbb{E}_{\Omega_{obs}}\big[\mathfrak{R}(F_\Theta)\big]$ can be bounded in terms of $\mathcal{M}$, $\mathcal{N}$ and $\mathcal{S}$ by the following lemma:

Lemma 3 *Let $\mathcal{X} = \max_i \|\mathbf{x}_i\|_2$, $\mathcal{Y} = \max_i \|\mathbf{y}_i\|_2$, $n = \max(n_1, n_2)$ and $d = \max(d_1, d_2)$. Suppose ℓ is a convex surrogate loss satisfying conditions in Lemma 2 with the Lipschitz constant L_ℓ. Then for F_Θ in problem (9), its model complexity $\mathbb{E}_{\Omega_{obs}}\big[\mathfrak{R}(F_\Theta)\big]$ is upper bounded by:*

$$2L_\ell \mathcal{M}\mathcal{X}\mathcal{Y}\sqrt{\frac{\log 2d}{m}} + \min\left\{2L_\ell\mathcal{N}\sqrt{\frac{\log 2n}{m}}, \sqrt{9CL_\ell\mathcal{B}\frac{\mathcal{N}(\sqrt{n_1}+\sqrt{n_2})}{m}}\right\} + L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}},$$

where L_ℓ and $\mathcal{B}$ are constants appearing in Lemma 2.

Thus, from Lemma 2 and 3, one should carefully construct a feasible solution set (by setting $\mathcal{M}$, $\mathcal{N}$ and $\mathcal{S}$) such that both $\hat{R}_\ell(f^*)$ and $\mathbb{E}_{\Omega_{obs}}\big[\mathfrak{R}(F_\Theta)\big]$ are controlled to be reasonably small. We now suggest a witness setting of $(\mathcal{M},\mathcal{N},\mathcal{S})$ as follows. Let $\mathcal{T}_\mu(\cdot) : \mathbb{R}^+ \to \mathbb{R}^+$ be the thresholding operator where $\mathcal{T}_\mu(x) = x$ if $x \geq \mu$ and $\mathcal{T}_\mu(x) = 0$ otherwise. In addition, let $X = \sum_{i=1}^{d_1}\sigma_i\mathbf{u}_i\mathbf{v}_i^T$ be the reduced SVD of X, and $X_\mu = \sum_{i=1}^{d_1}\sigma_1\mathcal{T}_\mu(\sigma_i/\sigma_1)\mathbf{u}_i\mathbf{v}_i^T$ be the "μ-informative" part of X. The ν-informative part of Y, denoted as Y_ν, can also be defined similarly. We then propose to set:

$$\mathcal{M} = \|\hat{M}\|_* \qquad \mathcal{N} = \|L_0 - X_\mu\hat{M}Y_\nu^T\|_* \qquad \mathcal{S} = \|S_0\|_1, \tag{10}$$

where $\hat{M} := \arg\min_M \|X_\mu M Y_\nu^T - L_0\|_F^2 = (X_\mu^TX_\mu)^\dagger X_\mu^T L_0 Y_\nu (Y_\nu^TY_\nu)^\dagger$ is the optimal solution for approximating L_0 under the informative feature spaces X_μ and Y_ν. The following lemma further shows that the trace norm of $\hat{M}$ will not grow as a function of n.

Lemma 4 *Fix $\mu, \nu \in (0, 1]$, and let γ be a constant defined by*

$$\gamma := \min\left(\frac{\min_i\|\mathbf{x}_i\|}{\mathcal{X}}, \frac{\min_i\|\mathbf{y}_i\|}{\mathcal{Y}}\right)$$

where $\mathcal{X}, \mathcal{Y}$ are constants defined in Lemma 3. Then the trace norm of $\hat{M}$ is upper bounded by:

$$\|\hat{M}\|_* \leq \frac{C_L d^2}{\mu^2\nu^2\gamma^2\mathcal{X}\mathcal{Y}},$$

where $C_L \geq \max_{i,j}|\mathbf{e}_i^T L_0 \mathbf{e}_j|$ is the constant upper bounding the entries of L_0.

Therefore, by combining Lemma 2-4, we derive a generalization error bound on $R_\ell(f^*)$ of problem (9) as follows.

Theorem 5 *Suppose ℓ is a convex surrogate loss function with Lipschitz constant L_ℓ bounded by $\mathcal{B}$ with respect to its first argument and assume that $\ell(t,t) = 0$. Consider problem (9) where the constraints $(\mathcal{M}, \mathcal{N}, \mathcal{S})$ are set as (10) with some fixed $\mu, \nu \in (0, 1]$. Then with probability at least $1 - \delta$, the expected ℓ-risk of the optimal solution $R_\ell(f^*)$ is bounded by:*

$$R_\ell(f^*) \leq \min\left\{4L_\ell\mathcal{N}\sqrt{\frac{\log 2n}{m}}, \sqrt{36CL_\ell\mathcal{B}\frac{\mathcal{N}(\sqrt{n_1}+\sqrt{n_2})}{m}}\right\} + 2L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}} + \frac{4L_\ell C_L d^2}{\mu^2\nu^2\gamma^2}\sqrt{\frac{\log 2d}{m}} + \mathcal{B}\sqrt{\frac{\log\frac{1}{\delta}}{2m}},$$

where C, C_L and γ are constants appearing in Lemma 3 and 4.

As a result, Theorem 5 leads us to deem $\mathcal{N}$ and $\mathcal{S}$ in (10) to be the measurement of feature quality and noise level respectively, where features with better quality (or observations with less corruption) lead to a smaller $\mathcal{N}$ (or $\mathcal{S}$) and thus a smaller risk quantity. Note that the measurement $\mathcal{N}$ is consistent with the stated intuition of feature quality, since given a good feature set such that most true latent space of L_0 lies on the informative part of the feature spaces, $X_\mu\hat{M}Y_\nu^T$ will absorb most of L_0, resulting in a small $\mathcal{N}$. Given Theorem 5, we can further discuss the effect of side information in the proposed model (9) on the sample complexity in several important scenarios. To make the comparison more clear, we fix $d = O(1)$ so the feature dimensions do not grow as a function of n in the following discussion.

3.2. Sample Complexity for Matrix Completion

First, consider the matrix completion case where the observations are partial yet not corrupted, i.e. $S_0 = 0$. Then, as mentioned, our model can be further reduced to IMCNF (problem (6), or equivalently problem (9) with $\mathcal{S} = 0$) which exploits noisy side information to solve the matrix completion problem. In addition, from Theorem 5, we can derive the sample complexity of IMCNF as follows.

Corollary 6 *Suppose we aim to (approximately) recover L_0 from partial observations $R = \mathcal{P}_{\Omega_{obs}}(L_0)$ in the sense that $\mathbb{E}_{(i,j)\sim\mathcal{D}}\left[\ell((XM^*Y^T+N^*)_{ij}, \mathbf{e}_i^T L_0 \mathbf{e}_j)\right] < \epsilon$ given an arbitrary $\epsilon > 0$. Then by solving problem (9) with constraints to be set as (10), $O(\min(\mathcal{N}\sqrt{n}, \mathcal{N}^2\log n)/\epsilon^2)$ samples are sufficient to guarantee that the estimated low-rank matrix $XM^*Y^T+N^*$ recovers L_0 with high probability, provided a sufficiently large n.*

Corollary 6 suggests that the sample complexity of IMCNF can be lowered with the aid of (sufficiently informative) noisy side information. The significance of this result can be further explained by comparing with the sample complexity of other models. First, if features are perfect ($\mathcal{N} = O(1)$), Corollary 6 suggests that our IMCNF model only requires $O(\log n)$ samples for recovery. This result coincides with the sample complexity of IMC, in which researchers have shown that given perfect features, $O(\log n)$ observations are enough

for exact recovery (Xu et al., 2013; Zhong et al., 2015). However, IMC does not guarantee recovery when features are not perfect, while Corollary 6 suggests that recovery is still attainable by IMCNF with $O(\min(\mathcal{N}\sqrt{n}, \mathcal{N}^2 \log n)/\epsilon^2)$ samples.

On the other hand, our analysis suggests that sample complexity of IMCNF is at most $O(n^{3/2})$ given any features by applying the following inequality to Corollary 6:

$$\mathcal{N} \leq \|L_0\|_* \leq C_L \|E\|_* \leq C_L \sqrt{\text{rank}(E)} \|E\|_F = C_L \sqrt{n_1 n_2} = O(n),$$

where $E \in \mathbb{R}^{n_1 \times n_2}$ is the matrix with all entries to be one. To explain the result, we compare this result to the sample complexity of standard matrix completion where no side information is considered. At the first glance, it may appear that the result is worse than pure matrix completion in the worst case, since many well-known matrix completion guarantees showed that under certain spikiness and distributional conditions, one can achieve $O(n \operatorname{polylog} n)$ sample complexity for both approximate recovery (Srebro and Shraibman, 2005; Negahban and Wainwright, 2012) and exact recovery (Candès and Recht, 2012). However, all of the above $O(n \operatorname{polylog} n)$ results require additional distributional assumptions on observed entries, while our analysis does not make distributional assumptions. To make a fairer comparison, Shamir and Shalev-Shwartz (2014) have shown that for pure matrix completion, $O(n^{3/2})$ entries are sufficient for approximate recovery without any distributional assumptions, and furthermore, the bound is *tight* if no further distributional assumptions on observed entries is allowed. Therefore, Corollary 6 indicates that IMCNF is at least as good as pure matrix completion even *in the worst case* under the distribution-free setting. Notice that it is reasonable to meet the matrix completion lower bound $\Omega(n^{3/2})$ even given features, since for completely useless feature case (e.g. X, Y are random matrices), the given information is exactly the same as that in standard matrix completion, so any method cannot beat the matrix completion lower bound even by taking features into account.

However, in most applications, the given features are expected to be far from random, and Corollary 6 provides a theoretical insight to show that even noisy features can be useful in matrix completion. Indeed, as long as features are informative enough such that $\mathcal{N} = o(n)$, sample complexity of the IMCNF model will be asymptotically lower than standard matrix completion. Here we provide a concrete example for such a scenario. We consider the rank-r matrix L_0 to be generated from random orthogonal model (Candès and Recht, 2012) as follows:

Corollary 7 *Let $L_0 \in \mathbb{R}^{n \times n}$ be generated from random orthogonal model, where $U = \{\mathbf{u}_i\}_{i=1}^r$, $V = \{\mathbf{v}_i\}_{i=1}^r$ are random orthogonal bases, and $\sigma_1 \dots \sigma_r$ are singular values with arbitrary magnitude. Let σ_t be the largest singular value such that $\lim_{n\to\infty} \sigma_t/\sqrt{n} = 0$. Then, given the noisy features X, Y where $X_{:i} = \mathbf{u}_i$ (and $Y_{:i} = \mathbf{v}_i$) if $i < t$ and $X_{:i}$ (and $V_{:i}$) be any basis orthogonal to U (and V) if $i \geq t$, $o(n)$ samples are sufficient for IMCNF to achieve recovery of L_0.*

Corollary 7 suggests that, under random orthogonal model, if features are not too noisy in the sense that noise only perturbs the true subspace associated with smaller singular values, the sample complexity of IMCNF can be asymptotically lower than the lower bound of standard matrix completion (which is $\Omega(n^{3/2})$).

All in all, for the matrix completion case where observations are partial yet uncorrupted, our proposed problem (4) reduces to the IMCNF model (6) and moreover, Corollary 6

suggests that it can attain recovery more efficiently than other existing models by exploiting noisy yet informative side information.

3.3. Sample Complexity given Partial and Corrupted Observations

We now further consider the case where observations are *both* missing and corrupted. In the presence of corruption, Theorem 5 results in the following Corollary 8 which shows that the learned matrix $XM^*Y^T + N^* + S^*$ will be close to $L_0 + S_0$ with sufficient observations, where the number of required samples depends on both the quality of features and the noise level of sparse error. Since there always exists a solution of problem (9) with $\mathcal{P}_{\Omega_{obs}^{\perp}}(S^*) = 0$ and the generalization bound in Theorem 5 holds for any solution, the result in Corollary 8 implies that $XM^*Y^T + N^*$ is close to L_0 on *missing entries* $(i, j) \notin \Omega_{obs}$, which means we can recover the missing entries of the underlying low-rank matrix with small error. Moreover, if we apply the proposed Algorithm 1 to solve the soft-constrained form (4), the solution S^* will satisfy $\mathcal{P}_{\Omega_{obs}^{\perp}}(S^*) = 0$ automatically. In the following, we formally state the recovery guarantee for partial and corrupted observations:

Corollary 8 *Suppose we are given a data matrix $R = \mathcal{P}_{\Omega_{obs}}(L_0 + S_0)$ containing both missing and corrupted observations of L_0 along with side information X, Y. Then for problem* (9) *with constraints to be set as* (10)*, if we apply Algorithm 1 to solve its equivalent form in* (4)*, $O(\{\min(\mathcal{N}\sqrt{n}, \mathcal{N}^2 \log n) + \mathcal{S}^2 \log n\}/\epsilon^2)$ samples are sufficient to guarantee that with high probability, $\mathbb{E}_{(i,j)\sim\mathcal{D}}\big[\ell((XM^*Y^T + N^* + S^*)_{ij}, R_{ij})\big] < \epsilon$ for any $\epsilon > 0$ provided a sufficiently large n, where S^* satisfies $\mathcal{P}_{\Omega_{obs}^{\perp}}(S^*) = 0$.*

Corollary 8 suggests that if observations are both missing and corrupted, then to guarantee the learned low-rank matrix $XM^*Y^T + N^*$ is accurate on missing entries, the number of required samples depends not only on the quality of features $\mathcal{N}$, but also on the noise level of corruption $\mathcal{S}$. In addition, larger $\mathcal{S}$ results in a higher complexity guarantee. The reasoning behind this result is intuitive: compared to the matrix completion setting in Section 3.2, allowing observed samples to be corrupted makes the problem harder, and therefore may increase sample complexity. However, suppose the corruption is not too severe as the total magnitude of error $\mathcal{S}$ is in the order of $o(n/\sqrt{\log n})$, Corollary 8 still provides a non-trivial bound on required samples for learning the missing entries accurately. Furthermore, better quality of features becomes helpful for faster learning if corruption is small enough. For example, suppose the allowed corruption budget is upper bounded as $\mathcal{S} = O(1)$, then the sample complexity will again be $O(\min(\mathcal{N}\sqrt{n}, \mathcal{N}^2 \log n)/\epsilon^2)$. As discussed, it implies that the number of samples can be $o(n^{3/2})$ provided sufficiently good features, while the required samples will be $O(n^{3/2})$ if no features are given.

Remark. Overall, we provide sample complexity analysis to justify that our model (4) is able to learn the missing information of L_0 more effectively by leveraging side information. The analysis is based on the generalization bounds of the missing values, where more informative side information (and less corruption) results in fewer required samples for accurate estimation, justifying the usefulness of side information.

Again, we emphasize that our results are relatively loose compared to those exact recovery guarantees in both matrix completion (Candès and Tao, 2009; Candès and Recht,

2012) and robust PCA (Chandrasekaran et al., 2011; Candès et al., 2011) as we only consider an approximate recovery on missing entries. However, it is important to note that those stronger recovery guarantees require additional assumptions, such as incoherence of the underlying low-rank matrices and distributional assumptions, to ensure the sampled observations are sufficiently representative. On the other hand, our analysis does not require distributional or incoherence assumptions, since in generalization analysis we only need to ensure the average loss of the missing entries are sufficiently small, and therefore, the average loss can still be controlled even if few spots are wrongly estimated in a high incoherence L_0.

However, in some circumstances, it is in fact possible to provide a stronger argument to justify the usefulness of side information in the exact recovery context. For example, in the robust PCA setting where observations are grossly corrupted yet full, one can further show that by exploiting perfect side information, a large amount of low-rank matrices L_0, which cannot be recovered by standard robust PCA without features, can be exactly recovered using our proposed model. Interested readers can consult Chiang et al. (2016) for such a result in detail. A theoretical analysis on how much the side information can improve the exact recovery guarantees in general low-rank matrix learning would be an interesting research direction to explore in the future.

4. Experimental Results

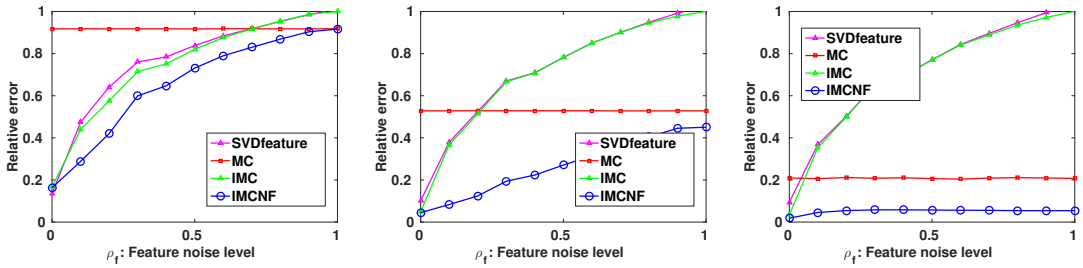
We now present experimental results on exploiting side information using the proposed model (4) for various low-rank matrix learning problems. For synthetic experiments, we show that our model performs better with the aid of side information given observations are either only missing (i.e. matrix completion setting), only corrupted (i.e. robust PCA setting) or both missing and corrupted. For real-world applications, we consider three machine learning applications—relationship prediction, semi-supervised clustering and noisy image classification—and show that each of them can be viewed as a problem of learning a low-rank modeling matrix from missing/corrupted entries with side information. As a result, by applying our model, we can achieve better performance compared to other state-of-the-art methods in these applications.

4.1. Synthetic Experiments

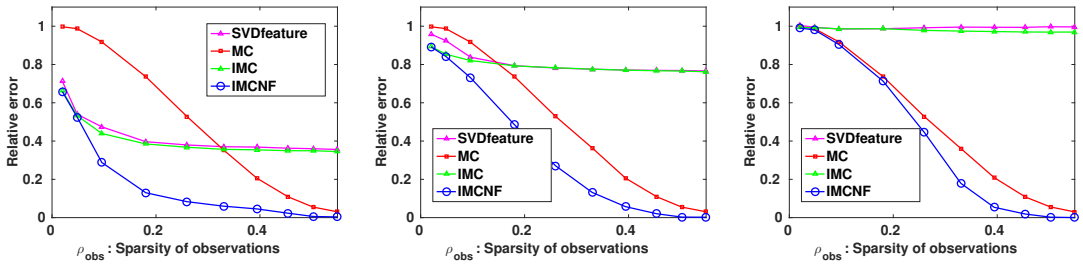
To begin with, we show the usefulness of (both perfect and noisy) side information in our model under different synthetic settings.

4.1.1. Experiments on Matrix Completion Setting

We first examine the effect of side information in our model in the case of matrix completion. We create a low rank matrix $L_0 = UV^T$ where the true latent row/column space $U, V \in \mathbb{R}^{200\times 20}$, $U_{ij}, V_{ij} \sim \mathcal{N}(0,1)$. We then randomly sample ρ_{obs} of entries Ω_{obs} from L_0 to form the observed matrix $R = \mathcal{P}_{\Omega_{obs}}(L_0)$. In addition, we construct perfect side information $X^*, Y^* \in \mathbb{R}^{200\times 40}$ satisfying (2), from which we generate different quality of features X, Y with a noise parameter $\rho_f \in [0,1]$, where X and Y are derived by replacing ρ_f of bases



(a) $\rho_{obs} = 0.1$ (b) $\rho_{obs} = 0.25$ (c) $\rho_{obs} = 0.4$



(d) $\rho_f = 0.1$ (e) $\rho_f = 0.5$ (f) $\rho_f = 0.9$

Figure 1: Performance of various methods for matrix completion under certain fixed sparsity of observations ρ_{obs} (upper figures) and fixed feature quality ρ_f (lower figures). We observe that all feature-based methods perform better than standard matrix completion (MC) given perfect features ($\rho_f = 0$). However, IMCNF is less sensitive to feature noise as ρ_f increases, indicating that it better exploits information from noisy features.

in X^* (and Y^*) with bases orthogonal to X^* (and Y^*). We then consider recovering the underlying matrix L_0 given R, X and Y.

In this experiment, we consider the proposed IMCNF model (problem 6) which is an instance of the general problem (4) for exploiting noisy side information in matrix completion case. We compare IMCNF with standard trace-norm regularized matrix completion (MC), IMC (Jain and Dhillon, 2013) and SVDfeature (Chen et al., 2012). The recovered matrix L^* from each algorithm is evaluated by the standard relative error:

$$\frac{\|L^* - L_0\|_F}{\|L_0\|_F}. \tag{11}$$

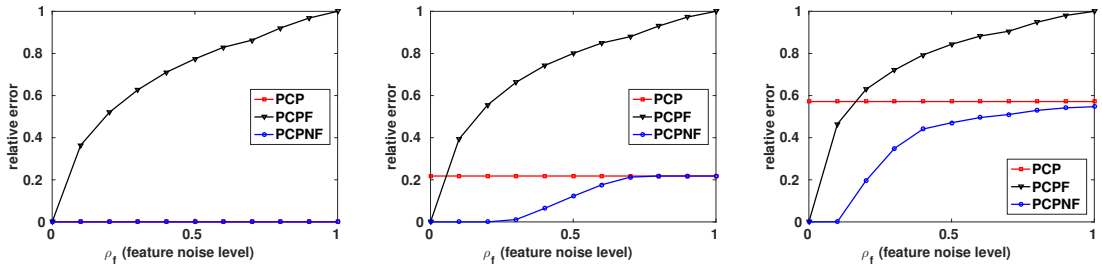
For each method, we select parameters from the set $\{10^\alpha\}_{\alpha=-3}^{2}$ and report the one with the best recovery. All results are averaged over 5 random trials.

Figure 1 shows results of each method under different $\rho_{obs} = 0.1, 0.25, 0.4$ and $\rho_f = 0.1, 0.5, 0.9$. We can first observe in upper figures that IMC and SVDfeature perform similarly under each ρ_{obs}, and moreover, their performance mainly depends on feature quality and will not be affected much by the number of observations. Although their performance is comparable to IMCNF given perfect features ($\rho_f = 0$), their performance quickly drops when features become noisy. This phenomenon is more clear in figure 1c and 1f where we see that given noisy features, IMC and SVDfeature will be easily trapped by feature noise and perform even worse than pure MC. Another interesting finding is that even if feature quality is as good as $\rho_f = 0.1$ (Figure 1d), IMC (and SVDfeature) still fails to achieve 0 relative error as the number of observations increases, suggesting that IMC is sensitive to feature noise and cannot guarantee recoverability when features are not perfect. On the other hand, we see that performance of IMCNF can be improved by both better features and more observations. In particular, it makes use of informative features to achieve lower error compared to MC and is also less sensitive to feature noise compared to IMC and SVDfeature. These results empirically support the analysis presented in Section 3.

4.1.2. Experiments on Robust PCA Setting

In this experiment, we examine the effect of both perfect and noisy side information in the proposed model for robust PCA as follows. We create a low-rank matrix $L_0 = UV^T$, where $U, V \in \mathbb{R}^{n\times 40}$, $U_{ij}, V_{ij} \sim N(0, 1/n)$ with $n = 200$. We also form a sparse noise matrix S_0 where each entry will be a non-zero entry with probability ρ_s, and each non-zero entry will take values from $\{\pm 1\}$ with equal probability. We then construct noisy features $X, Y \in \mathbb{R}^{n\times 50}$ with a noise parameter ρ_f using the same construction in the previous experiment, i.e. features X/Y will only span $40 \times (1 - \rho_f)$ true bases of U/V. We then consider to recover the low-rank matrix given the fully observed matrix $R = L_0 + S_0$ along with noisy side information X and Y.

We consider the following three methods: PCP (Candès et al., 2011) which does not exploit features, PCPF (problem 8) which theoretically exploits perfect features using bilinear embedding, and PCPNF (problem 7) for incorporating noisy side information. Note that PCPF and PCPNF are instances of our proposed model (4) that exploits side information for robust PCA problem as discussed in Section 2.3. The same relative error criterion (11) is used for evaluation.



(a) $\rho_s = 0.1$ (b) $\rho_s = 0.2$ (c) $\rho_s = 0.3$

Figure 2: Performance of various methods for robust PCA given different feature noise level ρ_f and sparsity of corruption ρ_s. These results show that PCPNF can make use of noisy yet informative features for better recovery.

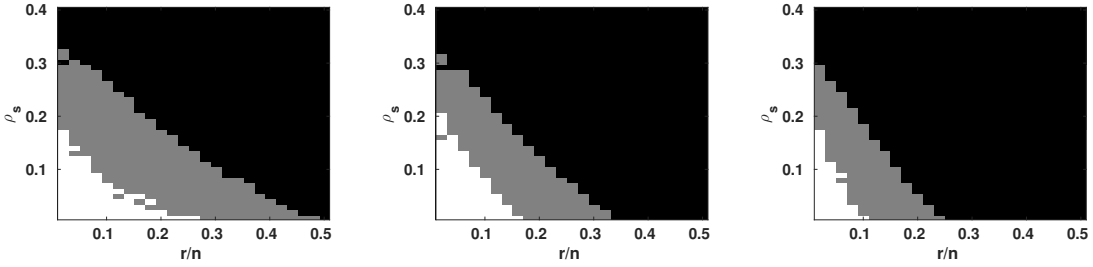
Figure 2 shows the performance of each method given different feature quality under $\rho_s = 0.1, 0.2, 0.3$. We first see that when features are perfect ($\rho_f = 0$), both PCPF and PCPNF can exactly recover the underlying matrix, while pure PCP fails to recover L_0 if $\rho_s \geq 0.2$. This result confirms that both PCPNF and PCPF can leverage perfect features for better recovery. However, as features become noisy (larger ρ_f), we see that PCPF quickly performs worse as it is misled by noise in features, while PCPNF can better exploit noisy features for recovery. In particular, in Figure 2b, we observe that PCPNF still recovers L_0 given noisy yet reasonably good features ($0 < \rho_f < 0.4$), whereas PCP and PCPF fail to recover L_0. These results show that PCPNF can take advantage of noisy side information for learning L_0 given corrupted observations.

4.1.3. Experiments on Learning with Missing and Corrupted Observations

We now further examine to what extent can side information help the learning using our model when observations are both missing and corrupted. We consider the same construction of L_0 and S_0 as in the previous experiment, and generate perfect feature matrices $X, Y \in \mathbb{R}^{n \times d}$ with $d = r + 10$. We then form the observation set Ω_{obs} by randomly sampling ρ_{obs} of entries from all n^2 indexes, and take $R = \mathcal{P}_{\Omega_{obs}}(L_0 + S_0)$ as the observed matrix. The goal is therefore to recover L_0 given R along with side information X and Y.

To exploit the advantage of side information, we consider the proposed model in form (5) where we further set $\alpha = 1$ and $\beta = \infty$ to force N^* to be zero for better exploiting perfect features, and compare it with the problem (1) which tries to recover L_0 only using structural information. Notice that when $\rho_{obs} = 1.0$, the given problem becomes a robust PCA problem where R is a fully observed matrix, in which case problem (1) reduces to PCP method and our model reduces to PCPF objective (problem 8), respectively. From this aspect, we refer to problem (1) as "PCP with partial observations" (PCP-part) and our model as "PCPF with partial observations" (PCPF-part). The relative error criterion (11) is again used to evaluate the recovered matrix. Here, we regard the recovery to be successful if the error is less than 10^{-4}. The parameter λ in both methods are set to be $1/\sqrt{\rho_{obs} n}$.

We compare the recoverability of PCP-part and PCPF-part by varying rank of L_0 (r) and sparsity of S_0 (ρ_s) under different $\rho_{obs} = 1.0, 0.7$ and 0.5. For each pair of (r, ρ_s),



(a) $\rho_{obs} = 1.0$ (b) $\rho_{obs} = 0.7$ (c) $\rho_{obs} = 0.5$

Figure 3: Performance of PCP-part and PCPF-part with perfect features for recovering L_0 from missing and corrupted observations (controlled by ρ_{obs} and ρ_s respectively). Both methods achieve recovery in white region and fail in black region, yet there is a gray region where only PCPF-part achieves recovery. This shows that by leveraging perfect features, PCPF-part can recover a much larger class of L_0 given both missing and corrupted observations are present.

we apply both methods to obtain the estimated low-rank matrix L^*. We then mark the grid point (r, ρ_s) to be white if recovery is attained by both methods and black if both fail. We also observe that in several cases recovery cannot be attained by PCP-part but can be attained by PCPF-part, and these grid points are marked as gray. The results are shown in Figure 3. We observe that for each ρ_{obs}, there exists a substantial gray region where matrices in such a region can be recovered only by PCPF-part. This result shows that in the case where both missing and corrupted entries are present, by exploiting side information, the proposed model is able to further recover a large amount of matrices which cannot be recovered if no side information is provided.

4.2. Real-world Applications

We now consider three applications—relationship prediction in signed networks, semi-supervised clustering and noisy image classification—which can be cast to problems of low-rank matrix learning from missing/corrupted entries with additional side information. As a consequence, we show that by learning the low-rank modeling matrix using our proposed model, we can achieve better performance compared to other methods for these applications as our model can better exploit side information in learning.

4.2.1. Relationship Prediction in Signed Networks

We first consider relationship prediction problem in an online review website Epinions (Massa and Avesani, 2006), where people can write product reviews and choose to trust or distrust others based on their reviews. Such a social network can be modeled as a *signed network* where each person is treated as an entity and trust/distrust relationships between people are modeled as positive/negative edges between entities (Leskovec et al., 2010). The relationship prediction problem in signed network is to predict unknown relationship between any two users given the current network snapshot. While several methods

Method	IMCNF	IMC	MF-ALS	HOC-3	HOC-5
Accuracy	**0.9474**±0.0009	0.9139±0.0016	0.9412±0.0011	0.9242±0.0010	0.9297±0.0011
AUC	**0.9506**	0.9109	0.9020	0.9432	0.9480

Table 2: Relationship prediction on Epinions network. We see that given noisy user features, IMC performs worse even than methods without features (MF-ALS and HOCs), while IMCNF outperforms others by successfully exploiting noisy features.

are proposed, a state-of-the-art approach is the low-rank model (Hsieh et al., 2012; Chiang et al., 2014) which first conducts matrix completion on the adjacency matrix and then uses the sign of the completed matrix for prediction. However, these methods are developed based only on network structure. Therefore, if features of users are available, we can also extend the low-rank model by incorporating user features in the completion step.

The experiment setup is described as follows. In this data set, there are about $n = 105$K users and $m = 807$K observed relationship pairs where 15% relationships are distrust. In addition to the who-trust-to-whom information, we are also given a user feature matrix $Z \in \mathbb{R}^{n\times 41}$ where for each user a 41-dimensional feature is collected based on the user's review history, such as number of positive/negative reviews the user gave or received. We consider the following prediction methods: walk and cycle-based methods including HOC-3 and HOC-5 (Chiang et al., 2014), and the original low-rank model with matrix factorization for the completion step (LR-ALS) (Hsieh et al., 2012). These methods make the prediction based on network structure without considering user features. We further consider the extended low-rank model where the completion step is replaced by IMCNF and IMC (Jain et al., 2013), both of which thus incorporate user features implicitly for prediction. Since row and column entities are both users, $X = Y = Z$ is set for both IMCNF and IMC methods. We randomly divide the edges of the network into 10 folds and conduct the experiment using 10-fold cross validation, in which 8 folds are used for training, one fold for validation and the other for testing. Parameters for validation in each method are chosen from the set $\sqcup_{\alpha=-3}^{2}\{10^\alpha, 5\times 10^\alpha\}$.

The averaged accuracy and AUC of each method are reported in Table 2. We first observe that IMC performs worse than LR-ALS even though IMC takes features into account. It is because these user features are only partially related to the relationship matrix, and IMC is misled by such noisy features. On the other hand, IMCNF performs the best among all prediction methods, as it performs slightly better than LR-ALS in terms of accuracy and much better in terms of AUC. This result shows that IMCNF can exploit weakly informative features to make better prediction without being trapped by feature noise.

4.2.2. Semi-supervised Clustering

Semi-supervised clustering is another application which can be translated to learning a low-rank matrix with partial observations. Given a feature matrix $Z \in \mathbb{R}^{n\times d}$ of n items and m pairwise constraints specifying whether item i and j are similar or dissimilar, the goal is to find a clustering of items such that most similar items are within the same cluster.

First, note that the problem can be sub-optimally solved by dropping either constraint or feature information. For example, traditional clustering algorithms (such as k-means) can solve the problem based purely on features of items. On the other hand, one can also

obtain a clustering purely from the pairwise constraints using matrix completion as follows. Let $S \in \mathbb{R}^{n \times n}$ be the (signed) similarity matrix constructed from the constraint set where $S_{ij} = 1$ if item i and j are similar, -1 if dissimilar and 0 if similarity is unknown. Then finding a clustering of n items becomes equivalent to finding a clustering on the signed graph S, where the goal is to put items (denoted as nodes) into k groups so that most edges within the same group are positive and most edges between groups are negative (Chiang et al., 2014). As a result, one can apply a matrix completion approach proposed in Chiang et al. (2014) to solve the signed graph clustering problem, which first conducts matrix completion on S and runs k-means on the top-k eigenvectors of completed S to obtain a clustering of nodes.

Apparently, either dropping features or constraint set is not optimal for semi-supervised clustering problem. Thus, many algorithms are proposed to take both item features and constraints into account, such as metric-learning-based approaches (Davis et al., 2007), spectral kernel learning (Li and Liu, 2009) and MCCC algorithm (Yi et al., 2013). Among many of them, MCCC algorithm is a cutting edge approach which essentially solves semi-supervised clustering using IMC objective. Observing that each pairwise constraint can be viewed as a sampled entry from the matrix $L_0 = UU^T$ where $U \in \mathbb{R}^{n \times k}$ is the clustering membership matrix, MCCC tries to complete L_0 back as ZMZ^T using IMC objective. Furthermore, since the completed matrix is ideally L_0 whose subspace spans U, it thus conducts k-means on the top-k eigenvectors of the completed matrix to obtain a clustering.

However, since MCCC is based on IMC, its performance thus heavily depends on the quality of features. Therefore, we propose to replace IMC with IMCNF in the matrix completion step of MCCC, and then run k-means on the top-k eigenvectors of the completed matrix to obtain a clustering. Both X and Y are again set to be Z as the target low-rank matrix describes the similarity between items. This algorithm can be viewed as an improved version of MCCC to handle noisy features Z.

We now compare our algorithm with k-means, signed graph clustering with matrix completion (Chiang et al., 2014) (SignMC) and MCCC (Yi et al., 2013) on three real-world data sets: Mushrooms, Segment and Covtype. [5] All of them are classification benchmarks where features and ground-truth labels of items are both available, and the ground-truth cluster of each item is defined by its ground-truth label. The statistics of data sets are summarized in Table 3. For each data set, we randomly sample $m = [1, 5, 10, 15, 20, 25, 30] \times n$ clean pairwise constraints and input both constraints and features to each algorithm to obtain a clustering π, where π_i is the cluster index of item i. We then evaluate π using the following pairwise error:

$$\frac{n(n-1)}{2}\left(\sum_{(i,j):\pi_i^*=\pi_j^*} \mathbf{1}\left[\pi_i \neq \pi_j\right] + \sum_{(i,j):\pi_i^*\neq\pi_j^*} \mathbf{1}\left[\pi_i = \pi_j\right]\right)$$

where π_i^* is the ground-truth cluster of item i.

Figure 4 shows the clustering result of each method given various numbers of constraints on each data set. We first see that for the Mushrooms data set where features are perfect (100% training accuracy can be attained by a linear-SVM for classification), both MCCC

5. All data sets are available at `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`. For Covtype, we subsample from the entire data set to make each cluster has balanced size.

	number of items n	feature dimension d	number of clusters k
Mushrooms	8124	112	2
Segment	2319	19	7
Covtype	11455	54	7

Table 3: Statistics of semi-supervised clustering data sets.

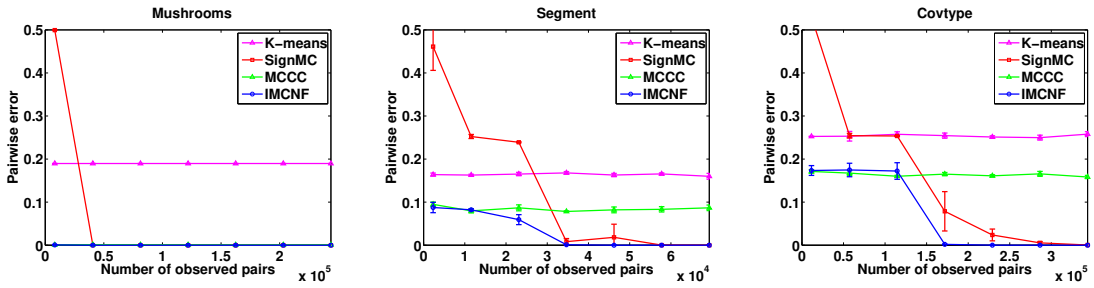


Figure 4: Performance of various semi-supervised clustering methods on real-world data sets. For the Mushrooms data set where features are perfect, both MCCC and IMCNF can output the ground-truth clustering with 0 error rate. For Segment and Covtype where features are more noisy, IMCNF model outperforms MCCC as its error decreases given more constraints.

and IMCNF can obtain the ground-truth clustering with 0 error rate, which indicates that MCCC (and IMC) is indeed effective with perfect features. For the Segment and Covtype data sets, we observe that the performance of k-means and MCCC is dominated by feature quality. Although MCCC is still benefited from constraint information as it outperforms k-means, it clearly does not make the best use of constraints since its performance is not improved even if number of constraints increases. On the other hand, the error rate of SignMC can always decrease down to 0 by increasing m; however, since it disregards features, it suffers from a much higher error rate than other methods when constraints are few. Finally, we see that IMCNF combines advantages from both MCCC and SignMC, as it not only makes use of features when few constraints are observed, but also leverages constraint information to avoid being trapped by feature noise. Therefore, the experiment shows that by carefully handling side information using IMCNF model, we can further improve the state-of-the-art semi-supervised clustering algorithm.

4.2.3. Noisy Image Classification

Finally, we consider noisy image classification problem as an application of low-rank matrix learning with corrupted observations. In this problem, we are given a set of correlated images in which a few of pixels are corrupted, and the task is to denoise the images so that one can classify the images correctly. Since the underlying clean images are correlated and thus share an implicit low-rank structure, standard robust PCA could be used to identify sparse noise and recover the (low-rank approximation of) clean images. However, in certain cases, low-dimensional features of images may also be available from other sources. For example, suppose the set of images are human faces, then the principal components of

linear SVM classifiers					kernel SVM classifiers				
ρ_s	Clean	Noisy	PCP	PCPF	ρ_s	Clean	Noisy	PCP	PCPF
0.1		59.63	86.33	**87.88**	0.1		18.47	94.85	**95.89**
0.2	91.96	38.16	85.94	**87.48**	0.2	98.33	10.32	94.55	**95.48**
0.3		25.63	78.52	**79.84**	0.3		10.32	87.00	**87.78**

Table 4: Digit classification accuracy of PCP and PCPF with Eigendigit features. The column Clean shows the accuracy on L_0 and the column Noisy shows the accuracy on R. Denoised images from both PCP and PCPF achieve much higher accuracy than noisy images, and PCPF further outperforms PCP by incorporating Eigendigit features.

general human faces—known as Eigenface (Turk and Pentland, 1991)—could be used as features, and such features could be helpful in the denoising process.

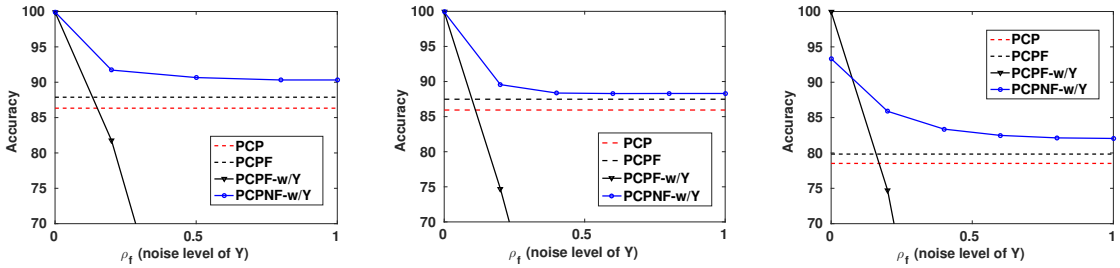
Motivated by the above realization, here we consider multiclass classification on a set of noisy images from the MNIST data set. The data set includes $50,000$ training images and $10,000$ testing images, and each image is a 28×28 handwritten digit represented as a 784-dimensional vector. We first pre-train both multiclass linear and kernel SVM classifiers on the clean training images, and perturb the testing image set to generate noisy images R. Precisely, let $L_0 \in \mathbb{R}^{784\times 10000}$ be the set of (clean) testing images, where each row denotes a pixel and each column denotes an image. We then construct a sparse noise matrix $S_0 \in \mathbb{R}^{784\times 10000}$ where ρ_s of entries are randomly picked to be corrupted by setting their values to be 255. The observed noisy images is thus given by $R = \min(L_0 + S_0, 255)$. In the following, we show that by exploiting features of row and column entities in this problem, we can better denoise the noisy images for classification.

Exploiting Eigendigit Features. We first exploit "Eigendigit" features to help denoising. We take the training image set to produce the Eigendigit features $X \in \mathbb{R}^{784\times 300}$ using PCA and simply set $Y = I$ as we do not consider any column features here. We then input R into PCP to derive a set of denoised images L^*_{pcp} and input R, X and Y (which is I) into PCPF (problem 8) to derive another set of denoised images $L^*_{pcpf} = XM^*$. Both L^*_{pcp} and L^*_{pcpf} will be low rank approximations of the clean images. Note that although the Eigendigit features X will not satisfy (2) which is assumed in the derivation of PCPF, we could heuristically incorporate it using PCPF in this circumstance because X is still expected to contain unbiased information of the low-rank approximation of the clean digits. [6]

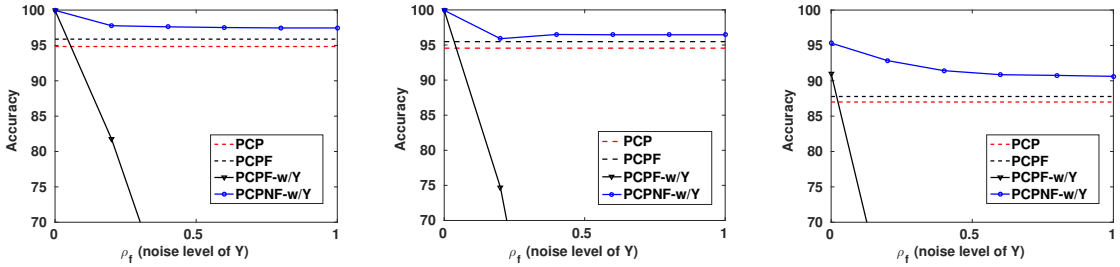
To compare the quality of denoised images of PCP and PCPF, we input L^*_{pcp} and L^*_{pcpf} to pre-trained SVMs for digit classification and report the results in Table 4. Both methods are somehow effective for denoising sparse noise, since accuracies achieved by the denoised images are much closer to the clean images compared to the noisy images. Furthermore, PCPF consistently achieves better accuracies than PCP under different ρ_s, showing that incorporating Eigendigit features using PCPF is helpful on denoising process for classification.

Exploiting both Eigendigit and Label-relevant Features In addition to the Eigendigit features X, now we further exploit features for column entities. Ideally, the

6. Rigorously speaking, the ground-truth L_0 is not even low-rank, but only approximately low-rank.



(a) Linear SVM, $\rho_s = 0.1$ (b) Linear SVM, $\rho_s = 0.2$ (c) Linear SVM, $\rho_s = 0.3$



(d) Kernel SVM, $\rho_s = 0.1$ (e) Kernel SVM, $\rho_s = 0.2$ (f) Kernel SVM, $\rho_s = 0.3$

Figure 5: Digit classification accuracy of various methods with both Eigendigit and label-relevant features. For each ρ_s, we construct the label-relevant features Y with different quality by varying ρ_f. The results show that PCPNF-w/Y is able to better exploit noisy label-relevant features Y.

column features Y may describe the relevant information between images, which could be extremely useful for classification. Thus, we generate the "label-relevant" features Y for column entities as follows. Let $Y^* \in \mathbb{R}^{10000\times 10}$ be a perfect column feature matrix where the i-th column of Y^* is the indicator vector of digit $i-1$ (so Y^* contains ground-truth label information). We then randomly pick ρ_f of rows in Y^* and shuffle these rows to form $\tilde{Y}$, which correspondingly means that $10,000 \times \rho_f$ images have noisy relevant information in $\tilde{Y}$. Finally, we form the column feature $Y \in \mathbb{R}^{10000\times 50}$ which spans $\tilde{Y}$. Thus, the quality of Y depends on the parameter $\rho_f \in [0,1]$ and smaller ρ_f results in better label-relevant features.

We consider four approaches for denoising in the following experiment. The first two baseline methods are PCP and PCPF with only Eigendigit features X. Both methods are the ones we considered in the previous experiment which do not take label-relevant features into account. Moreover, we consider using PCPF and PCPNF to incorporate *both* the Eigendigit features X and the label-relevant features Y for denoising, and we name them as "PCPF-w/Y" and "PCPNF-w/Y" to emphasize that they embed the label-relevant features Y. We apply each method to denoise noisy images under different ρ_f and ρ_s and examine the quality of denoised images by testing the accuracies they achieve in pre-trained SVMs.

The results are shown in Figure 5. In each figure, we fix the sparsity of noise ρ_s and try to recover the clean images using each method with different quality of Y. We can see that the perfect label-relevant features are extremely useful, as when $\rho_f = 0$, recovered images

from both PCPF-w/Y and PCPNF-w/Y achieve even much higher accuracies compared to the clean images (reported in Table 4). However, once ρ_f increases, PCPF-w/Y quickly fails as its accuracy drops drastically (accuracies become much lower than 70 for $\rho_f > 0.5$ and thus are not shown in figures). On the other hand, we see that PCPNF-w/Y performs much better than PCPF-w/Y on exploiting noisy label-relevant features, as it still achieves better accuracies compared to both PCPF and PCP when $\rho_f > 0$. The results again demonstrate the effectiveness of our proposed model on exploiting noisy side information.

5. Related Work

Learning a low-rank matrix from imperfect observations is an expansive domain in machine learning including many fundamental problems, such as Principal Component Analysis (PCA) (Hotelling, 1933), matrix completion (Candès and Tao, 2009), low-rank matrix sensing (Zhong et al., 2015) and robust PCA (Wright et al., 2009). While each of the above topics is an independent research area burgeoning in recent years, our main focus is to study the usefulness of side information in low-rank matrix learning where the observations are partial and/or corrupted in both theoretical and practical aspects.

Learning a low-rank matrix from partial observations is well-known as matrix completion problem, which has been successfully applied to many machine learning tasks including recommender systems (Koren et al., 2009), social network analysis (Hsieh et al., 2012; Chiang et al., 2014) and clustering (Chen et al., 2014). Several theoretical foundations have also been established. One of the most striking results is the exact recovery guarantee provided by Candès and Tao (2009) and Candès and Recht (2012) where the authors showed that $O(n \operatorname{polylog} n)$ observations are sufficient for exact recovery with high probability, provided that entries are uniformly sampled at random. Several works also study recovery under non-uniform distributional assumptions (Negahban and Wainwright, 2012), distribution-free settings (Shamir and Shalev-Shwartz, 2014) and noisy observations (Keshavan et al., 2010; Candès and Plan, 2010).

A few research papers also consider side information in the matrix completion setting (Menon and Elkan, 2011; Chen et al., 2012; Natarajan and Dhillon, 2014; Shin et al., 2015). Although most of them found that features are helpful in certain applications (Menon and Elkan, 2011; Shin et al., 2015) and in the cold-start setting (Natarajan and Dhillon, 2014), they mainly focus on the non-convex matrix factorization formulation without any theoretical analysis on the effect of side information. More recently, Jain et al. (2013) studied Inductive Matrix Completion (IMC) objective to incorporate side information, and several follow-up works also consider IMC with trace norm regularization (Xu et al., 2013; Zhong et al., 2015). All of them showed that recovery can be achieved by IMC with much lower sample complexity provided perfect features. However, as we have discussed in the paper, given imperfect features, IMC cannot recover the underlying matrix and may even suffer from poor performance in practice. This observation leads us to further develop an improved model which better exploits noisy side information in learning (see Section 2.3).

Robust PCA is another prominent instance of low-rank matrix learning from imperfect observations, where the goal is to recover a low-rank matrix from a full matrix in which a few of entries are arbitrarily corrupted by sparse noise. This sparse structure of noise is common in many applications such as image processing and bioinformatics (Wright et al.,

2009). Researchers have also investigated several approaches to robust PCA with theoretical guarantees (Chandrasekaran et al., 2011; Candès et al., 2011). Perhaps the most remarkable milestone is the strong guarantee provided by Candès et al. (2011), in which the authors showed that under mild conditions, low-rank and sparse structure are exactly distinguishable. Several extensions of robust PCA have also been considered, such as robust PCA with column-sparse errors (Xu et al., 2010), with missing data (Candès et al., 2011; Chen et al., 2013) and with compressed data (Ha and Barber, 2015).

However, unlike matrix completion, there is little research that directly exploits side information in the robust PCA problem, leaving the advantage of side information in robust PCA unexplored. Though it may appear that one can extend the analysis of side information in matrix completion to robust PCA as both problems share certain similarities, the robust PCA problem is still essentially different—in fact harder—from matrix completion in many aspects. In particular, matrix completion has been mostly used for *missing value estimation*, where the emphasis is to accurately recover the missing entries given trustable, partial observations, while robust PCA is a *matrix separation problem* where one has to identify the corrupted entries given full yet untrustable observations. This difference naturally precludes a direct extension from the analyses of matrix completion to robust PCA. Nevertheless, Chiang et al. (2016) has recently shown that given perfect features, exact recovery of higher-rank matrices becomes attainable in the robust PCA problem, indicating that side information in robust PCA can be exploited. In this paper, we extend Chiang et al. (2016) and develop a more general model which can further exploit noisy side information to help solve the robust PCA problem.

Another model that shares certain similarities to robust PCA with side information is Low-Rank Representation (LRR), which emerged from the subspace clustering problem (Liu et al., 2010, 2013). Given that the observed data matrix is corrupted by sparse errors, LRR model assumes that the underlying low-rank matrix can be represented by a linear combination of a provided dictionary. Interestingly, LRR can be thought of as a special case of the proposed PCPF model (see Section 2.3) where the given dictionary serves as the row features X. Our problem setting is also more general than LRR as we consider incorporating both row and column features to help recovery.

6. Conclusions

In this paper, we study the effectiveness of side information for low-rank matrix learning from missing and corrupted observations. We propose a general model (problem (4)) which incorporates both perfect and noisy side information by balancing information from features and observations simultaneously, from which we can derive several instances of the model, including IMCNF and PCPNF, that better solve matrix completion and robust PCA by leveraging side information. In addition, we provide a formal analysis to justify the effectiveness of side information in the proposed model, in which we quantify the quality of features and show that the sample complexity of learning can be asymptotically improved given sufficiently informative features, provided a small enough noise level. This analysis therefore quantifies the merits of side information in our model for low-rank matrix learning in theory. Finally, we verify our model in several synthetic experiments as well as in real-world machine learning applications including relationship prediction, semi-supervised

clustering and noisy image classification. By viewing each application as a low-rank matrix learning problem from missing or corrupted observations given certain additional features, we show that employing our model results in competitive algorithms whose performance is comparable to or better than other state-of-the-art approaches. All of our results consistently demonstrate that the proposed model learns the low-rank matrix from missing and corrupted observations more effectively by properly exploiting side information.

Acknowledgments

We would like to acknowledge support for this research from CCF-1320746, IIS-1546452 and CCF-1564000.

Appendix A. Proofs

Preliminary Lemmas

We first introduce two lemmas required in the proof of Lemma 3. These two lemmas provide bounds on the Rademacher complexity of the function class with bounded trace norm and $\ell 1$ norm respectively.

Lemma 9 *Let* $S_w = \{W \in \mathbb{R}^{n\times n} \mid \|W\|_* \leq \mathcal{W}\}$ *and* $\mathcal{A} = \max_i \|A_i\|_2$*, where each* $A_i \in \mathbb{R}^{n\times n}$*, then:*

$$\mathbb{E}_\sigma\big[\sup_{W\in S_w} \frac{1}{m}\sum_{i=1}^{m} \sigma_i \text{trace}(WA_i)\big] \leq 2\mathcal{A}\mathcal{W}\sqrt{\frac{\log 2n}{m}}.$$

Proof This Lemma is directly from the Lemma 3 in Hsieh et al. (2015). ■

Lemma 10 *Let* $S_w = \{W \in \mathbb{R}^{n_1\times n_2} \mid \|W\|_1 \leq \mathcal{W}\}$*, and each* E_i *is in the form of* $E_i = \mathbf{e}_x\mathbf{e}_y^T$*, where* $\mathbf{e}_x \in \mathbb{R}^{n_2}$*,* $\mathbf{e}_y \in \mathbb{R}^{n_1}$ *are two unit vectors. Then:*

$$\mathbb{E}_\sigma\big[\sup_{W\in S_w} \frac{1}{m}\sum_{i=1}^{m} \sigma_i \text{trace}(WE_i)\big] \leq \mathcal{W}\sqrt{\frac{2\log(2n_1n_2)}{m}}.$$

Proof This Lemma is a special case of Theorem 1 in Kakade et al. (2008) with the fact that $\|E_i\|_\infty := \max_{a,b} |(E_i)_{ab}| = 1$. ■

Proof of Lemma 3

Proof First, we can use a standard Rademacher contraction principle (e.g. Lemma 5 in Meir and Zhang, 2003) to bound $\mathfrak{R}(F_\Theta)$ to be:

$$\begin{aligned}\mathfrak{R}(F_\Theta) \leq & L_\ell \mathbb{E}_\sigma\big[\sup_{\theta\in\Theta}\frac{1}{m}\sum_{\sigma=1}^{m}\sigma_\alpha(XMY^T+N+S)_{i_\alpha j_\alpha}\big] \\ = & L_\ell \mathbb{E}_\sigma\big[\sup_{\|M\|_*\leq\mathcal{M}}\frac{1}{m}\sum_{\alpha=1}^{m}\sigma_\alpha \mathrm{trace}(M\mathbf{y}_{j_\alpha}\mathbf{x}_{i_\alpha}^T)\big] + L_\ell \mathbb{E}_\sigma\big[\sup_{\|N\|_*\leq\mathcal{N}}\frac{1}{m}\sum_{\alpha=1}^{m}\sigma_\alpha \mathrm{trace}(N\mathbf{e}_{j_\alpha}\mathbf{e}_{i_\alpha}^T)\big] \\ & + L_\ell \mathbb{E}_\sigma\big[\sup_{\|S\|_1\leq\mathcal{S}}\frac{1}{m}\sum_{\alpha=1}^{m}\sigma_\alpha \mathrm{trace}(S\mathbf{e}_{j_\alpha}\mathbf{e}_{i_\alpha}^T)\big] \\ \leq & 2L_\ell\mathcal{M}\max_{i,j}\|\mathbf{y}_j\mathbf{x}_i^T\|_2\sqrt{\frac{\log 2d}{m}} + 2L_\ell\mathcal{N}\sqrt{\frac{\log 2n}{m}} + L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}}\end{aligned}$$

where the last inequality is derived by applying Lemma 9 and Lemma 10. Moreover, since $\max_{i,j}\|\mathbf{y}_j\mathbf{x}_i^T\|_2 = \max_j\|\mathbf{y}_j\|_2\max_i\|\mathbf{x}_i\|_2$, we can thus upper bound $\mathfrak{R}(F_\Theta)$ by:

$$\mathfrak{R}(F_\Theta) \leq 2L_\ell\mathcal{M}\mathcal{X}\mathcal{Y}\sqrt{\frac{\log 2d}{m}} + 2L_\ell\mathcal{N}\sqrt{\frac{\log 2n}{m}} + L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}}. \tag{12}$$

However, in some circumstances, the above bound (12) is too loose for sample complexity analysis. To deal with these cases, we follow Shamir and Shalev-Shwartz (2014) to derive a tighter bound on the trace norm of residual (i.e. $\mathcal{N}$). To begin with, we rewrite $\mathfrak{R}(F_\Theta)$ as:

$$\mathfrak{R}(F_\Theta) = \mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\frac{1}{m}\sum_{\alpha=1}^{m}\sigma_\alpha\ell(f(i_\alpha,j_\alpha),R_{i_\alpha,j_\alpha}))\big] = \mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\frac{1}{m}\sum_{(i,j)}\Gamma_{ij}\ell(f(i,j),R_{ij})\big],$$

where $\Gamma\in\mathbb{R}^{n_1\times n_2}$, $\Gamma_{ij} = \sum_{\alpha:i_\alpha=i,j_\alpha=j}\sigma_\alpha$. Now, using the same trick in Shamir and Shalev-Shwartz (2014), we can divide Γ based on the "hit-time" of each $(i,j)\in\Omega_{obs}$, with some threshold $p>0$ whose value will be set later. Formally, let $h_{ij} = |\{\alpha : i_\alpha = i, j_\alpha = j\}|$, and let $A, B\in\mathbb{R}^{n_1\times n_2}$ be defined by:

$$A_{ij} = \begin{cases}\Gamma_{ij}, & \text{if } h_{ij}>p,\\ 0, & \text{otherwise.}\end{cases} \qquad B_{ij} = \begin{cases}0, & \text{if } h_{ij}>p,\\ \Gamma_{ij}, & \text{otherwise.}\end{cases}$$

By construction, $\Gamma = A+B$. Therefore, we can separate $\mathfrak{R}(F_\Theta)$ to be:

$$\mathfrak{R}(F_\Theta) = \mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\frac{1}{m}\sum_{(i,j)}A_{ij}\ell(f(i,j),R_{ij})\big] + \mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\frac{1}{m}\sum_{(i,j)}B_{ij}\ell(f(i,j),R_{ij})\big]. \tag{13}$$

For the first term in (13), since $|\ell(f(i,j),R_{ij})|\leq\mathcal{B}$, it can be upper bounded by:

$$\mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\frac{1}{m}\sum_{(i,j)}A_{ij}\ell(f(i,j),R_{ij})\big] \leq \frac{\mathcal{B}}{m}\mathbb{E}_\sigma\big[\sum_{(i,j)}|A_{ij}|\big] \leq \frac{\mathcal{B}}{\sqrt{p}}$$

where the last inequality is derived by applying Lemma 10 in Shamir and Shalev-Shwartz (2014). Now consider the second term of (13). Again, by Rademacher contraction principle, it can be upper bounded by:

$$\begin{aligned}&\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{f\in F_\Theta}\sum_{(i,j)}B_{ij}f(i,j)\big]\\ =&\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{\|M\|_*\le\mathcal{M}}\sum_{(i,j)}B_{ij}\mathbf{x}_i^TM\mathbf{y}_j\big]+\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{\|N\|_*\le\mathcal{N}}\sum_{(i,j)}B_{ij}N_{ij}\big]+\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{\|S\|_1\le\mathcal{S}}\sum_{(i,j)}B_{ij}S_{ij}\big].\end{aligned}\tag{14}$$

We can again upper bound the first and third term of (14) using Lemma 9 and Lemma 10. Precisely, the first term can be upper bounded by:

$$\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{\|M\|_*\le\mathcal{M}}\sum_{\alpha=1}^m\sigma_\alpha\mathbf{x}_{i_\alpha}^TM\mathbf{y}_{j_\alpha}\big]=L_\ell\mathbb{E}_\sigma\big[\sup_{\|M\|_*\le\mathcal{M}}\frac{1}{m}\sum_{\alpha=1}^m\sigma_\alpha\mathrm{trace}(M\mathbf{y}_{j_\alpha}\mathbf{x}_{i_\alpha}^T)\big]\le 2L_\ell\mathcal{M}\mathcal{X}\mathcal{Y}\sqrt{\frac{\log 2d}{m}},$$

and the third term of (14) is upper bounded by:

$$L_\ell\mathbb{E}_\sigma\big[\sup_{\|S\|_1\le\mathcal{S}}\frac{1}{m}\sum_{\alpha=1}^m\sigma_\alpha\mathrm{trace}(S\mathbf{e}_{j_\alpha}\mathbf{e}_{i_\alpha}^T)\big]\le L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}}.$$

In addition, by applying Hölder's inequality, the second term of (14) is upper bounded by:

$$\frac{L_\ell}{m}\mathbb{E}_\sigma\big[\sup_{\|N\|_*\le\mathcal{N}}\sum_{(i,j)}B_{ij}N_{ij}\big]\le\frac{L_\ell}{m}\sup_{N:\|N\|_*\le\mathcal{N}}\|B\|_2\|N\|_*=\frac{L_\ell\mathcal{N}}{m}\mathbb{E}_\sigma\big[\|B\|_2\big]\le\frac{2.2CL_\ell\mathcal{N}\sqrt{p}(\sqrt{n_1}+\sqrt{n_2})}{m},$$

where the last inequality is derived by applying Lemma 11 in Shamir and Shalev-Shwartz (2014). Therefore, putting all of the above upper bounds to (13) and choosing p to be $m\mathcal{B}/(2.2CL_\ell\mathcal{N}(\sqrt{n_1}+\sqrt{n_2}))$, we obtain another upper bound on $\mathfrak{R}(F_\Theta)$ as:

$$\mathfrak{R}(F_\Theta)\le 2L_\ell\mathcal{M}\mathcal{X}\mathcal{Y}\sqrt{\frac{\log 2d}{m}}+\sqrt{9CL_\ell\mathcal{B}\frac{\mathcal{N}(\sqrt{n_1}+\sqrt{n_2})}{m}}+L_\ell\mathcal{S}\sqrt{\frac{2\log(2n_1n_2)}{m}}.\tag{15}$$

Lemma 3 thus follows by combining (12) and (15). ■

Proof of Lemma 4

Proof To begin with, we have:

$$\|X_\mu^TL_0Y_\nu\|_2\le\|X_\mu\|_2\|L_0\|_2\|Y_\nu\|_2\le\sigma_x\sigma_y\|L_0\|_*,$$

where σ_x (or σ_y) is the largest singular value of X_μ (or Y_ν). Therefore, by the definition of $\hat{M}$, we have:

$$\|\hat{M}\|_*\le d\|\hat{M}\|_2=d\|(X_\mu^TX_\mu)^\dagger X_\mu^TL_0Y_\nu(Y_\nu^TY_\nu)^\dagger\|_2\le\frac{\sigma_x\sigma_yd\|L_0\|_*}{\sigma_{xm}^2\sigma_{ym}^2},\tag{16}$$

where σ_{xm} (or σ_{ym}) is the smallest non-zero singular value of X_μ (or Y_ν). Furthermore, by the construction of X_μ and Y_ν, we have $\sigma_{xm} \geq \mu\sigma_x$ and $\sigma_{ym} \geq \nu\sigma_y$. We can further lower bound σ_x (and σ_y) by:

$$\sigma_x^2 = \|X_\mu\|_2^2 = \|X\|_2^2 \geq \frac{\|X\|_F^2}{d} \geq \frac{n \min \|\mathbf{x}_i\|^2}{d} \geq \frac{n\gamma^2\mathcal{X}^2}{d}.$$

Therefore, from (16), we can further bound $\|\hat{M}\|_*$ by:

$$\|\hat{M}\|_* \leq \frac{d\|L_0\|_*}{\mu^2\nu^2\sigma_x\sigma_y} \leq \frac{d^2\|L_0\|_*}{\mu^2\nu^2\gamma^2\mathcal{X}\mathcal{Y}\sqrt{n_1 n_2}}.$$

The lemma is thus concluded by the fact that $\|L_0\|_* \leq C_L\sqrt{n_1 n_2}$. ■

Proof of Theorem 5

Proof The claim is directly proved by plugging Lemma 3 - 4 to Lemma 2, in which $\hat{R}_\ell(f^*) = 0$ because $(\hat{M}, L_0 - X\hat{M}Y^T, S_0) \in \Theta$ and such an instance makes $\hat{R}_\ell = 0$. ■

Proof of Theorem 6

Proof Note that since $S_0 = S^* = 0$ in matrix completion case, we have:

$$R_\ell(f^*) = \mathbb{E}_{(i,j)\sim\mathcal{D}}\big[\ell(XM^*Y_{ij}^T, \mathbf{e}_i^T L_0 \mathbf{e}_j)\big].$$

The claim therefore directly follows from Theorem 5 by setting $R_\ell(f^*) < \epsilon$. ■

Proof of Theorem 7

Proof By the construction of X and Y, we can rewrite them as follows:

$$X = \sum_{i=1}^{t-1} \mathbf{u}_i\mathbf{e}_i^T + \sum_{i=t}^{d} \tilde{\mathbf{u}}_i\mathbf{e}_i^T, \qquad Y = \sum_{i=1}^{t-1} \mathbf{v}_i\mathbf{e}_i^T + \sum_{i=t}^{d} \tilde{\mathbf{v}}_i\mathbf{e}_i^T, \tag{17}$$

where for each $\tilde{\mathbf{u}}_i$, $\tilde{\mathbf{u}}_i^T\mathbf{u}_j = 0, \forall j$. Therefore, we can upper bound $\mathcal{N}$ by:

$$\begin{aligned}\|L_0 - X\hat{M}Y^T\|_* &= \|\tilde{U}\tilde{U}^T L_0 + L_0\tilde{V}\tilde{V}^T - \tilde{U}\tilde{U}^T L_0\tilde{V}\tilde{V}^T\|_* \\ &\leq 2\|\tilde{U}\tilde{U}^T U\Sigma V^T\|_* + \|U\Sigma V^T\tilde{V}\tilde{V}^T\|_* \\ &\leq 3\sum_{i=t}^{k}\sigma_i,\end{aligned}$$

where $\tilde{U}$, $\tilde{V}$ are the second term of X and Y in (17). Moreover, we have $\sigma_i = o(\sqrt{n})$ for all $i \geq t$. To see this, suppose $\sigma_p = \Omega(\sqrt{n})$ for any $t \leq p \leq k$, then:

$$\lim_{n\to\infty}\frac{\sigma_t}{\sqrt{n}} \geq \lim_{n\to\infty}\frac{\sigma_p}{\sqrt{n}} > 0,$$

leading a contradiction to the definition of σ_t. Therefore we can conclude:

$$\mathcal{N} = \|L_0 - X\hat{M}Y^T\|_* \leq 3\sum_{i=t}^{k} \sigma_i \leq 3k \times o(\sqrt{n}) = o(\sqrt{n}),$$

and the Theorem is thus proved by plugging the above bound on $\mathcal{N}$ to Theorem 6. ■

Proof of Theorem 8

Proof The sample complexity claim directly follows from Theorem 5 by setting $R_\ell(f^*) < \epsilon$, and the claim of $\mathcal{P}_{\Omega_{obs}^\perp}(S^*) = 0$ is directly from the construction of Algorithm 1 as discussed in Section 2.4. ■

References

P. L. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2003.

J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *Society for Industrial and Applied Mathematics*, 20(4):1956–1982, 2010.

E. J. Candès and Y. Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6): 925–936, 2010.

E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transaction of Information Theory*, 56(5):2053–2080, 2009.

E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of ACM*, 58(3):11:1–11:37, 2011.

V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2), 2011.

T. Chen, W. Zhang, Q. Lu, K. Chen, Z. Zheng, and Y. Yu. SVDFeature: A toolkit for feature-based collaborative filtering. *Journal of Machine Learning Research*, 13:3619–3622, 2012.

Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis. Low-rank matrix recovery from errors and erasures. *IEEE Transaction of Information Theory*, 59(7):4324–4337, 2013.

Y. Chen, A. Jalali, S. Sanghavi, and H. Xu. Clustering partially observed graphs via convex optimization. *Journal of Machine Learning Research*, 15(1):2213–2238, 2014.

K.-Y. Chiang, C.-J. Hsieh, N. Natarajan, I. S. Dhillon, and A. Tewari. Prediction and clustering in signed networks: A local to global perspective. *Journal of Machine Learning Research*, 15:1177–1213, 2014.

K.-Y. Chiang, C.-J. Hsieh, and I. S. Dhillon. Matrix completion with noisy side information. In *Advances in Neural Information Processing Systems*, 2015.

K.-Y. Chiang, C.-J. Hsieh, and I. S. Dhillon. Robust principal component analysis with side information. In *International Conference on Machine Learning*, 2016.

J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *International Conference on Machine Learning*, pages 209–216, 2007.

M. Fazel, H. Hindi, and S. P. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference*, volume 6, pages 4734–4739, 2001.

W. Ha and R. F. Barber. Robust PCA with compressed data. In *Advances in Neural Information Processing Systems*, 2015.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.

C.-J. Hsieh and P. A. Olsan. Nuclear norm minimization via active subspace selection. In *International Conference on Machine Learning*, 2014.

C.-J. Hsieh, K.-Y. Chiang, and I. S. Dhillon. Low rank modeling of signed networks. In *International Conference on Knowledge Discovery and Data Mining*, pages 507–515, 2012.

C.-J. Hsieh, N. Natarajan, and I. S. Dhillon. PU learning for matrix completion. In *International Conference on Machine Learning*, 2015.

P. Jain and I. S. Dhillon. Provable inductive matrix completion. *CoRR*, abs/1306.0626, 2013.

P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Symposium on Theory of Computing*, pages 665–674, 2013.

S. M. Kakade, K. Sridharan, and A. Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in Neural Information Processing Systems*, pages 793 – 800, 2008.

R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11:2057–2078, 2010.

Y. Koren, R. M. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42:30–37, 2009.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *International Conference on World Wide Web*, pages 641–650, 2010.

Z. Li and J. Liu. Constrained clustering by spectral kernel learning. In *International Conference on Computer Vision*, 2009.

D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.*, 58(7):1019–1031, 2007.

G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *International Conference on Machine Learning*, 2010.

G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Aanalysis and Machine Intelligence*, 35(1):171–184, 2013.

P. Massa and P. Avesani. Trust-aware bootstrapping of recommender systems. In *ECAI Workshop on Recommender Systems*, pages 29–33, 2006.

R. Meir and T. Zhang. Generalization error bounds for bayesian mixture algorithms. *Journal of Machine Learning Research*, 4:839–860, 2003.

A. K. Menon and C. Elkan. Link prediction via matrix factorization. *European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 437–452, 2011.

N. Natarajan and I. S. Dhillon. Inductive matrix completion for predicting gene-disease associations. *Bioinformatics*, 30(12):60–68, 2014.

S. Negahban and M. J. Wainwright. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research*, 13(1): 1665–1697, 2012.

B. Recht. A simpler approach to matrix completion. *Journal of Machine Learning Research*, 12:3413–3430, 2011.

O. Shamir and S. Shalev-Shwartz. Matrix completion with the trace norm: Learning, bounding, and transducing. *Journal of Machine Learning Research*, 15(1):3401–3423, 2014.

D. Shin, S. Cetintas, K.-C. Lee, and I. S. Dhillon. Tumblr blog recommendation with boosted inductive matrix completion. In *International Conference on Information and Knowledge Management*, pages 203–212, 2015.

N. Srebro and A. Shraibman. Rank, trace-norm and max-norm. In *Annual Conference on Learning Theory*, pages 545–560, 2005.

Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3 (1):71–86, 1991.

J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *Advances in Neural Information Processing Systems*, 2009.

H. Xu, C. Caramanis, and S. Sanghavi. Robust PCA via outlier pursuit. In *Advances in Neural Information Processing Systems*, pages 2496–2504, 2010.

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*, 2013.

J. Yi, L. Zhang, R. Jin, Q. Qian, and A. Jain. Semi-supervised clustering by input pattern assisted pairwise similarity matrix completion. In *International Conference on Machine Learning*, 2013.

K. Zhong, P. Jain, and I. S. Dhillon. Efficient matrix sensing using rank-1 Gaussian measurements. In *International Conference on Algorithmic Learning Theory*, 2015.