

TRANSDUCTIVE MATRIX COMPLETION WITH CALIBRATION FOR MULTI-TASK LEARNING

Hengfang Wang*, Yasi Zhang[◇], Xiaojun Mao[†], Zhonglei Wang[‡]

* Fujian Normal University, hengfang@fjnu.edu.cn; [◇] UCLA, yasminzhang@ucla.edu;

[†] Shanghai Jiao Tong University, maobj@sju.edu.cn; [‡] Xiamen University, wangzl@xmu.edu.cn

ABSTRACT

Multi-task learning has attracted much attention due to growing multi-purpose research with multiple related data sources. Moreover, transduction with matrix completion is a useful method in multi-label learning. In this paper, we propose a transductive matrix completion algorithm that incorporates a calibration constraint for the features under the multi-task learning framework. The proposed algorithm recovers the incomplete feature matrix and target matrix simultaneously. Fortunately, the calibration information improves the completion results. In particular, we provide a statistical guarantee for the proposed algorithm, and the theoretical improvement induced by calibration information is also studied. Moreover, the proposed algorithm enjoys a sub-linear convergence rate. Several synthetic data experiments are conducted, which show the proposed algorithm out-performs other methods, especially when the target matrix is associated with the feature matrix in a nonlinear way.

Index Terms— Matrix completion, multi-task learning

1. INTRODUCTION

With the advent of the big data era, massive amounts of information and data have been collected by modern high-tech devices, including web servers, environmental sensors, x-ray imaging machines, and so on. Based on multiple data sources related to learning purposes, algorithms have been developed to achieve various learning goals. Multi-task learning (MTL) [1], for example, implements a robust learner for multiple tasks incorporating multiple sources, and it can be widely used in practice, including web search, medical diagnosis, natural language processing, and computer vision.

MTL has advantages for analyzing the multi-response structure. For $i = 1, \dots, n$, let $\mathbf{y}_i \in \mathbb{R}^m$ be the vector of interest of length m , where n is the number of instances, and each element corresponds to a particular task. Denote $\mathbf{Y} = (\mathbf{y}_{ij})$ to be the $n \times m$ response matrix, consisting of the n realizations of \mathbf{y}_i . If the responses for a specific task are categorical, it is essentially a multi-label learning problem. As for continuous responses, it turns out to be a multi-response regression problem if additional features are available. As the exponential family is flexible to handle different data types, regardless of categorical or continuous, for each task, we assume that y_{ij} independently comes from an exponential family distribution with parameter $z_{*,ij}$, which forms $\mathbf{Z}_* = (z_{*,ij})$. The goal is to learn the underlying \mathbf{Z}_* based on \mathbf{Y} . Moreover, entries of \mathbf{Y} may suffer from missingness which makes it difficult to learn \mathbf{Z}_* . To tackle this difficulty and

estimate \mathbf{Z}_* simultaneously, matrix completion (MC) [2, 3, 4, 5, 6] algorithms have been developed under the low-rank and other regularity assumptions. The recovery of the target matrix without any additional features is studied in [7]. Noisy MC is investigated in [8, 9] from the exponential family with fully observed features.

In MTL problems, features related to the responses may exist, but it is inevitable that such features also suffer from missingness. Goldberg et al. (2010) [10] studied such a problem under multi-label learning. Specifically, they studied single-task binary response matrix given feature matrix where both matrices were subject to missingness. Xu et al. (2018) [11] proposed the co-completion method by additionally penalizing the trace norm of the feature matrix. A recent work [12] considered this problem via a smoothed rank function approach. Under the noiseless case, [13] studied the error rate in the scenario with corrupted side information.

Calibration [14, 15, 16] is widely used to incorporate such prior knowledge in the learning procedure, and it improves the estimation efficiency. In this vein, [17] studied the calibration with local polynomial regression between responses and features. Later, [18] generalized this idea to the so-called model calibration under a parametric framework. Afterward, [19] generalized the idea to a functional type calibration equation. As far as we know, no calibration related work has been done under the MC framework.

In this paper, we focus on MTL problems incorporating incomplete features, and we assume that certain prior knowledge about the features is also available. Our work can be embodied in the following toy example. For the well-known Netflix movie rating problem [20], we aim to simultaneously complete rating and like-dislike matrices incorporating an incomplete feature matrix, consisting of age, gender, and so on. However, all three matrices suffer from missingness. When additional information such as the summary statistics for age, gender, etc., can be obtained from the census survey, we investigate the benefits of such additional information incorporated by the calibration method. In a nutshell, we propose a *Transductive Matrix Completion with Calibration* (TMCC) algorithm, and the prior information about the features is considered by an additional calibration constraint. As far as we know, this is the first paper exploring multi-task learning under a matrix completion framework with calibration. Methodologically, our method has two merits: (i) the target and feature matrices can be completed simultaneously; (ii) calibration information can be incorporated. Theoretically, we show the statistical guarantee of our method, and the benefit of calibration is also discussed. Besides, we have validated that the convergence rate of the proposed algorithm is $O(1/k^2)$. Numerically, synthetic data experiments show that our proposed method performs better than other existing methods, especially when the target matrix has a nonlinear transformation from the feature matrix.

Notations. Denote $[n]$ as the set $\{1, \dots, n\}$. Given an arbitrary matrix $\mathbf{S} = (s_{ij}) \in \mathbb{R}^{n_1 \times n_2}$, the Frobenius norm of

This work is supported by Fujian Provincial Department of Education (JAT210059), the Fundamental Research Funds for the Central Universities, NSFC (12001109, 11901487, 72033002, 71988101), and Science and Technology Commission of Shanghai Municipality Grant (20dz1200600). X. Mao is the corresponding author (E-mail: maobj@sju.edu.cn).

\mathbf{S} is $\|\mathbf{S}\|_F = (\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} s_{ij}^2)^{1/2}$. Denote the singular values of \mathbf{S} as $\sigma_1(\mathbf{S}), \dots, \sigma_{n_1 \wedge n_2}(\mathbf{S})$ in descending order. The operator norm is $\|\mathbf{S}\| = \sigma_{\max}(\mathbf{S}) = \sigma_1(\mathbf{S})$ and the nuclear norm $\|\mathbf{S}\|_* = \sum_{i=1}^{n_1 \wedge n_2} \sigma_i(\mathbf{S})$. Besides, let $\sigma_{\min}(\mathbf{S}) = \sigma_{n_1 \wedge n_2}(\mathbf{S})$.

2. MODEL AND ALGORITHM

Suppose there are S response matrices, $\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(S)}$. For example, $\mathbf{Y}^{(1)}$ can be a like-dislike matrix, and $\mathbf{Y}^{(2)}$ can be a rating matrix, whose rows correspond to users, columns correspond to movies, and $S = 2$. Denote the number of instances by n , the number of tasks for s -th response matrix by m_s . For $s \in [S]$, we have $\mathbf{Y}^{(s)} = (y_{ij}^{(s)}) \in \mathbb{R}^{n \times m_s}$, and $y_{ij}^{(s)}$ can be either discrete or continuous for $i \in [n]$. We assume that within the same response matrix, all entries follow the same generic exponential family. For example, all entries of $\mathbf{Y}^{(1)}$ follow Bernoulli distributions with different success probabilities. Let $\mathbf{R}_y^{(s)} = (r_{y,ij}^{(s)}) \in \mathbb{R}^{n \times m_s}$ be the corresponding indicator matrix for $\mathbf{Y}^{(s)}$. If $y_{ij}^{(s)}$ is observed, then $r_{y,ij}^{(s)} = 1$; otherwise, $r_{y,ij}^{(s)} = 0$. Furthermore, assume that $\mathbf{Y}^{(s)}$ is generated from a low-rank matrix $\mathbf{Z}_*^{(s)} = (z_{*,ij}^{(s)}) \in \mathbb{R}^{n \times m_s}$ by the exponential family via a base function $h^{(s)}$ and a link function $g^{(s)}$, namely, the density function $f^{(s)}(y_{ij}^{(s)} | z_{*,ij}^{(s)}) = h^{(s)}(y_{ij}^{(s)}) \exp\{y_{ij}^{(s)} z_{*,ij}^{(s)} - g^{(s)}(z_{*,ij}^{(s)})\}$, for $s \in [S]$. For instance, suppose $g^{(s)}(z_{*,ij}^{(s)}) = \sigma^2 (z_{*,ij}^{(s)})^2 / 2$ and $h^{(s)}(y_{ij}^{(s)}) = (2\pi\sigma^2)^{-1/2} \exp\{-(y_{ij}^{(s)})^2 / (2\sigma^2)\}$, and the corresponding exponential family is the Gaussian distribution with mean $\sigma^2 z_{*,ij}^{(s)}$ and variance σ^2 with support \mathbb{R} .

Denote $\mathbf{X}_* = (x_{*,ij}) \in \mathbb{R}^{n \times d}$ as the true feature matrix consisting of d feature, and it is assumed to be low-rank. Let $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{n \times d}$ be a noisy version of the true feature matrix, i.e., $\mathbf{X} = \mathbf{X}_* + \epsilon$, where $\epsilon = (\epsilon_{ij}) \in \mathbb{R}^{n \times d}$ is a noise matrix with $\mathbb{E}(\epsilon) = \mathbf{0}$, and its entries are independent. As the feature matrix is also incomplete, in a similar fashion, we denote $\mathbf{R}_x = (r_{x,ij}) \in \mathbb{R}^{n \times d}$ as the corresponding indicator matrix of \mathbf{X} . That is, we only observe an incomplete matrix $\mathbf{R}_x \circ \mathbf{X}$, where \circ denotes the Hadamard product. Let target matrix $\mathbf{Z}_* = [\mathbf{Z}_*^{(1)}, \dots, \mathbf{Z}_*^{(S)}]$ be the collection of hidden parameter matrices. Our goal is to recover \mathbf{Z}_* . We believe that some hidden connection between \mathbf{X}_* and \mathbf{Z}_* may provide us benefits for MTL.

Our method can be illustrated in Fig 1, where $\text{logit}(p) = \log\{p/(1-p)\}$, for $p \in (0, 1)$. In this example, we have three observed matrices whose entries are from Bernoulli, Gaussian, and Poisson distributions, correspondingly, and are subject to missingness. In addition, we have an incomplete feature matrix related to the target matrix and calibration information for the true feature matrix. By the proposed TMCC algorithm, we can complete the true feature matrix and target matrix simultaneously.

Given density functions $\{f^{(s)}\}_{s=1}^S$, or collection of $\{g^{(s)}\}_{s=1}^S$ and $\{h^{(s)}\}_{s=1}^S$, the negative quasi log-likelihood function for \mathbf{Z}^\dagger is

$$\begin{aligned} \ell_0(\mathbf{Z}^\dagger) &= - \sum_{s=1}^S \sum_{(i,j) \in [n] \times [m_s]} r_{y,ij}^{(s)} \log \{f^{(s)}(y_{ij}^{(s)} | z_{ij}^{\dagger(s)})\} \\ &= \sum_{s=1}^S \sum_{(i,j) \in [n] \times [m_s]} r_{y,ij}^{(s)} \left[\{-y_{ij}^{(s)} z_{ij}^{\dagger(s)} + g^{(s)}(z_{ij}^{\dagger(s)})\} + \log \{h^{(s)}(y_{ij}^{(s)})\} \right] \end{aligned} \quad (1)$$

where $\mathbf{Z}^\dagger = [\mathbf{Z}^{\dagger(1)}, \dots, \mathbf{Z}^{\dagger(S)}]$ and $\mathbf{Z}^{\dagger(s)} = (z_{ij}^{\dagger(s)}) \in \mathbb{R}^{n \times m_s}$. As \mathbf{Z}^\dagger is irrelevant with the second term of (1), we refine (1) as

$$\ell(\mathbf{Z}^\dagger) = \sum_{s=1}^S \sum_{(i,j) \in [n] \times [m_s]} r_{y,ij}^{(s)} \{-y_{ij}^{(s)} z_{ij}^{\dagger(s)} + g^{(s)}(z_{ij}^{\dagger(s)})\}.$$

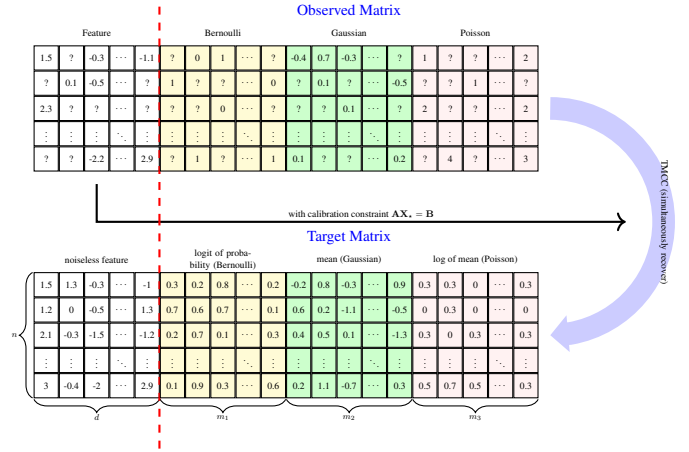


Fig. 1: Algorithm Illustration.

Let $\mathbf{Y} = [\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(S)}]$ be the collection of tasks, $D = d + \sum_{s=1}^S m_s$ and \mathbf{M}_* be the concatenated matrix $[\mathbf{X}_*, \mathbf{Z}_*] \in \mathbb{R}^{n \times D}$. Suppose we have an additional calibration constraint $\mathbf{A}\mathbf{X}_* = \mathbf{B}$, where \mathbf{A} and \mathbf{B} are available. To incorporate the calibration constraint, the estimation procedure can be formulated as

$$\begin{aligned} \widehat{\mathbf{M}} &= \arg \min_{\mathbf{M}^\dagger \in \mathbb{R}^{n \times D}} \left\{ \frac{1}{nD} \left\{ \ell(\mathbf{Z}^\dagger) + \frac{1}{2} \|\mathbf{R}_x \circ (\mathbf{X}^\dagger - \mathbf{X})\|_F^2 \right\} \right. \\ &\quad \left. + \tau_1 \|\mathbf{A}\mathbf{X}^\dagger - \mathbf{B}\|_F^2 + \tau_2 \|\mathbf{M}^\dagger\|_* \right\} \\ &=: \arg \min_{\mathbf{M}^\dagger \in \mathbb{R}^{n \times D}} f_{\tau_1}(\mathbf{M}^\dagger) + \tau_2 \|\mathbf{M}^\dagger\|_* \\ &=: \arg \min_{\mathbf{M}^\dagger \in \mathbb{R}^{n \times D}} \mathcal{L}_{\tau_1, \tau_2}(\mathbf{M}^\dagger), \end{aligned} \quad (2)$$

where $\mathbf{M}^\dagger = [\mathbf{X}^\dagger, \mathbf{Z}^\dagger]$, $\widehat{\mathbf{M}} = [\widehat{\mathbf{X}}, \widehat{\mathbf{Z}}]$ and we employ the commonly used square loss to complete the feature part. One trivial case is that when the calibration information is strong enough, i.e., \mathbf{A} is invertible, the feature matrix \mathbf{X}_* can be exactly recovered. Note that our learning object is quite general. When $S = 1$, the entries within the response matrix come from Bernoulli distributions, and there's no third term related to calibration in (2), our learning object degenerates to the case considered in [10], and [11]. However, [10] had an additional assumption that there exists a linear relationship between the feature matrix and the hidden parameter matrix. We do not make such a structural assumption. Besides, [11] used additional nuclear norms to penalize the feature matrix for single task learning. When there is no feature information, the objective function will degenerate to the case in [7].

We propose a *Transductive Matrix Completion with Calibration* (TMCC) algorithm to obtain the estimator in (2). In the algorithm, the ij -th element of the gradient $\partial f_{\tau_1}(\mathbf{M}^\dagger)$ with respect to \mathbf{M}^\dagger is

$$\partial f_{\tau_1}(\mathbf{M}^\dagger)_{ij} = \begin{cases} \frac{r_{x,ij}}{nD} (x_{ij}^\dagger - x_{ij}) + 2\tau_1 (\mathbf{A}^T \mathbf{A} \mathbf{X}^\dagger - \mathbf{A}^T \mathbf{B})_{ij} & \text{for } j \in [d], \\ \frac{r_{y,ij}}{nD} \left\{ -y_{ij}^{(s)} + \frac{\partial g^{(s)}(z_{ij}^{\dagger(s)})}{\partial z_{ij}^{\dagger(s)}} \right\}, & \\ \text{for } \sum_{k=1}^{s-1} m_k + d < j \leq \sum_{k=1}^s m_k + d, & \end{cases}$$

where $j_\circ = j - d - \sum_{k=1}^{s-1} m_k$. For any generic matrix \mathbf{S} with the

Algorithm 1: TMCC algorithm

Input: Incomplete matrices \mathbf{X}, \mathbf{Y} ; indicator matrices $\mathbf{R}_x, \mathbf{R}_y$; calibration constraint matrices \mathbf{A} and \mathbf{B} , tuning parameters τ_1, τ_2 ; learning depth K , step size η , stopping criterion κ .

Initialize: Random matrices $\mathbf{M}^{(0)} = \mathbf{M}^{(1)} \in \mathbb{R}^{n \times D}$, $c = 1$.

```

1 for  $k = 1$  to  $K$  do
2   Compute  $\theta = (c - 1)/(c + 2)$ .
3   Compute  $\mathbf{Q} = (1 + \theta)\mathbf{M}^{(k)} - \theta\mathbf{M}^{(k-1)}$ .
4   Compute  $\mathbf{T} = \mathbf{Q} - \eta\partial f_{\tau_1}(\mathbf{Q})$ .
5   Compute  $\mathbf{M}^{(k+1)} = \mathcal{T}_{\eta\tau_2}(\mathbf{T})$ .
6   if  $\mathcal{L}_{\tau_1, \tau_2}(\mathbf{M}^{(k+1)}) > \mathcal{L}_{\tau_1, \tau_2}(\mathbf{M}^{(k)})$  then
7      $c = 1$ ;
8   else
9      $c = c + 1$ ;
10  if  $|\mathcal{L}_{\tau_1, \tau_2}(\mathbf{M}^{(k+1)}) - \mathcal{L}_{\tau_1, \tau_2}(\mathbf{M}^{(k)})| \leq \kappa$  then
11     $\mathbf{M}^\dagger = \mathbf{M}^{(k+1)}$ ;
12    break;

```

Output: \mathbf{M}^\dagger .

singular value decomposition (SVD) $\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, $\mathbf{\Sigma} = \text{diag}(\sigma_i)$, $\sigma_1 \geq \dots \geq \sigma_r$ and $r = \text{rank}(\mathbf{S})$. Denote the singular value soft threshold operator by $\mathcal{T}_c(\mathbf{S}) = \mathbf{U}\text{diag}((\sigma_i - c)_+) \mathbf{V}^T$ for a constant $c > 0$ and $x_+ = \max(x, 0)$. The detailed algorithm is presented in Algorithm 1, where a question mark represents a Missing Value. The parameter K is a predetermined integer that controls the learning depth of TMCC, and η is a predetermined constant for the step size. Within the algorithm, κ in the Algorithm 1 is a predetermined stopping criterion. The proposed algorithm iteratively updates the gradient and does singular value thresholding (SVT) [21], in addition to an accelerated proximal gradient decent [22] step to get a fast convergence rate.

3. THEORETICAL GUARANTEE

We make the following technical assumptions.

Assumption 1. There exists a positive constant p_{\min} such that

$$\min \left[\left\{ \min_{s \in [S]} \min_{(i,j) \in [n] \times [m_s]} \pi_{y,ij}^{(s)} \right\}, \left\{ \min_{(i,j) \in [n] \times [d]} \pi_{x,ij} \right\} \right] \geq p_{\min},$$

where $\pi_{x,ij} = \mathbb{P}(r_{x,ij} = 1)$, $\pi_{y,ij}^{(s)} = \mathbb{P}(r_{y,ij}^{(s)} = 1)$. Further, there exists a positive constant γ such that

$$\max \left(\left[\max_{i \in [n]} \{ \pi_{x,i} + \pi_{y,i} \} \right], \left[\max_{j \in [d]} \pi_{x,j} \right], \left[\max_{s \in [S]} \max_{j \in [m_s]} \pi_{y,j}^{(s)} \right] \right) \leq \gamma,$$

where $\pi_{x,i} = \sum_{j \in [d]} \pi_{x,ij}$, $\pi_{x,j} = \sum_{i \in [n]} \pi_{x,ij}$, $\pi_{y,j}^{(s)} = \sum_{i \in [n]} \pi_{y,ij}^{(s)}$, $\pi_{y,i} = \sum_{s \in [S]} \sum_{j \in [m_s]} \pi_{y,ij}^{(s)}$.

Assumption 2. There exists a positive constant α , such that $\max \{ \|\mathbf{X}_*\|_\infty, \|\mathbf{Z}_*\|_\infty \} \leq \alpha$.

Assumption 3. Let $\mathcal{D} = [-\alpha - \delta, \alpha + \delta]$, for some $\delta > 0$. For any $z \in \mathcal{D}$, there exist positive constants L_α and U_α , such that

$L_\alpha \leq (g^{(s)})''(z) \leq U_\alpha$, where $g^{(s)}$ is the link function of exponential family for s -th response matrix, for $s \in [S]$.

Assumption 4. There exists a positive constant ζ , for any $\lambda \in \mathbb{R}$, such that $\mathbb{E}(e^{\lambda \epsilon_{ij}}) \leq e^{\lambda^2 \zeta/2}$, where ϵ_{ij} 's are noises for the feature matrix.

Assumption 5. There exists a constant C such that $\sigma_{\min}(\mathbf{A}) \geq C > 0$.

Assumption 1 controls the sampling probabilities for our model. The first part ensures that all the sampling probabilities are bounded away from zero. The second part aims to bound the operator norm of a Rademacher matrix of the same dimension as \mathbf{M}_* stochastically. Although we assume that the parameter matrix \mathbf{Z}_* is bounded in Assumption 2, the support of entries in \mathbf{Y} can be unbounded. For instance, the support of a Poisson distribution is unbounded, while its mean is fixed. In other words, Assumption 2 implies $\mathbf{M}_* \in \mathcal{G}_\alpha(\alpha) := \{ \mathbf{G} \in \mathbb{R}^{n \times D} : \|\mathbf{G}\|_\infty \leq \alpha \}$. Assumption 3 is mild under the canonical exponential family framework. That is, we have $\text{Var}(y_{ij}^s | g^s(z_{*,ij}^s)) = (g^{(s)})''(z_{*,ij}^s) > 0$ for each i, j . We extend it a little bit with tolerance δ for ease of proof. Furthermore, define $\tilde{L}_\alpha = (L_\alpha \wedge 1)$ and $\tilde{U}_\alpha = (U_\alpha \vee \zeta \vee 1)$. Assumption 4 implies that the errors for the feature matrix come from sub-Gaussian distributions. Assumption 5 indicates that the calibration matrix \mathbf{A} should be of full rank. The following Theorem 1 shows the convergence rate of the proposed algorithm is $O(1/k^2)$.

Theorem 1. Suppose that Assumption 1 ~ 4 hold, $\tau_1 \leq c_1 \min \{ \sigma_{\max}(\mathbf{A}) \}^{-2} (nD\alpha)^{-1/2}, \{ nD\sigma_{\min}^2(\mathbf{A}) \}^{-1} \tilde{L}_\alpha p_{\min} \}$ and $\tau_2 = (nD)^{-1} 2c_2 \{ (\tilde{U}_\alpha)^{1/2} \vee 1/\delta \} \{ \gamma^{1/2} + (\log(n \vee D))^{3/2} \}$. The sequences $\{ \mathbf{M}^{(k)} \}$ generated by Algorithm 1 satisfy

$$f_{\tau_1}(\mathbf{M}^{(k)}) - f_{\tau_1}(\mathbf{M}_*) \leq \frac{2\tilde{L} \|\mathbf{M}^{(0)} - \mathbf{M}_*\|_F^2}{\eta(k+1)^2},$$

with probability at least $1 - 4/(n+D)$, where \tilde{L} is a constant related to c_1 and $\alpha^{1/2} \tilde{U}_\alpha$.

The following theorem presents the statistical guarantee for the proposed method.

Theorem 2. Suppose that Assumption 1 ~ 4 hold, $\tau_1 \leq \{ nD\sigma_{\min}^2(\mathbf{A}) \}^{-1} c_1 \tilde{L}_\alpha p_{\min}$ and $\tau_2 = (nD)^{-1} 2c_2 \{ (\tilde{U}_\alpha)^{1/2} \vee 1/\delta \} \{ \gamma^{1/2} + (\log(n \vee D))^{3/2} \}$. Then, with probability at least $1 - 4/(n+D)$,

$$\begin{aligned} & \left\{ \frac{1}{nD} + \frac{8\tau_1}{\tilde{L}_\alpha p_{\min}} \sigma_{\min}^2(\mathbf{A}) \right\} \|\hat{\mathbf{X}} - \mathbf{X}_*\|_F^2 + \frac{1}{nD} \|\hat{\mathbf{Z}} - \mathbf{Z}_*\|_F^2 \\ & \leq \frac{c \text{rank}(\mathbf{M}_*)}{nD p_{\min}^2} \left\{ \alpha^2 + \frac{\tilde{U}_\alpha \vee 1/\delta^2}{\tilde{L}_\alpha^2} \right\} \{ \gamma + \log^3(n \vee D) \}, \end{aligned} \quad (3)$$

where c, c_1, c_2 are positive constants.

Theorem 2 implies that our recovered error for the target matrix, in the sense of squared Frobenius norm, is bounded by the right hand side of (3), with probability approaching 1 when n and D are large enough. Suppose the feature matrix is also regarded as a response matrix from Gaussian noise with unit variance, by directly applying the Theorem 7 in [7], we have $(nD)^{-1} (\|\hat{\mathbf{X}} - \mathbf{X}_*\|_F^2 + \|\hat{\mathbf{Z}} - \mathbf{Z}_*\|_F^2)$ less than the terms in the second line of (3). Fortunately, with the help of calibration information, we have a constant order improvement. Specifically, together with Assumption 5, we have

$$\frac{1}{nD} \|\hat{\mathbf{X}} - \mathbf{X}_*\|_F^2 + \frac{1}{nD} \|\hat{\mathbf{Z}} - \mathbf{Z}_*\|_F^2$$

$$\stackrel{(i)}{<} \left\{ \frac{1}{nD} + \frac{8\tau_1}{L_\alpha p_{\min}} \sigma_{\min}^2(\mathbf{A}) \right\} \|\hat{\mathbf{X}} - \mathbf{X}_*\|_F^2 + \frac{1}{nD} \|\hat{\mathbf{Z}} - \mathbf{Z}_*\|_F^2,$$

where the inequality (i) is strict, which is one of the main theoretical contributions of this paper.

4. EXPERIMENTS

We conduct a simulation study to illustrate the performance of the proposed method. Let $\mathbf{P} \in \mathbb{R}^{n \times r}$ and $\mathbf{Q} \in \mathbb{R}^{d \times r}$ whose entries are generated independently from a uniform distribution over $(0, 1)$. Then, let $\mathbf{X}_*^0 = \mathbf{P}\mathbf{Q}^\top$ and $\mathbf{X}_* = \mathbf{X}_*^0 / \|\mathbf{X}_*^0\|_\infty$. We further generate coefficient matrix $\mathbf{W}^{(s)} \in \mathbb{R}^{d \times m_s}$ whose elements are independently generated from a uniform distribution over $(0, 1)$ for $s = 1, 2, 3$. Let $\tilde{\mathbf{Z}}_*^{(s)} = \mathbf{X}_* \mathbf{W}^{(s)}$. We call this setting as ‘‘linear’’ case. By normalization, we have $\mathbf{Z}_*^{(s)} = \tilde{\mathbf{Z}}_*^{(s)} / \|\tilde{\mathbf{Z}}_*^{(s)}\|_\infty$. Specifically, $\mathbf{Y}^{(1)}$ has Bernoulli entries with support $\{0, 1\}$, $\mathbf{Y}^{(2)}$ has Poisson entries and $\mathbf{Y}^{(3)}$ has Gaussian entries with known $\sigma^2 = 1$. All the link between $\mathbf{Y}^{(s)}$ and $\mathbf{Z}^{(s)}$ are the same as those in Section 2. For calibration information, suppose we know the column means for \mathbf{X}_* , i.e., $\mathbf{A} = (1/n)\mathbf{1}_{1 \times n}$ and $\mathbf{B} = \mathbf{A}\mathbf{X}_*$. On the other hand, we have a ‘‘nonlinear’’ case, i.e., we assume that $\tilde{\mathbf{Z}}_*^{(s)}$ is generated by an element-wise nonlinear transformation of \mathbf{X}_* . Specifically, let $z_{*,ij}^{(s)} = t^{(s)}(x_{*,ij})$, where $t^{(1)}(x) = x^2 + x + 0.5$, $t^{(2)}(x) = -x^2 - x$, $t^{(3)}(x) = -x^2 - 2x + 0.2$. The normalization procedure is the same as the ‘‘linear’’ case. The proposed method TMCC is compared with three other approaches.

1. **MC_0**: Exactly the same as modified TMCC except for the gradient updating procedure. No calibration information is considered. Therefore in $\partial f_{\tau_1}(\mathbf{M}^\dagger)_{ij}$ the term $2\tau_1(\mathbf{A}^\top \mathbf{A} \mathbf{X}^\dagger - \mathbf{A}^\top \mathbf{B})_{ij}$ is eliminated.

2. **CMC_SI**: Collective matrix completion (CMC) [7] is used to complete the parameters for the target matrix, and Soft-Impute (SI) method from [23] is used to complete the feature matrix separately.

3. **TS**: A two-stage method, where, at the first stage, only the feature matrix is imputed by the Soft-Impute method, and at the second stage, the method **MC_0** is applied to the concatenated matrix joined by the feature matrix and the observed response matrices.

Specifically, **MC_0** and **TMCC** share the same strategy, i.e. simultaneously recovering all matrices, while **CMC_SI** chooses to recover separately and **TS** opts to recover step by step.

In the experiments, we set $n = 1500$, $d = 500$, $m_1 = m_2 = m_3 = 500$ and choose learning depth $K = 1000$ and stopping criterion $\kappa = 10^{-7}$. Further, we compare different methods with rank $r \in \{5, 10, 15\}$. The missing rate $\nu \in \{60\%, 80\%\}$ of both the feature matrix and response matrix are the same in each experiment. For **TMCC**, we tune τ_1 and τ_2 on one independent validation set and apply the same parameters to all other repeated 50 simulations. Further, other compared methods employ the same procedure as **TMCC** while they only have to be tuned for the parameter τ_2 .

The performance of each method is evaluated via the mean value and standard deviation of the relative errors (RE) based on repeated experiments. Specifically, the relative error of a recovered feature matrix is $\text{RE}(\hat{\mathbf{X}}) = \|\hat{\mathbf{X}} - \mathbf{X}_*\|_F / \|\mathbf{X}_*\|_F$ and that of target matrix $\text{RE}(\hat{\mathbf{Z}}) = \|\hat{\mathbf{Z}} - \mathbf{Z}_*\|_F / \|\mathbf{Z}_*\|_F$. Experiment results are summarized in Fig 2 and Fig 3.

In Fig 2, it is an unsurprising fact that **TMCC** surpasses **MC_0**

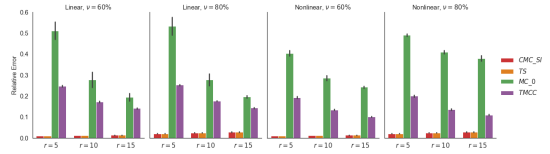


Fig. 2: Relative Error of Feature Matrix (with the Black Lines Representing \pm the Standard Error)

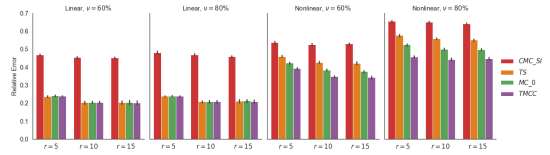


Fig. 3: Relative Error of Target Matrix (with the Black Lines Representing \pm the Standard Error)

with the help of calibration information. For instance, in the linear case with $\nu = 60\%$, the mean of RE of $\hat{\mathbf{X}}$ by **MC_0** is 0.51 with standard error (SE) 0.0417 when $r = 5$ while that by **TMCC** is 0.25 with SE 0.0040. **TMCC**’s RE is less than half of **MC_0**’s and situations in other cases are alike. Besides, **CMC_SI** and **TS** perform the best in all scenarios. Specifically, the mean of RE of $\hat{\mathbf{X}}$ by **CMC_SI** is 0.01 with SE 0.0005, and that by **TS** is 0.01 with SE 0.0006 when $r = 10$ in the nonlinear case with $\nu = 60\%$. It is because both **CMC_SI** and **TS** complete the feature matrix without considering the target matrix. However, our primary goal is to recover \mathbf{Z}_* , i.e., achieving a low relative error of $\hat{\mathbf{Z}}$. In Fig 3, the far lower relative error of $\hat{\mathbf{X}}$ by **CMC_SI** and **TS** does not bring a lower relative error of $\hat{\mathbf{Z}}$. Of all the four approaches, **CMC_SI** is the only one that fails to take advantage of the feature matrix to recover the target matrix. That is why it attains the greatest relative error of the target matrix. For example, its mean of RE of $\hat{\mathbf{Z}}$ is 0.46 with SE 0.0046 when $r = 15$ in the linear case with $\nu = 80\%$ and 0.53 with SE 0.0057 when $r = 15$ in the nonlinear case with $\nu = 60\%$, more than twice of the other three methods. Besides, **TS**, **MC_0**, and **TMCC** behave similarly in the linear case while they display great differences in the nonlinear case. When the relationship between \mathbf{X}_* and \mathbf{Z}_* is not linear, simultaneously recovering demonstrates great strength compared with recovering step by step and recovering separately. In the nonlinear case with $\nu = 80\%$, the means of RE of $\hat{\mathbf{Z}}$ by **CMC_SI**, **TS**, **MC_0**, and **TMCC** are 0.66, 0.58, 0.52, and 0.46 respectively when $r = 5$. Situations in other cases are almost the same. What is noteworthy is that **TMCC** also overtakes **MC_0** with respect to target matrices, which implies the power of calibration information again. Overall, relatively low standard errors indicate the stability of these algorithms.

5. CONCLUSION

We proposed a statistical framework for multi-task learning under exponential family matrix completion framework with known calibration information. Statistical guarantee of our proposed estimator has been shown and a constant order improvement is achieved compared with existing methods. We have also shown that the proposed algorithm has a convergence rate of $O(1/k^2)$. The simulation study also shows that our proposed method has numerical benefits.

6. REFERENCES

- [1] Rich Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [2] Benjamin Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.
- [3] Raghunandan Keshavan, Andrea Montanari, and Sewoong Oh, "Matrix completion from noisy entries," *Advances in neural information processing systems*, vol. 22, 2009.
- [4] Vladimir Koltchinskii, Karim Lounici, and Alexandre B Tsybakov, "Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion," *The Annals of Statistics*, vol. 39, no. 5, pp. 2302–2329, 2011.
- [5] Sahand Negahban and Martin J Wainwright, "Restricted strong convexity and weighted matrix completion: Optimal bounds with noise," *Journal of Machine Learning Research*, vol. 13, no. May, pp. 1665–1697, 2012.
- [6] Tony Cai and Wen-Xin Zhou, "A max-norm constrained minimization approach to 1-bit matrix completion," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 3619–3647, 2013.
- [7] Mokhtar Z Alaya and Olga Klopp, "Collective matrix completion," *Journal of Machine Learning Research*, vol. 20, no. 148, pp. 1–43, 2019.
- [8] Jianqing Fan, Wenyan Gong, and Ziwei Zhu, "Generalized high-dimensional trace regression via nuclear norm regularization," *Journal of Econometrics*, vol. 212, no. 1, pp. 177–202, 2019.
- [9] Geneviève Robin, Olga Klopp, Julie Josse, Éric Moulines, and Robert Tibshirani, "Main effects and interactions in mixed and incomplete data frames," *Journal of the American Statistical Association*, vol. 115, no. 531, pp. 1292–1303, 2020.
- [10] Andrew Goldberg, Ben Recht, Junming Xu, Robert Nowak, and Jerry Zhu, "Transduction with matrix completion: Three birds with one stone," in *Advances in neural information processing systems*, 2010, pp. 757–765.
- [11] Miao Xu, Gang Niu, Bo Han, Ivor W Tsang, Zhi-Hua Zhou, and Masashi Sugiyama, "Matrix co-completion for multi-label classification with missing features and labels," *arXiv preprint arXiv:1805.09156*, 2018.
- [12] Ashkan Esmaeili, Kayhan Behdin, Mohammad Amin Fakharian, and Farokh Marvasti, "Transduction with matrix completion using smoothed rank function," *arXiv preprint arXiv:1805.07561*, 2018.
- [13] Kai-Yang Chiang, Inderjit S Dhillon, and Cho-Jui Hsieh, "Using side information to reliably learn low-rank matrices from missing and corrupted observations," *Journal of Machine Learning Research*, vol. 19, no. 1, pp. 3005–3039, 2018.
- [14] Jean-Claude Deville and Carl-Erik Särndal, "Calibration estimators in survey sampling," *Journal of the American Statistical Association*, vol. 87, no. 418, pp. 376–382, 1992.
- [15] Carl-Erik Särndal, Bengt Swensson, and Jan Wretman, *Model assisted survey sampling*, Springer Science & Business Media, 2003.
- [16] Wayne A Fuller, *Sampling statistics*, John Wiley & Sons, 2011.
- [17] F Jay Breidt and Jean D Opsomer, "Local polynomial regression estimators in survey sampling," *The Annals of Statistics*, vol. 28, no. 4, pp. 1026–1053, 2000.
- [18] Changbao Wu and Randy R Sitter, "A model-calibration approach to using complete auxiliary information from survey data," *Journal of the American Statistical Association*, vol. 96, no. 453, pp. 185–193, 2001.
- [19] Jae Kwang Kim and Mingue Park, "Calibration estimation in survey sampling," *International Statistical Review*, vol. 78, no. 1, pp. 21–39, 2010.
- [20] James Bennett and Stan Lanning, "The netflix prize," in *Proceedings of KDD cup and workshop*. Citeseer, 2007, vol. 2007, p. 35.
- [21] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [22] Shuiwang Ji and Jieping Ye, "An accelerated gradient method for trace norm minimization," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 457–464.
- [23] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *Journal of Machine Learning Research*, vol. 11, no. Aug, pp. 2287–2322, 2010.