Low-Rank Matrix Recovery with Unknown Correspondence (Supplementary Material)

Zhiwei Tang[1] **Tsung-Hui Chang**[1] **Xiaojing Ye**[2] **Hongyuan Zha**[1]

[1]The Chinese University of Hong Kong, Shenzhen
[2]Georgia State University

A PROOF FOR THE THEORETICAL RESULTS

Proof of Proposition 2.2. We denote that $a_1, .., a_{r_A}$ as the linear bases of the column space of A. We can extend them to the bases of the column space of M as $a_1, .., a_{r_A}, b_1, ..., b_{r-r_A}$. In this way, there must exists a matrix $Q \in \mathbb{R}^{r \times m_B}$ such that

$$B = [a_1, .., a_{r_A}, b_1, ..., b_{r-r_A}]Q.$$

Hence, we have

$$PB = [Pa_1, .., Pa_{r_A}, Pb_1, ..., Pb_{r-r_A}]Q.$$

Similarly, there must exists a matrix $T \in \mathbb{R}^{r_A \times m_A}$ such that

$$A = [a_1, .., a_{r_A}]T.$$

Hence, we obtain that

$$[A, PB] = [a_1, .., a_{r_A}, Pa_1, .., Pa_{r_A}, Pb_1, ..., Pb_{r-r_A}] \begin{bmatrix} T & 0 \\ 0 & Q \end{bmatrix}.$$

Now, we have

$$\begin{aligned} &\text{rank}([A, PB]) \le \text{rank}([a_1, .., a_{r_A}, Pa_1, .., Pa_{r_A}, Pb_1, ..., Pb_{r-r_A}]) \\ &\le \text{rank}([a_1, .., a_{r_A}, Pa_1, .., Pa_{r_A}]) + r - r_A \\ &= \text{rank}([a_1, .., a_{r_A}, Pa_1, .., Pa_{r_A}] \begin{bmatrix} I_{r_A} & -I_{r_A} \\ 0 & I_{r_A} \end{bmatrix}) + r - r_A \\ &\le r_A + r - r_A + \text{rank}([Pa_1 - a_1, .., Pa_{r_A} - a_{r_A}]). \end{aligned} \tag{1}$$

Now we denote the cycles in π_P with length greater than 1 as $C_1, ..., C_{\mathcal{C}(\pi_P)}$, and $\zeta_1, ..., \zeta_{n-H(\pi_p)}$ as the indexes that are not in any one of $C_1, ..., C_{\mathcal{C}(\pi_P)}$. We construct a matrix $Y \in \mathbb{R}^{(n+\mathcal{C}(\pi_P)-H(\pi_p))\times n}$ as:

$$\begin{aligned} &Y(i,j) = 1 \text{ if } j = \zeta_i \text{ else } Y(i,j) = 0, \text{ for } i = 1, ..., (n - H(\pi_p)); \\ &Y(i,j) = 1\ \forall j \in C_i, \text{ and } Y(i,j) = 0\ \forall j \notin C_i, \\ &\qquad \text{for } i = (n - H(\pi_p) + 1), ..., (n + \mathcal{C}(\pi_P) - H(\pi_p)). \end{aligned}$$

It can be verified that

$$Y(Pa_i - a_i) = 0,\ i = 1, ..., r_A.$$

We denote the null space of Y as $\text{Null}(Y) = \{x \in \mathbb{R}^n | Yx = 0\}$. From the construction of Y we can see that $\dim(\text{Null}(Y)) = H(\pi_P) - \mathcal{C}(\pi_P)$. Hence we have

$$\text{rank}([Pa_1 - a_1, .., Pa_{r_A} - a_{r_A}]) \le H(\pi_P) - \mathcal{C}(\pi_P). \tag{2}$$

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

On the other hand, we have

$$\operatorname{rank}([A, PB]) \leq \operatorname{rank}(A) + \operatorname{rank}(PB) = \operatorname{rank}(A) + \operatorname{rank}(B) = r_A + r_B. \tag{3}$$

Combining (1), (2) and (3) , we can obtain (3). □

Following the proof of Proposition 2.2, it is easy to show the similar result for the case with multiple permutation, which is summarized as the Corollary A.1

Corollary A.1. *For the matrix* $M = [A, B_1, .., B_d] \in \mathbb{R}^{n\times m}$ *with* $\operatorname{rank}(M) = r$, $\operatorname{rank}(A) = r_A$, *and* $\operatorname{rank}(B_i) = r_{B_i}$, $i = 1, ...d$, *we have* $\forall P_1, ..., P_d \in \mathcal{P}_n$,

$$\operatorname{rank}([A, P_1B_1, ..., P_dB_d]) \leq \min\{n, m, r_A + \sum_{i=1}^{d} r_{B_i}, r + \sum_{i=1}^{d} H(\pi_{P_i}) - \mathcal{C}(\pi_{P_i})\}. \tag{4}$$

Proof of Proposition 2.4. To prove Proposition 2.4, we need an important lemma on measure theory from [Halmos, 2013].

Lemma A.2. *Let* $p(x)$ *be a polynomial on* $\mathbb{R}^n$. *If there exists a* $x_0 \in \mathbb{R}^n$ *such that* $p(x_0) \neq 0$, *then the Lebesgue measure of the set* $\{x|p(x) = 0\}$ *is 0.*

$\forall P \in \mathcal{P}_n$, we define the polynomial on $\mathbb{R}^{n\times r} \otimes \mathbb{R}^{r\times m}$ as

$$p_P^r(R, E) = \sum_{S\in\mathcal{S}_r([A,PB])} \det(S)^2,$$

where $\det(\cdot)$ is the determinant of matrix, and $\mathcal{S}_r(X)$ is the set of all $r \times r$ sub-matrices in X. We denote that $r_P = \min\{2r, r + H(\pi_P) - \mathcal{C}(\pi_P)$. We can see that $\operatorname{rank}([A, PB]) \geq r_P$ if and if only $p_P^{r_P}(R, E) > 0$. Therefore, from Lemma A.2 and Proposition 2.2 we can conclude that if there exists two matrices $R_0 \in \mathbb{R}^{n\times r}$ and $E_0 \in \mathbb{R}^{r\times m}$ such that $p_P^{r_P}([R_0, E_0]) > 0$, then $\operatorname{rank}([A, PB]) = r_P$ holds with probability 1. In this way, we only need to construct such R_0 and E_0 for every $P \in \mathcal{P}_n$. For simplicity, we denote that $k = H(\pi_p) - \mathcal{C}(\pi_P)$. We will discuss how to construct such R_0 and E_0 for the two cases $0 < k \leq n - r$ and $k \geq n - r$, respectively.

(1) If $0 < k \leq n - r$:

We construct the matrix $Y \in \mathbb{R}^{(n+\mathcal{C}(\pi_P)-H(\pi_p))\times n}$ the same way with that in the proof of Proposition 2.2. Firstly, we show that Null(Y) =col$(P - I)$.

col$(P - I)$ ⊆Null(Y): We can verify that $Y(P - I) = 0$.

Null(Y) ⊆col$(P - I)$: This is equivalent to prove that Null$(P - I)$ ⊆col(Y). Now we have $Px = x, \forall x$ ∈Null$(P - I)$. It can be verified that if $Px = x$, then we must have $x(s) = x(q)$ if s and q belong to the same cycle C_i, where C_i is one of the cycles in $C_1, ..., C_{C(\pi_P)}$. By the definition of Y, we can see that $x \in$ col(Y).

Now we know that rank$(P - I)$ =dim(Null(Y)) $= k$. We denote the eigen vectors of $P - I$ with non-zero eigen values as $\phi_1, ..., \phi_k$, and the eigen vectors with zero eigen values as $\phi_{k+1}, ..., \phi_n$. Now we have $(P - I)\phi_i = \lambda_i\phi_i$ for $i = 1, ..., k$ and $(P - I)\phi_i = \lambda_i\phi_i$ for $i = k + 1, ..., n$.

We construct the matrices R_0 and E_0 as

$$\begin{aligned} R_0 &= [\phi_1 + \phi_{k+1}, \phi_{\min\{2,k\}} + \phi_{k+2}, ..., \phi_{\min\{r,k\}} + \phi_{k+r}], \\ E_0 &= [I_r, \mathbf{0}_{r\times(m_A-r)}, I_r, \mathbf{0}_{r\times(m_B-r)}]. \end{aligned}$$

Now we have

$$\begin{aligned} A &= [\phi_1 + \phi_{k+1}, \phi_{\min\{2,k\}} + \phi_{k+2}, ..., \phi_{\min\{r,k\}} + \phi_{k+r}, \mathbf{0}_{n\times(m_A-r)}], \\ B &= [\phi_1 + \phi_{k+1}, \phi_{\min\{2,k\}} + \phi_{k+2}, ..., \phi_{\min\{r,k\}} + \phi_{k+r}, \mathbf{0}_{n\times(m_B-r)}], \end{aligned}$$

since $[A, B] = R_0E_0$. Therefore, we have

$$\begin{aligned} &\operatorname{rank}([A, PB]) = \operatorname{rank}([\phi_1 + \phi_{k+1}, ..., \phi_{\min\{r,k\}} + \phi_{k+r}, \lambda_1\phi_1, ..., \lambda_{\min\{r,k\}}\phi_{\min\{r,k\}}]) \\ &= \operatorname{rank}([\phi_{k+1}, ..., \phi_{k+r}, \phi_1, .., \phi_{\min\{r,k\}}]) \\ &= r + \min\{k, r\} = \min\{2r, r + k\}. \end{aligned}$$

Now $\operatorname{rank}([A, PB]) = r_P$ by this construction of R_0 and E_0. Hence $p_P^{r_P}([R_0, E_0]) > 0$.

(2) If $k > n - r$:

We denote that the length of a cycle C as len(C), and denote the cycle with maximum length among the $C_1, ..., C_{\mathcal{C}(\pi_P)}$ as C^*. Now we have

$$\text{len}(C^*) \geq \frac{H(\pi_P)}{\mathcal{C}(\pi_P)} \geq \frac{n}{n-k} > \frac{n}{r} \geq 2r.$$

To simplify the notations, we assume that the cycle C^* permute the first j numbers, i.e.,

$$C^* = (123...(j-2)(j-1)j),$$

where $j > 2r$. We define the vector u as $u = [1, 2, 3, ..., j-2, j-1, j, 0, ..., 0]^\top \in \mathbb{R}^n$, and denote the corresponding permutation matrix to C^* as $P_* \in \mathcal{P}_n$. We construct the matrices R_0 and E_0 as

$$\begin{aligned} R_0 &= \begin{bmatrix} u & P_*^2 u & \dots & P_*^{2r-2} u \end{bmatrix}, \\ E_0 &= [I_r, \mathbf{0}_{r\times(m_A - r)}, I_r, \mathbf{0}_{r\times(m_B - r)}]. \end{aligned}$$

Now we have

$$\begin{aligned} A &= [u, P_*^2 u, \dots, P_*^{2r-2} u, \mathbf{0}_{n\times(m_A - r)}], \\ B &= [u, P_*^2 u, \dots, P_*^{2r-2} u, \mathbf{0}_{n\times(m_B - r)}]. \end{aligned}$$

Therefore, we have

$$\operatorname{rank}([A, PB]) = \operatorname{rank}([u, P_* u, \dots, P_*^{2r-1} u]) = 2r,$$

because now $[u, P_* u, \dots, P_*^{2r-1} u]$ is a circulant matrix. Now $\operatorname{rank}([A, PB]) = r_P = 2r$ by this construction of R_0 and E_0. Hence $p_P^{r_P}([R_0, E_0]) > 0$. □

Proof of Proposition 2.6.. To prove Proposition 2.6, we need to derive a series results. We first start with a very important inequality w.r.t nuclear norm.

Proposition A.3. *Let P be a permutation matrix, then,*

$$\|A\|_* + \|B\|_* \geq \|[A, PB]\|_* \geq \frac{\|A\|_* + \|B\|_*}{\|[U_A V_A^\top, P U_B V_B^\top]\|} \geq \frac{\|A\|_* + \|B\|_*}{\sqrt{2}}. \tag{5}$$

Based on (5), the general idea is that under the Assumptions 2.5, we will have $\|M\|_* \approx \frac{\|A\|_* + \|B\|_*}{\sqrt{2}}$ and $\|[U_A V_A^\top, P U_B V_B^\top]\| \to 1$ as $H(\pi_P)$ increases.

Firstly, we show that under the Assumptions 2.5, the nuclear norm of the original matrix M will reach the lower bound in (5) approximately, which is summarized as Lemma A.4.

Lemma A.4. *Under the Assumptions 2.5, we have*

$$\|M\|_* \leq (\|A\|_* + \|B\|_*)/\sqrt{2} + (\sqrt{2} + 1)\epsilon_1 r + \epsilon_2 \max\{\|A\|_*, \|B\|_*\}. \tag{6}$$

Then, we show that under the Assumptions 2.5, $\|[U_A V_A^\top, P U_B V_B^\top]\| \to 1$ as $H(\pi_P)$ increases, which is summarized as Lemma A.5.

Lemma A.5. *Under the Assumptions 2.5, we have*

$$\|[U_A V_A^\top, P U_B V_B^\top]\| \leq \sqrt{2 - H(\pi_P)\epsilon_3^2/2} + \sqrt{T}\epsilon_2. \tag{7}$$

Finally, we need a classical result on the tail bound for the operator norm of Gaussian matrix, whose proof can be found in [Wainwright, 2019].

Lemma A.6. *Consider the random matrix $W \in \mathbb{R}^{n\times m}$ whose elements follow $\mathcal{N}(0,\sigma^2)$ i.i.d. For any $\delta > 0$, we have*

$$\|W\| \leq \sqrt{L}(2+\delta)\sigma \tag{8}$$

holds with probability greater than $1 - 2\exp\{\frac{-L\delta^2}{2}\}$, where $L = \max\{n, m\}$.

Based on Lemma A.6, we have

$$\|W\|_* \leq L\|W\| \leq \sqrt{DL\sigma}$$

holds with probability greater than $1 - 2\exp\{-\frac{D}{8L\sigma}\}$.

From Proposition A.3, Lemma A.4 and Lemma A.5 we can know that, for any $P \in \mathcal{P}_n$ with $H(\pi_P)$ satisfies that

$$\frac{D}{\sqrt{2 - \frac{H(\pi_p)\epsilon_3}{2}} + \sqrt{T}\epsilon_2} - \|W\|_* > \frac{D}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2 N + \|W\|_*,$$

we must have

$$\begin{aligned}\|A_o, PB_o\|_* &\geq \|A, PB\|_* - \|W\|_* \\ &\geq \frac{D}{\sqrt{2 - \frac{H(\pi_p)\epsilon_3^2}{2}} + \sqrt{T}\epsilon_2} - \|W\|_* \\ &> \frac{D}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2 N + \|W\|_* \\ &\geq \|A, B\|_* + \|W\|_* \geq \|A_o, B_o\|_*.\end{aligned}$$

Therefore, with probability greater than $1 - 2\exp\{-\frac{D}{8L\sigma}\}$, if $H(\pi_P)$ satisfies that

$$\frac{D}{\sqrt{2 - \frac{H(\pi_p)\epsilon_3^2}{2}} + \sqrt{T}\epsilon_2} > \frac{D}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2 N + 2\sqrt{DL\sigma}, \tag{@}$$

we have $\|A_o, PB_o\| > \|A_o, B_o\|_*$. Now we simplify (@) as

$$\begin{aligned}&\frac{D}{\sqrt{2 - \frac{H(\pi_p)\epsilon_3^2}{2}} + \sqrt{T}\epsilon_2} > \frac{D}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2 N + 2\sqrt{DL\sigma} \\ \Leftrightarrow &\sqrt{2 - \frac{H(\pi_p)\epsilon_3^2}{2}} < \frac{\sqrt{2}D}{D + (\sqrt{2}+2)\epsilon_1 r + \sqrt{2}\epsilon_2 N + 2\sqrt{2DL\sigma}} - \sqrt{T}\epsilon_2.\end{aligned}$$

It can be verified that

$$\frac{\sqrt{2}D}{D + (\sqrt{2}+2)\epsilon_1 r + \sqrt{2}\epsilon_2 N + 2\sqrt{2DL\sigma}} - \sqrt{T}\epsilon_2 > 0$$

from the condition on ϵ_1, ϵ_2 and σ.

Therefore, we have

$$\begin{aligned}&\sqrt{2 - \frac{H(\pi_p)\epsilon_3^2}{2}} < \frac{\sqrt{2}D}{D + (\sqrt{2}+2)\epsilon_1 r + \sqrt{2}\epsilon_2 N + 2\sqrt{2DL\sigma}} - \sqrt{T}\epsilon_2 \\ \Leftrightarrow &H(\pi_P) > \frac{2}{\epsilon_3^2}\left(2 - (\frac{\sqrt{2}D}{D + (\sqrt{2}+2)\epsilon_1 r + \sqrt{2}\epsilon_2 N + 2\sqrt{2DL\sigma}} - \sqrt{T}\epsilon_2)^2\right).\end{aligned}$$

Since P^* is the optimal solution to (5), we must have

$$\|[A_o, P^*\tilde{P}B_o]\|_* \leq \|[A_o, B_o]\|_*.$$

Besides, $P^*\tilde{P}$ is also a permutation matrix, we denote its corresponding permutation as $\hat{\pi}$. Now we have

$$d_H(\pi_*, \tilde{\pi}) = H(\hat{\pi}) \leq \frac{2}{\epsilon_3^2}\Bigg(2 - (\frac{\sqrt{2}D}{D + (\sqrt{2}+2)\epsilon_1 r + \sqrt{2}\epsilon_2 N + 2\sqrt{2DL\sigma}} - \sqrt{T}\epsilon_2)^2\Bigg).$$

□

The proof to the auxiliary results used in the proof of Proposition 2.6 are provided below.

Proof of Proposition A.3. Since $\|\cdot\|_*$ is a norm, we have

$$\|[A, PB]\|_* = \|[A, \mathbf{0}] + [\mathbf{0}, PB]\|_* \leq \|A\|_* + \|PB\|_* = \|A\|_* + \|B\|_*.$$

Then since $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$, we have

$$\begin{aligned}
\|[A, PB]\|_* &= \sup_{\|Q\|\leq 1} \langle [A, PB], Q\rangle \\
&\geq \langle [A, PB], \frac{[U_A V_A^\top, PU_B V_B^\top]}{\|[U_A V_A^\top, PU_B V_B^\top]\|}\rangle \\
&= \frac{\|A\|_* + \|B\|_*}{\|[U_A V_A^\top, PU_B V_B^\top]\|}.
\end{aligned}$$

Finally, we have

$$\begin{aligned}
\|[U_A V_A^\top, PU_B V_B^\top]\| &= \sup_{\substack{x\in\mathbb{R}^m \\ \|x\|\leq 1}} \|[U_A V_A^\top, PU_B V_B^\top]x\| \\
&= \sup_{\substack{x_1\in\mathbb{R}^{m_A}, x_2\in\mathbb{R}^{m_B} \\ \|[x_1^\top, x_2^\top]\|\leq 1}} \|[U_A V_A^\top x_1, PU_B V_B^\top x_2]\| \\
&\leq \sup_{\substack{x_1\in\mathbb{R}^{m_A}, x_2\in\mathbb{R}^{m_B} \\ \|[x_1^\top, x_2^\top]\|\leq 1}} \|U_A V_A^\top x_1\| + \|PU_B V_B^\top x_2\| \\
&\leq \sup_{\substack{x_1\in\mathbb{R}^{m_A}, x_2\in\mathbb{R}^{m_B} \\ \|[x_1^\top, x_2^\top]\|\leq 1}} \|x_1\| + \|x_2\| = \sqrt{2}.
\end{aligned}$$

□

Proof of Lemma A.4. If $r_A \geq r_B$, we have

$$\begin{aligned}
\|M\|_* &= \|[U_A\Sigma_A V_A^\top, U_B\Sigma_B V_B^\top]\|_* \\
&= \|[U_A\Sigma_A V_A^\top, [u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}]\Sigma_B V_B^\top] + \\
&\qquad [\mathbf{0}, [u_A^1 - u_B^1, ..., u_A^T - u_B^T, u_B^{T+1}, ..., u_B^r]\Sigma_B V_B^\top]\|_* \\
&\leq \|[U_A\Sigma_A V_A^\top, [u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}]\Sigma_B V_B^\top]\|_* + \\
&\qquad \|[u_A^1 - u_B^1, ..., u_A^T - u_B^T, u_B^{T+1}, ..., u_B^r]\Sigma_B V_B^\top\|_* \\
&\leq \|[U_A\Sigma_A V_A^\top, [u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}]\Sigma_B V_B^\top]\|_* + \epsilon_2\|B\|_* \\
&= \|[U_A\Sigma_A V_A^\top, U_A\Sigma_B V_B^\top]\|_* + \epsilon_2\|B\|_*. \qquad (*)
\end{aligned}$$

We denote that $trace(\cdot)$ as the trace of matrix. One property of nuclear norm is

$$\|A\|_* = trace(\sqrt{AA^\top}).$$

Then we have

$$\begin{aligned}
\|[U_A\Sigma_A V_A^\top, U_A\Sigma_B V_B^\top]\|_* &= trace(\sqrt{U_A(\Sigma_A^2+\Sigma_B^2)U_A^\top}) \\
&= \sum_{i=1}^{r}\sqrt{(\sigma_A^i)^2+(\sigma_B^i)^2} \\
&\le \sum_{i=1}^{r}\frac{\sigma_A^i+\sigma_B^i}{\sqrt{2}} + (\sqrt{(\sigma_A^i)^2+(\sigma_B^i)^2} - \frac{\sigma_A^i+\sigma_B^i}{\sqrt{2}}) \\
&\le \sum_{i=1}^{r}\frac{\sigma_A^i+\sigma_B^i}{\sqrt{2}} + (\sqrt{(\sigma_A^i)^2+(\sigma_A^i+\epsilon_1)^2} - \frac{2\sigma_A^i-\epsilon_1}{\sqrt{2}}) \\
&\le \frac{\sqrt{2}\epsilon_1 r}{2} + \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + \\
&\qquad \sum_{i=1}^{r}\frac{2\sigma_A^i\epsilon_1+\epsilon_1^2}{\sqrt{2(\sigma_A^i)^2+2\sigma_A^i\epsilon_1+\epsilon_1^2}+\sqrt{2(\sigma_A^i)^2}} \\
&\le \frac{\sqrt{2}\epsilon_1 r}{2} + \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + \sum_{i=1}^{r}\frac{\sqrt{2}\epsilon_1}{2} + \epsilon_1 \\
&= \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r. \qquad (**)
\end{aligned}$$

Combining (*) and (**), we have

$$\|[A,B]\|_* \le \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2\|B\|_*.$$

Similarly, if $r_B \ge r_A$, we have

$$\|[A,B]\|_* \le \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2\|A\|_*.$$

Combining them together, we have

$$\|[A,B]\|_* \le \frac{\|A\|_*+\|B\|_*}{\sqrt{2}} + (\sqrt{2}+1)\epsilon_1 r + \epsilon_2\max\{\|A\|_*,\|B\|_*\}.$$

□

Proof pf Lemma A.5. Firstly, if $r_A \ge r_B$ we have

$$\begin{aligned}
\|[U_AV_A^\top, PU_BV_B^\top]\| &= \|[U_AV_A^\top, P[u_A^1,...,u_A^T,\mathbf{0},...,\mathbf{0}]V_B^\top]\| + \\
&\qquad \|[0, P[u_B^1-u_A^1,...,u_B^T-u_A^T,\mathbf{0},...,\mathbf{0}]V_B^\top]\| \\
&\le \|[U_AV_A^\top, P[u_A^1,...,u_A^T,\mathbf{0},...,\mathbf{0}]V_B^\top]\| + \sqrt{T}\epsilon_2. \qquad (***)
\end{aligned}$$

To simplify the notations, we denote that $k = H(\pi_P)$ and assume that π_P permutes the indexes $(1,...,k)$ into $(\zeta_1,...,\zeta_k)$. Now we have

$$\langle u_A^i, Pu_A^i\rangle = \sum_{i=1}^{k} u_A^i(i)u_A^i(\zeta_i) + \sum_{i=k+1}^{n}(u_A^i(i))^2,$$

and

$$
\begin{aligned}
|\sum_{i=1}^{k} u_A^i(i) u_A^i(\zeta_i)| &\leq \sum_{i=1}^{k} |u_A^i(i) u_A^i(\zeta_i)| \\
&= \sum_{i=1}^{k} \frac{(u_A^i(i))^2 + (u_A^i(\zeta_i))^2}{2} - (\frac{(u_A^i(i))^2 + (u_A^i(\zeta_i))^2}{2} - |u_A^i(i) u_A^i(\zeta_i)|) \\
&\leq \sum_{i=1}^{k} (u_A^i(i))^2 - (\frac{(u_A^i(i))^2 + (|u_A^i(i)| - \epsilon_3)^2}{2} - |u_A^i(i)|(|u_A^i(i)| + \epsilon_3)) \\
&= \sum_{i=1}^{k} (u_A^i(i))^2 - (\frac{\epsilon_3^2}{2} + 2|u_A^i(i)|\epsilon_3) \leq \sum_{i=1}^{k} (u_A^i(i))^2 - \frac{\epsilon_3^2}{2}.
\end{aligned}
$$

Hence we must have

$$
|\langle u_A^i, P u_A^i \rangle| \leq 1 - \frac{k\epsilon_3^2}{2}.
$$

Therefore, we have

$$
\begin{aligned}
\delta(U_A, P) &\overset{\text{def.}}{=} \max_{\substack{x, y \in \mathbb{R}^T, \\ \|x\|=1, \|y\|=1}} \langle [u_A^1, ..., u_A^T] x, [P u_A^1, ..., P u_A^T] y \rangle \\
&= \max_{\substack{x, y \in \mathbb{R}^T, \\ \|x\|=1, \|y\|=1}} \sum_{i=1}^{T} x(i) y(i) \langle u_A^i, P u_A^i \rangle \\
&\leq \max_{\substack{x, y \in \mathbb{R}^T, \\ \|x\|=1, \|y\|=1}} (1 - \frac{k\epsilon_3^2}{2}) \sum_{i=1}^{T} x(i) y(i) \\
&= 1 - \frac{k\epsilon_3^2}{2}.
\end{aligned}
$$

Now we have,

$$
\begin{aligned}
\|[U_A V_A^\top, P[u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}] V_B^\top]\| &= \sup_{\substack{x \in \mathbb{R}^n, \\ \|x\|=1}} \|[U_A V_A^\top, P[u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}] V_B^\top] x\| \\
\leq \sup_{\substack{x_1 \in \mathbb{R}^{m_A}, x_2 \in \mathbb{R}^{m_B} \\ \|[x_1^\top, x_2^\top]\| \leq 1}} & \sqrt{1 + \langle U_A V_A^\top x_1, P[u_A^1, ..., u_A^T, \mathbf{0}, ..., \mathbf{0}] V_B^\top x_2 \rangle} \\
\leq \sup_{\substack{x_1 \in \mathbb{R}^{m_A}, x_2 \in \mathbb{R}^{m_B} \\ \|[x_1^\top, x_2^\top]\| \leq 1}} & \sqrt{1 + \delta(U_A, P) \|x_1\| \|x_2\|} \leq \sqrt{2 - \frac{k\epsilon_3^2}{2}}.. \qquad (****)
\end{aligned}
$$

Combining (***) and (****), we have

$$
\|[U_A V_A^\top, P U_B V_B^\top]\| \leq \sqrt{2 - \frac{k\epsilon_3^2}{2}} + \sqrt{T}\epsilon_2.
$$

The proof is similar for the case $r_B \geq r_A$. □

B DISCUSSION ON ASSUMPTION 2.5

When ϵ_1 in Assumption 2.5 is sufficiently large: Consider $A = \sigma_A^1 u, B = \sigma_B^1 u, u \in \mathbb{R}^n$. If $\epsilon_1 > kD$ $(k < 1)$, according to inequality (6), for any permutation matrix P, we have $| \, \|[A, PB]\|_* - \|[A, B]\|_* \, | \leq \frac{1-k}{1+k} \|[A, B]\|_*$. Therefore, the larger the ϵ_1 is, the harder to distinguish $[A, PB]$ and $[A, B]$ through nuclear norm, especially with the perturbation of additive noise.

When ϵ_2 in Assumption 2.6 is sufficiently large: Consider $A = u_A \in \mathbb{R}^n$, $B = u_B \in \mathbb{R}^n$ and $\sigma = 0$, where $\|u_A\| = \|u_B\| = 1$. Let $\epsilon_2 = \|u_A - u_B\|$, we can obtain $\|[A, B]\|_* = \sqrt{2 + 2\sqrt{1 - (1 - \frac{\epsilon_2^2}{2})^2}}$. In this case, we can see that $\|[A, B]\|_*$ is in fact an increasing function of ϵ_2. Therefore, for any permutation matrix $P \in \mathcal{P}_n^{\epsilon_2} = \{S \in \mathcal{P}_n \mid \|u_A - Su_B\| \leq \epsilon_2\}$, we have $\|[A, PB]\|_* \leq \|[A, B]\|_*$, i.e., it is impossible to recover the original matrix through nuclear norm minimization. Especially, in this case, when $\epsilon_2 = \sqrt{2}$, the set $\mathcal{P}_n^{\epsilon_2} = \mathcal{P}_n$.

When $\epsilon_3 = 0$ in Assumption 2.7: Consider $A = B = u \in \mathbb{R}^n$ and $\sigma = 0$. We first define n set $S(i) = \{j \mid u(i) = u(j)\}$ for $i = 1, ..., n$. We let $S^* = \arg\max_{S(i)} \#|S(i)|$. For any permutation P that only permutes the indexes in S^* and $H(\pi_P) = \#|S^*| > 0$, we have $\|[A, B]\|_* = \|[A, PB]\|_*$, i.e., it is impossible to distinguish the permuted matrix and the original matrix through nuclear norm.

C ASYMPTOTIC BEHAVIOR OF PROPOSITION 2.8.

In this section, we will discuss about the asymptotic behavior ($n \to \infty$) of the error bound in Proposition 2.8.

We start with a simple observation: Without $\epsilon_1 \to 0, \epsilon_2 \to 0, \sigma \to 0$, the original matrix will be impossible to recover by minimizing nuclear norm for sufficient large n. This is also reflected in the error bound of Proposition 2.8, where the right hand side of (10) could become trivial, i.e., larger than n, when n is sufficiently large.

We provide a simple example to validate this observation. Suppose that the original matrix is $M = [u, u] + W$, where the elements of W follow $\mathcal{N}(0, \sigma^2)$ and $u \in \mathbb{R}^n$ is a random vector whose elements are i.i.d. following the uniform distribution on $[0, 1]$. From the result in [David and Nagaraja, 2004], p. 135, we know that

$$\mathbb{E}[\max_{i \neq j} |u(i) - u(j)|] \approx O(n^{-1} \log(n)).$$

Therefore, we can construct a permutation matrix $P \in \mathcal{P}_n$ with $H(\pi_P) = n$, such that the following inequality holds with high probability,

$$|\|[u, Pu]\|_* - \|[u, u]\|_*| \leq \|Pu - u\|_2 = O(n^{-\frac{1}{2}} \log(n)).$$

On the other hand, from Lemma A.6 we can know that $\|W\|_* \approx O(\sigma n)$ with high probability. Now if we need that $\|[u, Pu] + W\|_* > \|[u, u] + W\|_*$, we at least require that $\sigma = o(n^{-\frac{3}{2}} \log(n))$. Otherwise, it will be impossible to distinguish the matrices $[u, Pu] + W$ and $[u, u] + W$ through the value of nuclear norm.

Finally, for this simple example, we have $\epsilon_1 = \epsilon_2 = 0$. Besides, from [David and Nagaraja, 2004], we can also know that ϵ_3 is at most $O(n^{-\frac{3}{2}})$ with high probability. With a simple calculattion, we can find that the error bound in Proposition 2.8 is at least $O(n^{\frac{5}{2}} \sigma^{\frac{1}{2}})$. Therefore, in this example, we at least require that $\sigma = o(n^{-5})$ to guarantee a constant error bound for arbitrary n.

D DUAL PROBLEM OF (15)

To simplify the notation, we denote the primal problem as

$$\underset{P \in \Pi(\mathbf{1}_n, \mathbf{1}_n)}{\text{minimize}} \langle C, P \rangle + \epsilon \mathcal{H}(P).$$

We define two dual variables $\alpha, \beta \in \mathbb{R}^n$. The Lagrangian function is

$$L(P, \alpha, \beta) = \langle C, P \rangle + \epsilon \langle \log P - \mathbf{1}_{n \times n}, P \rangle + \langle \mathbf{1}_n - P\mathbf{1}_n, \alpha \rangle + \left\langle \mathbf{1}_n - P^T \mathbf{1}_n, \beta \right\rangle. \tag{9}$$

Now we minimize the Lagrangian function w.r.t P (We note that $\mathcal{H}(P)$ implicitly imposes that $P \in \mathbb{R}_+^{n \times n}$). From the first-order necessary condition of unconstrainted optimization, we have

$$\begin{aligned} C - &\alpha \oplus \beta + \epsilon \log(P) = 0, \\ &\Downarrow \\ P = &\exp\left\{ \frac{\alpha \oplus \beta - C}{\epsilon} \right\}. \end{aligned} \tag{10}$$

Substituting it into the Lagrangian function (9) we have the dual objective

$$q(\alpha,\beta) = \min_P L(P,\alpha,\beta) = \langle \mathbf{1}_n, \alpha\rangle + \langle \mathbf{1}_n, \beta\rangle - \epsilon\Big\langle \mathbf{1}_{n\times n}, \exp\Big\{\frac{\alpha \oplus \beta - C}{\epsilon}\Big\}\Big\rangle.$$

Therefore the dual problem is

$$\max_{\alpha,\beta\in\mathbb{R}^n} \langle \mathbf{1}_n, \alpha\rangle + \langle \mathbf{1}_n, \beta\rangle - \epsilon\Big\langle \mathbf{1}_{n\times n}, \exp\Big\{\frac{\alpha \oplus \beta - C}{\epsilon}\Big\}\Big\rangle. \tag{11}$$

We can recover the primal solution P from the dual solution α, β via (10).

E A STABLE IMPLEMENTATION FOR SINKHORN ALGORITHM

The Sinkhorn algorithm [Peyré et al., 2019] are often used to solve the dual problem (11), and the standard form of it reads

$$p^{(t+1)} \leftarrow \frac{\mathbf{1}_n}{Kq^{(t)}} \text{ and } q^{(t+1)} \leftarrow \frac{\mathbf{1}_n}{K^\top p^{(t+1)}},$$

where $K = \exp\Big\{\frac{\alpha\oplus\beta-C}{\epsilon}\Big\}$, and $p = \exp(\frac{\alpha}{\epsilon})$, $q = \exp(\frac{\beta}{\epsilon})$. If we adopt a small ϵ, the elements of K can overflow to infinity or zero, which causes a numerical issue. We can remedy this by using a different implementation from [Peyré et al., 2019].

$$\alpha^{(t+1)} \leftarrow \text{Min}_\epsilon^{\text{row}}(C - \alpha^{(t)} \oplus \beta^{(t)}) + \alpha^{(t)},$$
$$\beta^{(t+1)} \leftarrow \text{Min}_\epsilon^{\text{col}}(C - \alpha^{(t+1)} \oplus \beta^{(t)}) + \beta^{(t)},$$

where for any $A \in \mathbb{R}^{n\times m}$, we define the operator $\text{Min}_\epsilon^{\text{row}}$ and $\text{Min}_\epsilon^{\text{col}}$ as

$$\text{Min}_\varepsilon^{\text{row}}(\mathbf{A}) \overset{\text{def.}}{=} (\min{}_\varepsilon \mathbf{A}(i,\cdot))_i \in \mathbb{R}^n,$$
$$\text{Min}_\varepsilon^{\text{col}}(\mathbf{A}) \overset{\text{def.}}{=} (\min{}_\varepsilon \mathbf{A}(\cdot,j))_j \in \mathbb{R}^m,$$

and for any vector $z = [z_1, ..., z_n]^\top \in \mathbb{R}^n$, we denote

$$\min{}_\epsilon z \overset{\text{def.}}{=} \min_i z_i - \epsilon \log \sum_j e^{-(z_j - \min_i z_i)/\epsilon}$$

as the ϵ-soft minimum for the elements of z.

F RELATIONSHIP BETWEEN M^{3}O AND THE SOFT-IMPUTE ALGORITHM

Soft-Impute algorithm [Mazumder et al., 2010] is a classical algorithm for matrix completion. Specifically, it tries to solve the nuclear norm regularized problem

$$\underset{\widehat{M}}{\text{minimize}}\ \frac{1}{2}\left\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(\widehat{M})\right\|_F^2 + \lambda\left\|\widehat{M}\right\|_*. \tag{12}$$

Soft-Impute is a simple iterative algorithm with the following two steps:

$$\widehat{X} \leftarrow \mathcal{P}_\Omega(X) + \mathcal{P}_\Omega^\perp(\widehat{M}), \tag{13}$$
$$\widehat{M} \leftarrow \text{prox}_{\lambda\|\cdot\|_*}(\widehat{X}) = U\mathcal{S}_\lambda(D)V^\top, \tag{14}$$

where $\widehat{X} = UDV^\top$ denotes the singular value decomposition of $\widehat{X}$, and $\mathcal{P}_\Omega^\perp$ is the operator that selects entries whose indexes are not belonging to Ω. Here $\mathcal{S}_\lambda$ is the soft-thresholding operator that operates element-wise on the diagonal matrix D, i.e., replacing D_{ii} with $(D_{ii} - \lambda)_+$.

Algorithm 1 M^{3}O-AS-DE

```
Input: stepsize parameter $\omega$, number of correspondence $d$, number of iterations $N$, number of tolerance steps $K$, initial entropy coefficient $\epsilon$, tolerance $\varepsilon$, observation matrix $M_o = [A_o, B_o^1, ..., B_o^d]$, initial matrix $\widehat{M} = [\widehat{M}_A, \widehat{M}_{B_1}, ..., \widehat{M}_{B_d}]$, nuclear norm coefficient $\lambda$, the set of observable indexes $\Omega$.
Initialize $\widehat{P}^l_{\text{new}} = \mathbf{0}_{n\times n}$ for $l = 1, ..., d$.
for $k = 1 : N$ do
  for $l = 1 : d$ in parallel do
    $\widehat{P}^l_{\text{old}} = \widehat{P}^l_{\text{new}}$.
    $\hat{\alpha}^l = \hat{\beta}^l = \mathbf{1}_n$.
    Compute the partial pairwise cost matrix $C(\widehat{M}_{B_l})$.
    repeat
      $\hat{\alpha}^l \leftarrow \text{Min}_\epsilon^{\text{row}}(C(\widehat{M}_{B_l}) - \hat{\alpha}^l \oplus \hat{\beta}^l) + \hat{\alpha}^l$.
      $\hat{\beta}^l \leftarrow \text{Min}_\epsilon^{\text{col}}(C(\widehat{M}_{B_l}) - \hat{\alpha}^l \oplus \hat{\beta}^l) + \hat{\beta}^l$.
      $\widehat{P}^l_{\text{new}} \leftarrow \exp\left\{\frac{\hat{\alpha}^l \oplus \hat{\beta}^l - C(\widehat{M}_{B_l})}{\epsilon}\right\}$.
    until $\frac{1}{\sqrt{n}}\left\|\mathbf{1}_n^\top \widehat{P} - \mathbf{1}_n^\top\right\|_2 \leq \varepsilon$
    Compute the stepsize $\rho_l$ as discussed in Section 3.
    $\widehat{M}_{B_l} \leftarrow \widehat{M}_{B_l} - \rho_l \nabla_{\widehat{M}} F_\epsilon^l(\widehat{M}_{B_l}, \alpha^l, \beta^l)$, where

        $F_\epsilon^l(\widehat{M}_{B_l}, \alpha, \beta) \overset{\text{def.}}{=} \langle \mathbf{1}_n, \alpha\rangle + \langle \mathbf{1}_n, \beta\rangle - \epsilon\left\langle \mathbf{1}_{n\times n}, \exp\left\{\frac{\alpha \oplus \beta - C_\Omega(\widehat{M}_{B_l})}{\epsilon}\right\}\right\rangle.$

  end for
  $\widehat{M}_A \leftarrow \mathcal{P}_\Omega(A) + \mathcal{P}_\Omega^\perp(\widehat{M}_A)$.
  $\widehat{M} \leftarrow \text{prox}_{\lambda\|\cdot\|_*}([\widehat{M}_A, \hat{M}_{B_1}, ..., \widehat{M}_{B_d}]))$.
  if the objective value is not improved over $K$ steps then
    $\epsilon \leftarrow \epsilon/2$.
  end if
end for
```

Consider the partial observation extension. For the M^{3}O algorithm, if an exact permutation matrix is obtained, i.e., $\widehat{P} = \exp\left\{\frac{\alpha^* \oplus \beta^* - C(\widehat{M}_B)}{\epsilon}\right\} \in \mathcal{P}_n$, it is easy to verify that the the gradient in Algorithm 1 has the following form,

$$\nabla_{\widehat{M}} F_\epsilon(\widehat{M}, \alpha^*, \beta^*) = 2(\mathcal{P}_\Omega(\widehat{M}) - \mathcal{P}_\Omega([A, \widehat{P}\tilde{B}])).$$

In this way, if we adopts $\rho_k = 0.5$, the proximal gradient update becomes

$$\widehat{M}^{\text{k+1}} \leftarrow \text{prox}_{\lambda\|\cdot\|_*}(\mathcal{P}_\Omega([A, \widehat{P}\tilde{B}]) + \mathcal{P}_\Omega^\perp(\widehat{M}^k)).$$

In practice, $\widehat{P}$ often becomes very close to an exact permutation matrix and the stepsize often reaches the upper bound 0.5, when the algorithm is close to convergence. In this scenario, our algorithm becomes equivalent to the Soft-Impute algorithm. Therefore, we adopt the Soft-Impute algorithm as a baseline method for matrix completion without correspondence issue.

G M^{3}O-AS-DE FOR THE D-CORRESPONDENCE PROBLEM

In this section, we summarize our proposed algorithm M^{3}O-AS-DE for the general d-correspondence problem (18) in Algorithm 1. To determinate the stop of the Max-Oracle, we find that the criterion

$$\frac{1}{\sqrt{n}}\left\|\mathbf{1}_n^\top \widehat{P} - \mathbf{1}_n^\top\right\|_2 \leq \varepsilon$$

works well in practice, which serves as a good indicator for the ε-good optimality.

Algorithm 2 Baseline

```
Input: number of iterations $N$, number of Proximal Gradient iterations $N_p$, tolerance $\varepsilon$, observation matrix $M_o = [A_o, B_o^1, ..., B_o^d]$, initial matrix $\widehat{M} = [\widehat{M}_A, \widehat{M}_{B_1}, ..., \widehat{M}_{B_d}]$, nuclear norm coefficient $\lambda$, partial observation operator $\mathcal{P}_\Omega$.
for $k = 1 : N$ do
    for $l = 1 : d$ in parallel do
        Solving the inner problem of (15) for $\hat{P}^l$ up to tolerance $\varepsilon$ via Hungarian algorithm.
    end for
    $X \leftarrow [A_o, \hat{P}^1 B_o^1, ..., \hat{P}^d B_o^d]$.
    for $i = 1 : N_p$ do
        $\hat{X} \leftarrow \mathcal{P}_\Omega(X) + \mathcal{P}_\Omega^\perp(\hat{M})$.
        $\hat{M} \leftarrow \text{prox}_{\lambda\|\cdot\|_*}(\hat{X})$.
    end for
end for
```

H THE BASELINE ALGORITHM

We also extend the Baseline algorithm to a similar d-correspondence problem as (18). Specifically, the extended Baseline algorithm tries to solve the unsmoothed problem

$$\begin{aligned} &\min_{\widehat{M}} \min_{P_1,...,P_d} \left\|\mathcal{P}_\Omega(A_o) - \mathcal{P}_\Omega(\widehat{M}_A)\right\|_F^2 + \sum_{l=1}^{d} \langle C(\widehat{M}_{B_l}), P_l \rangle + \lambda \left\|\widehat{M}\right\|_*, \\ &\qquad\qquad \text{s.t. } P_l \in \mathcal{P}_n, \ \text{for } l = 1, ..., d. \end{aligned} \tag{15}$$

We summarize the algorithm in Algorithm 2.

I THE MUS ALGORITHM

In this section, we provide details for the MUS algorithm discussed in the Section 4. Firstly, inspired by [Yao et al., 2021], we first transform the MRUC problem, i.e, to recover $[A, B]$ from $[A, \tilde{P}B]$, into a MUS problem as follows,

$$\min_{P \in \mathcal{P}_n, W \in \mathbb{R}^{m_B \times m_A}} \|A - P\tilde{P}BW\|_F^2. \tag{16}$$

Then, for the scenario without multiple correspondence and missing values, we adopt the algorithm in [Zhang and Li, 2020] to solve (16).

To extend it into the d-correspondence problem considered by (18), we adopt tow simple procedures. Specifically, to deal with the missing value, we first fill in the missing entries of each submatrices using the Soft-Impute algorithm. As for the multiple correspondence issue, we simply run the MUS algorithm in multiple times. For example, if we want solve the d-correspondence problem, we typically apply the MUS algorithm to the following series of problems in turn,

$$\min_{P \in \mathcal{P}_n, W \in \mathbb{R}^{m_B \times m_A}} \|A_o - PB_o^l W\|_F^2, \ l = 1, ..., d.$$

J DISCUSSION ON US, MUS AND MRUC

In this section, we wil discuss about the difference and similarity among the US problem, MUS problem and our MRUC problem. Specifically, we wish to answer the following question:

- Why MUS algorithms, like the one in [Zhang and Li, 2020], are more suitable to be adapted for our MRUC problem than those US algorithms like AIEM [Tsakiris et al., 2020] and CCV-Min [Peng and Tsakiris, 2020] that adopted by [Yao et al., 2021]?

For this question, we note that the MUS problem (2) can be solved by US algorithms, because we can treat it as m_1 independent US problems just as what [Yao et al., 2021] did. In this way, we can view the key difference between our adapted MUS algorithm and the method proposed by [Yao et al., 2021] as whether to leverage the prior knowledge that multiple response vectors are shuffled by the same permutation, i.e., to recover the permutation for m_1 responses jointly or independently. Theoretically, it has been well studied in the works [Zhang and Li, 2020, Pananjady et al., 2017a, Slawski et al., 2020b,a] that one can resist stronger noise and estimate the ground-truth permutation better if we know that more columns are shuffled by the same permutation. We remark that this phenomenon is not a contradiction to the experiment results in [Yao et al., 2021], as they only reported the residual error for vector recovery instead of permutation recovery.

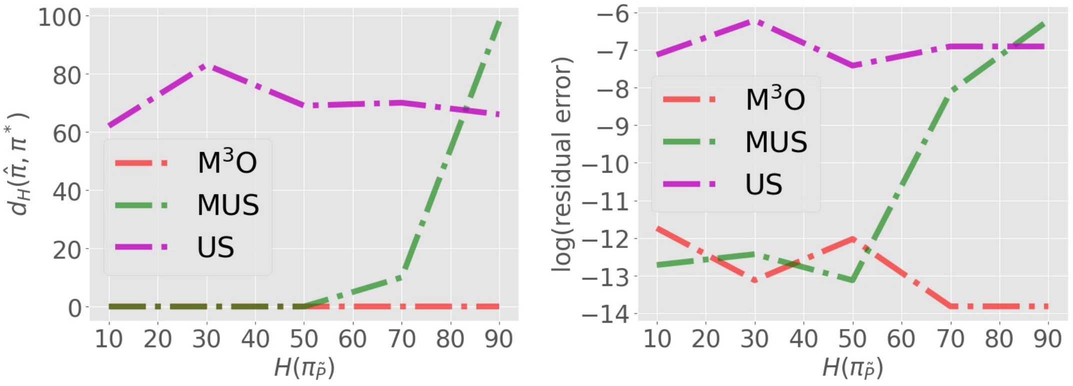
We also conduct our own experiment to corroborate our previous discussion. We generate the synthetic matrix $M_o = [A, \tilde{P}B]$ in the same way with the experiment in Figure 2. Here we use the full matrix M_o, i.e., no missing values, and hence the MRUC problem is now barely distinguishable to the MUS problem. We use the following three kinds of algorithm for comparison:

1. MRUC: Our proposed algorithm M^{3}O.
2. US: CCV-min algorithm[1] used in [Yao et al., 2021], which is shown to be the state-of-the-art US algorithm.
3. MUS: The algorithm in [Han, 2020].

In this experiment, we also propose improved versions of US algorithm and MUS algorithm, by replacing their inputs A and $\tilde{P}B$ with their top five left singular vectors U_A and $U_{\tilde{P}B}$. This process can be viewed as a simple version of the first step subspace learning in [Yao et al., 2021]. For the US algorithm, we run it for each column of $\tilde{P}B$ independently. We provide the result by varying the sparsity of $\tilde{P}$, i.e., $H(\pi_{\tilde{P}})$, and report the permutation recovery statistics $d_H(\hat{\pi}, \pi^*)$, where $\hat{\pi}$ is the recovered permutation and π^* is the ground-truth permutation, in Figure 1(a). Besides, we also report the residual error for the US algorithm, i.e.,

$$\text{residual error} = \frac{\|\hat{P}B - B\|_F^2}{\|B\|_F^2}$$

where $\hat{P}$ denotes the recovered permutation matrix, in Figure 1(b). Notably, these results verify our discussions that, although US algorithm can perform well in vector recovery (Achieving roughly 0.001 residual error on average.), it is extremely inferior when it comes to the permutation recovery.



(a) Permutation error

(b) Residual error

Figure 1: Performance of MRUC, MUS and US algorithms on a simulated 1-correspondence problem without missing values.

K DETAILS FOR THE EXPERIMENTS

We use Matlab 2020b for the numerical experiments. The computer environment consists of Intel i9-10920x for CPU and 32GB RAM.

[1] https://github.com/liangzu/CCVMIN.

K.1 HYPERPARAMETERS SETTING

Simulated data. We adopt fixed nuclear norm coefficient λ in the experiments on simulated data. Specifically, for each setting, we choose the best λ out of three candidate values that are 0.4, 0.5 and 0.6. Since adopting large ω will preserve the final performance and only degrade the convergence speed, we take $\omega = 3$ for all the experiments. For the tolerance of Sinkhorn algorithm, we take $\varepsilon = 0.01$ for all the experiments.

MovieLens 100K. For all the algorithms, we adopt a sequence of values for λ. Specifically, we start the algorithm with $\lambda = 300$, and once the algorithm stops improving the objective function for 10 steps, we shrink the value as $\lambda \leftarrow \lambda - 10$ until λ becomes lower than 10. We take $\omega = 0.5$ for all the experiments and also set the tolerance of Sinkhorn algorithm as $\varepsilon = 0.01$.

K.2 PHASE TRANSITION WITH DIFFERENT INITIALIZATIONS.

In this section, we conduct a simple experiment to explore the sensitivity of M^{3}O w.r.t initialization by varying the distance between initialization and the ground-truth matrix. We could expect that the variance of the performance of M^{3}O should decrease as the distance decreases.

We generate different initializations in the following way: We first generate two matrices M and W independently following the way described in Section 4.1, and we employ M as the ground-truth matrix. Then, we generate the initialization for M^{3}O as

$$\hat{M} = \Lambda M + (1 - \Lambda)W,$$

where $\Lambda \in (0, 1)$ is a coefficient designed for controlling the distance between initialization and the ground-truth matrix.

Figure 2 shows a phase transition phenomenon for M^{3}O algorithm w.r.t to the coefficient Λ, which is well aligned with our expectation.

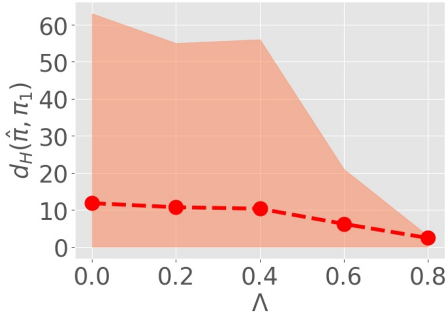


Figure 2: A phase transition phenomenon for M^{3}O algorithm w.r.t to the distance between initialization and the ground-truth matrix. The experiment is conducted on a 1-correspondence problem, with $|\Omega| \cdot 100\%/(n \cdot m) = 80\%$, $\eta = 0.1$, $n = m = 100$, $r = 5$, $m_A = 60$, and $m_1 = 40$. The mean with minimum and maximum are calculated from 10 different random initializations.

K.3 NUMBERS OF SINKHORN ITERATION

Typically, the numbers of Sinkhorn iteration required to retrieve an ε-good solution mainly depends on the entropy coefficient ϵ. This also implies that the decaying entropy regularization strategy can also accelerate the convergence process. Figure 3 shows the relationship between the numbers of Sinkhorn iteration and entropy coefficient ϵ under the same simulated data setting with Figure 2. The dash lines and intervals reflect mean, min, maximum aggregated from 20 independent trials. For a practical implementation, we restrict the maximum numbers of Sinkhorn iteration to 10000 on the numerical experiments.

K.4 PROBLEM FORMULATION FOR THE FACE RECOVERY PROBLEM

We show that M^{3}O is flexible and can also be used to recover matrix that is not in the form $[A, PB]$. We can see this from the problem formulation in (12), where the cost matrix $C(\cdot)$ can be constructed in other ways as long as it is a function of

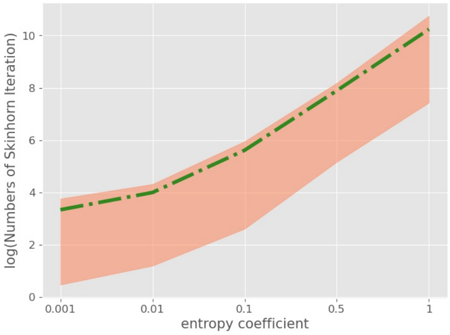


Figure 3: The required numbers of Sinkhorn iteration v.s. entropy coefficient ϵ

a permutation. Typically, M^{3}O can be used to solve a challenging face image recovery problem. The original face image with size 180×180 in Figure 4(a) comes from the Extend Yale B database [Georghiades et al., 2001]. The corrupted image is visualized in Figure 4(b), where the pixel blocks with size 30×30 in the upper left are shuffled randomly, and 30% of the total pixels are removed. This experiment setting is similar to that in [Yao et al., 2021] but the algorithm in [Yao et al., 2021] can not be applied since it can not work with the missing values. The MUS algorithm is also not applicable since this problem can not be written in the form of linear regression problem. From Figure 4(c) and 4(d) we can find that M^{3}O performs better than the Baseline, and can even recover the original orders of pixel blocks.

In the face recovery experiment, the cost matrix C is constructed as

$$C(i,j) = \|P_\Omega(B(i) - \widehat{M}(j))\|_F^2,$$

where $B(1), ..., B(13) \in \mathbb{R}^{30\times 30}$ are the shuffled pixel blocks from the upper left of the corrupted image shown in Figure 4(b), and $\widehat{M}(1), ..., \widehat{M}(13) \in \mathbb{R}^{30\times 30}$ are the corresponding recovered pixel blocks from the upper left of the current recovered image.

We choose fixed stepsize $\rho_k = 0.1$, and choose the initial entropy coefficient as $\epsilon = 100$. To obtain the initial matrix $\widehat{M}$, we first complete each pixel blocks independently using the Soft-Impute algorithm. We denote the filled matrix as M_1, and carry out the singular decomposition of it as $M1 = \sum_i \sigma_i u_i v_i^\top$. Then we set the initial matrix as $\widehat{M} = \sigma_1 u_1 v_1^\top$.

More results similar to Figure 4 are shown in Figure 4.

References

Abubakar Abid, Ada Poon, and James Zou. Linear regression with shuffled labels. *arXiv preprint arXiv:1705.01342*, 2017.

Zhidong Bai and Tailen Hsing. The broken sample problem. *Probability theory and related fields*, 131(4):528–552, 2005.

Babak Barazandeh and Meisam Razaviyayn. Solving Non-Convex Non-Differentiable Min-Max Games Using Proximal Gradient Method. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3162–3166. IEEE, 2020.

Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

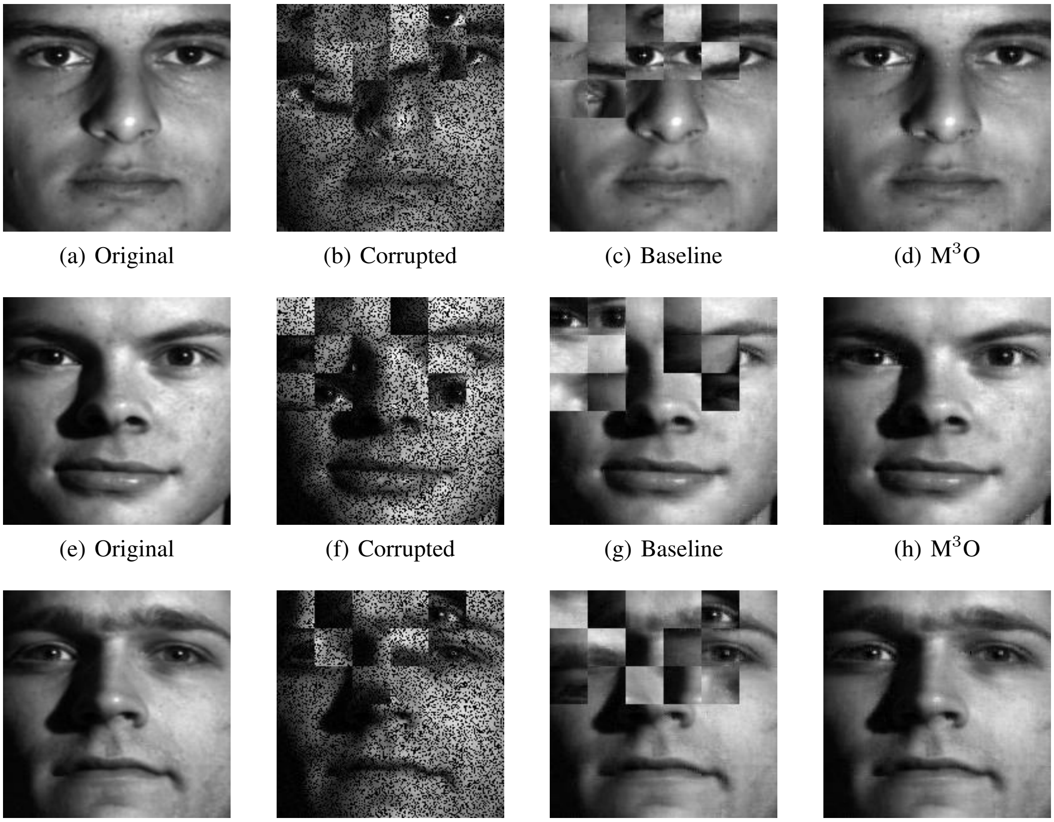
Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

Daniel Billsus, Michael J Pazzani, et al. Learning collaborative information filters. In *Icml*, volume 98, pages 46–54, 1998.

Garrett Birkhoff. Three observations on linear algebra. *Univ. Nac. Tacuman, Rev. Ser. A*, 5:147–151, 1946.

Emmanuel J Candès and Terence Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

Hock-Peng Chan and Wei-Liem Loh. A file linkage problem of degroot and goel revisited. *Statistica Sinica*, pages 1031–1045, 2001.



(a) Original (b) Corrupted (c) Baseline (d) M^{3}O

(e) Original (f) Corrupted (g) Baseline (h) M^{3}O

(i) Original (j) Corrupted (k) Baseline (l) M^{3}O

Figure 4: Performance of M^{3}O on more face images from Yale B database.

Debasmit Das and C. S. George Lee. Sample-to-Sample Correspondence for Unsupervised Domain Adaptation. *Engineering Applications of Artificial Intelligence*, 73:80–91, August 2018. ISSN 09521976. doi: 10.1016/j.engappai.2018.05.001. URL `http://arxiv.org/abs/1805.00355`. arXiv: 1805.00355.

Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246, 2018.

Herbert A David and Haikady N Nagaraja. *Order statistics*. John Wiley & Sons, 2004.

Morris H DeGroot and Prem K Goel. Estimation of the correlation coefficient from a broken random sample. *The Annals of Statistics*, pages 264–278, 1980.

David S Dummit and Richard M Foote. *Abstract algebra*, volume 1999. Prentice Hall Englewood Cliffs, NJ, 1991.

Varun Ganapathi, Christian Plagemann, Daphne Koller, and Sebastian Thrun. Real-time human pose tracking from range data. In *European conference on computer vision*, pages 738–751. Springer, 2012.

A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.

Michael Grant and Stephen Boyd. Cvx: Matlab software for disciplined convex programming, version 2.1, 2014.

Marco Gruteser, Graham Schelle, Ashish Jain, Richard Han, and Dirk Grunwald. Privacy-aware location sensor networks. In *HotOS*, volume 3, pages 163–168, 2003.

Paul R Halmos. *Measure theory*, volume 18. Springer, 2013.

Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondence from object proposals. *IEEE transactions on pattern analysis and machine intelligence*, 40(7):1711–1725, 2017. Publisher: IEEE.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

Trevor Hastie, Rahul Mazumder, Jason D. Lee, and Reza Zadeh. Matrix completion and low-rank SVD via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015. Publisher: JMLR. org.

Daniel Hsu, Kevin Shi, and Xiaorui Sun. Linear regression without correspondence. *arXiv preprint arXiv:1705.07048*, 2017.

Xiaoqiu Huang and Anup Madan. Cap3: A dna sequence assembly program. *Genome research*, 9(9):868–877, 1999.

Pan Ji, Hongdong Li, Mathieu Salzmann, and Yuchao Dai. Robust motion segmentation with unknown correspondence. In *European conference on computer vision*, pages 204–219. Springer, 2014.

Kui Jia, Tsung-Han Chan, Zinan Zeng, Shenghua Gao, Gang Wang, Tianzhu Zhang, and Yi Ma. ROML: A robust feature correspondence approach for matching objects in a set of images. *International Journal of Computer Vision*, 117(2): 173–197, 2016. Publisher: Springer.

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International Conference on Machine Learning*, pages 4880–4889. PMLR, 2020a.

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International Conference on Machine Learning*, pages 4880–4889. PMLR, 2020b.

Roy Jonker and Ton Volgenant. Improving the hungarian assignment algorithm. *Operations Research Letters*, 5(4):171–175, 1986.

Anatoli Juditsky and Arkadi Nemirovski. Solving variational inequalities with monotone operators on domains given by linear minimization oracles. *Mathematical Programming*, 156(1-2):221–256, 2016. Publisher: Springer.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Joseph B Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear algebra and its applications*, 18(2):95–138, 1977.

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pages 6083–6093. PMLR, 2020.

Sijia Liu, Songtao Lu, Xiangyi Chen, Yao Feng, Kaidi Xu, Abdullah Al-Dujaili, Mingyi Hong, and Una-May O'Reilly. Min-max optimization without gradients: Convergence and applications to black-box evasion and poisoning attacks. In *International Conference on Machine Learning*, pages 6282–6293. PMLR, 2020.

Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications. *IEEE Transactions on Signal Processing*, 2020. Publisher: IEEE.

Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, 11:2287–2322, 2010. Publisher: JMLR. org.

Sanjay Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on optimization*, 2(4): 575–601, 1992.

Amin Nejatbakhsh and Erdem Varol. Robust approximate linear regression without correspondence. *arXiv preprint arXiv:1906.00273*, 2019.

Arkadi Nemirovski. Prox-method with rate of convergence O (1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004. Publisher: SIAM.

Yurii Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2-3):319–344, 2007. Publisher: Springer.

Richard Nock, Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Jakub Nabaglo, Giorgio Patrini, Guillaume Smith, and Brian Thorne. The impact of record linkage on learning from feature partitioned data. In *International Conference on Machine Learning*, pages 8216–8226. PMLR, 2021.

Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D. Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. In *Advances in Neural Information Processing Systems*, pages 14934–14942, 2019.

Ashwin Pananjady, Martin J Wainwright, and Thomas A Courtade. Denoising linear models with permuted data. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 446–450. IEEE, 2017a.

Ashwin Pananjady, Martin J Wainwright, and Thomas A Courtade. Linear regression with shuffled data: Statistical and computational limits of permutation recovery. *IEEE Transactions on Information Theory*, 64(5):3286–3300, 2017b.

Liangzu Peng and Manolis C Tsakiris. Linear regression without correspondences via concave minimization. *IEEE Signal Processing Letters*, 27:1580–1584, 2020.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint arXiv:1810.02060*, 2018.

Meisam Razaviyayn, Tianjian Huang, Songtao Lu, Maher Nouiehed, Maziar Sanjabi, and Mingyi Hong. Nonconvex min-max optimization: Applications, challenges, and recent theoretical advances. *IEEE Signal Processing Magazine*, 37 (5):55–66, 2020. Publisher: IEEE.

Rodrigo Santa Cruz, Basura Fernando, Anoop Cherian, and Stephen Gould. Deeppermnet: Visual permutation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3949–3957, 2017.

J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.

Frederik Schaffalitzky and Andrew Zisserman. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *European conference on computer vision*, pages 414–431. Springer, 2002.

Martin Slawski, Emanuel Ben-David, and Ping Li. Two-stage approach to multivariate linear regression with sparsely mismatched data. *J. Mach. Learn. Res.*, 21(204):1–42, 2020a.

Martin Slawski, Mostafa Rahmani, and Ping Li. A sparse representation-based approach to linear regression with partially shuffled labels. In *Uncertainty in Artificial Intelligence*, pages 38–48. PMLR, 2020b.

Martin Slawski, Guoqing Diao, and Emanuel Ben-David. A pseudo-likelihood approach to linear regression with partially shuffled data. *Journal of Computational and Graphical Statistics*, pages 1–13, 2021.

Kiran K. Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. In *Advances in Neural Information Processing Systems*, pages 12680–12691, 2019.

Kim-Chuan Toh and Sangwoon Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of optimization*, 6(615-640):15, 2010.

Manolis Tsakiris and Liangzu Peng. Homomorphic sensing. In *International Conference on Machine Learning*, pages 6335–6344. PMLR, 2019.

Manolis C Tsakiris, Liangzu Peng, Aldo Conca, Laurent Kneip, Yuanming Shi, and Hayoung Choi. An algebraic-geometric approach for linear regression without correspondences. *IEEE Transactions on Information Theory*, 66(8):5130–5144, 2020.

Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. *submitted to SIAM Journal on Optimization*, 2(3), 2008.

Jayakrishnan Unnikrishnan, Saeid Haghighatshoar, and Martin Vetterli. Unlabeled sensing with random linear measurements. *IEEE Transactions on Information Theory*, 64(5):3237–3253, 2018.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019.

John Wright and Yi Ma. *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*. Cambridge University Press, 2021.

Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. A fast proximal point method for computing exact wasserstein distance. In Ryan P. Adams and Vibhav Gogate, editors, *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pages 433–453. PMLR, 22–25 Jul 2020. URL `https://proceedings.mlr.press/v115/xie20b.html`.

Yujia Xie, Yixiu Mao, Simiao Zuo, Hongteng Xu, Xiaojing Ye, Tuo Zhao, and Hongyuan Zha. A hypergradient approach to robust regression without correspondence. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=l35SB-_raSQ`.

Liu Yang, Ben Tan, Vincent W Zheng, Kai Chen, and Qiang Yang. Federated recommendation systems. In *Federated Learning*, pages 225–239. Springer, 2020.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

Yunzhen Yao, Liangzu Peng, and Manolis C Tsakiris. Unlabeled principal component analysis. *arXiv preprint arXiv:2101.09446*, 2021.

Zinan Zeng, Tsung-Han Chan, Kui Jia, and Dong Xu. Finding correspondence from multiple images via sparse and low-rank decomposition. In *European Conference on Computer Vision*, pages 325–339. Springer, 2012.

Hang Zhang and Ping Li. Optimal estimator for unlabeled linear regression. In *International Conference on Machine Learning*, pages 11153–11162. PMLR, 2020.

Hang Zhang, Martin Slawski, and Ping Li. The benefits of diversity: Permutation recovery in unlabeled sensing from multiple measurement vectors. *arXiv preprint arXiv:1909.02496*, 2019a.

Hang Zhang, Martin Slawski, and Ping Li. Permutation recovery from multiple measurement vectors in unlabeled sensing. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 1857–1861. IEEE, 2019b.

Jiawei Zhang, Peijun Xiao, Ruoyu Sun, and Zhi-Quan Luo. A Single-Loop Smoothed Gradient Descent-Ascent Algorithm for Nonconvex-Concave Min-Max Problems. *arXiv preprint arXiv:2010.15768*, 2020.

Yu Zhang, Bin Cao, and Dit-Yan Yeung. Multi-domain collaborative filtering. *arXiv preprint arXiv:1203.3535*, 2012.

Yu Zheng. Methodologies for cross-domain data fusion: An overview. *IEEE transactions on big data*, 1(1):16–34, 2015.

Tinghui Zhou, Philipp Krahenbuhl, Mathieu Aubry, Qixing Huang, and Alexei A. Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 117–126, 2016.

Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4032–4040, 2015.