Low-Rank Tensor Recovery with Euclidean-Norm-Induced Schatten-p Quasi-Norm Regularization

Anonymous authors
Paper under double-blind review

Abstract

The nuclear norm and Schatten-p quasi-norm are popular rank proxies in low-rank matrix recovery. Unfortunately, computing the nuclear norm or Schatten-p quasi-norm of a tensor is NP-hard, which is a pity for low-rank tensor completion (LRTC) and tensor robust principal component analysis (TRPCA). In this paper, we propose a new class of tensor rank regularizers based on the Euclidean norms of the CP component vectors of a tensor and show that these regularizers are monotonic transformations of tensor Schatten-p quasi-norm. This connection enables us to minimize the Schatten-p quasi-norm in LRTC and TRPCA implicitly. The methods do not use the singular value decomposition and hence scale to big tensors. Moreover, the methods are not sensitive to the choice of initial rank and provide an arbitrarily sharper rank proxy for low-rank tensor recovery compared to nuclear norm. On the other hand, we study the generalization abilities of LRTC with Schatten-p quasi-norm regularization and LRTC with our regularizers. The theorems show that a relatively sharper regularizer leads to a tighter error bound, which is consistent with our numerical results. Numerical results on syhthetic data and real data demonstrate the effectiveness and superiority of our methods compared to baseline methods.

1 Introduction

Low-rank tensor completion (LRTC) (Gandy et al., 2011; Acar et al., 2011; Liu et al., 2012; Romera-Paredes & Pontil, 2013; Kressner et al., 2014; Yuan & Zhang, 2016; Cheng et al., 2016; Zhou et al., 2017; Lacroix et al., 2018; Liu & Moitra, 2020; Fan, 2022), as a high-order generalization of low-rank matrix completion (LRMC) (Candès & Recht, 2009; Hardt, 2014), aims to recover the missing entries of a low-rank tensor. It is natural to extend the ideas of many LRMC methods, such as nuclear norm minimization (Candès & Recht, 2009), Schatten-p quasi-norm minimization (Nie et al., 2012), and low-rank factorization (Srebro & Shraibman, 2005; Fan et al., 2019), to tensor completion (Liu et al., 2012; Gandy et al., 2011; Bazerque et al., 2013; Huang et al., 2015; Zhang & Aeron, 2016; Kong et al., 2018; Ghadermarzy et al., 2019). Given an incomplete tensor, as mentioned before, one may recover the missing entries by performing matricization along every mode and minimizing the nuclear norms of these matrices, which we call tensor unfoldings. Liu et al. (2012) proposed to minimize a weighted sum of the nuclear norms of the d unfoldings. Yuan & Zhang (2016) pointed out that these unfolding-based tensor completion algorithms require much more observed entries for exact recovery than direct tensor completion without matricization. In general, LRTC methods can be organized into different categories according to the types of factorization model, e.g., CP (CANDECOMP/PARAFAC) decomposition based LRTC (Acar et al., 2011; Jain & Oh, 2014; Zhao et al., 2015; Liu & Moitra, 2020), Tucker decomposition bassed LRTC (Xu et al., 2013; Xu & Yin, 2013; Kressner et al., 2014; Kasai & Mishra, 2016; Xie et al., 2018; Kong et al., 2018), and tensor ring based LRTC (Yuan et al., 2019). This paper will focus on CP decomposition based LRTC.

It is known that tensor nuclear norm is hard to compute in practice. One has to consider other tractable solutions such as learning a CP model directly from the incomplete data (Acar et al., 2011; Jain & Oh, 2014). Jain & Oh (2014) proved that an $n \times n \times n$ tensor of rank r can be recovered from $O(n^{3/2}r^5 \log^4(n))$ randomly sampled entries, via alternating minimization. Barak & Moitra (2016), Potechin & Steurer (2017), and Foster & Risteski (2019) used the sum-of-squares hierarchy as a tractable relaxation to the tensor nuclear

norm. Particularly, Foster & Risteski (2019) showed that for noisy completion of a rank-r $n \times n \times n$ tensor, the generalization bound is $\tilde{O}(r^2 n^{3/2}/|\Omega|)$. Ghadermarzy et al. (2019) proposed to use their defined tensor max-qnorm and M-norm as a rank proxy for tensor completion, and proved that $O(r^{1.5d}dn)$ observed entries are sufficient to recover a dth-order tensor of rank r and size n with high probability. For 3rd-order tensor completion, Bazerque et al. (2013) used the sum of squared Frobenius norms as regularizer and showed that it is related to the $\ell_{2/3}$ norm of the weights in CP decomposition. Yang et al. (2016) applied group-sparse regularization to LRTC and showed that the regularizer is related to the $\ell_{1/3}$ norm. Shi et al. (2017) proposed to directly minimize the ℓ_1 norm of the weights in CP decomposition for LRTC. In fact, the regularizers presented in the above three works are closely related to the tensor Schatten-p (quasi) norm with $p = 1$, $2/3$, or $1/3$. Notice that these works are purely empirical-motivated and have no theoretical guarantee on the tensor completion performance. Moreover, the regularizations are limited in the discrete values $\{1, 2/3, 1/3\}$ and only for 3rd-order tensors. One may expect to exploit Schatten-p quasi-norm with arbitrary p on arbitrary-order tensor and have theoretical guarantees for the recovery.

Besides LRTC, tensor robust principal component analysis (TRPCA) is another important problem of tensor recovery. TRPCA is a generalization of robust PCA (De la Torre & Black, 2001; Ding et al., 2006; Candès et al., 2011; Haeffele & Vidal, 2019) and aims to decompose a noisy tensor to the sum of a low-rank tensor and a sparse tensor. Many recent works on TRPCA can be found in (Anandkumar et al., 2016; Lu et al., 2016; Xie et al., 2018; Goldfarb & Qin, 2014; Zheng et al., 2019; Lu et al., 2019; Wang et al., 2020). For example, Lu et al. (2019) defined a new tensor nuclear norm based on the t-product (Kilmer & Martin, 2011) of tensors and provided sufficient conditions for exact recovery. Notice that these TRPCA algorithms have high computational costs on large-scale data. If we take advantage of factorization model (e.g. (Anandkumar et al., 2016)) for fast TRPCA, we need to estimate the rank. In addition, one may use sharper rank proxies like the Schatten-p quasi-norm if it can be handled efficiently.

In this paper, we focus on fast and accurate LRTC and TRPCA. Our contributions are two-fold.

- We propose a new class of regularizers as a tensor rank proxy. These are based on the Euclidean norms of the component vectors in the form of CP decomposition. We show that the regularizers are monotonic transformations of Schatten-p quasi-norms, where p could be any positive value and the tensor could have arbitrary order. We also provide asymmetric variational forms for the tensor Schatten-p quasi-norm with discrete p. Each of the asymmetric regularizers could have only one nonconvex and nonsmooth term such that the optimization has fewer nonconvex and nonsmooth subproblems than the case of the symmetric regularizers. The regularizers enable us to minimize Schatten-p quasi-norms without performing SVD and hence solve large tensor recovery problems more efficiently. When applying the regularizers to LRTC and TRPCA, the recovery performance is robust to the choice of initial rank and the recovery accuracy is high when p is much less than 1.
- We provide generalization error bounds for LRTC with Schatten-p quasi-norm regularization and LRTC with our regularizers. We show that under some mild conditions, a smaller p (but not too small) in the tensor Schatten-p quasi-norm or a smaller q in our factorization regularizers lead to a tighter generalization error bound. Note that the bounds are also applicable to LRMC.

The numerical results of LRTC and TRPCA on synthetic data, image inpainting, and image denoising corroborate the effectiveness and superiority of our methods over state-of-the-art baseline methods.

2 Euclidean Regularization

First of all, we introduce the following definitions in terms of the CP decomposition.

Definition 1. *Let $\boldsymbol{x}_i^{(j)} \in \mathbb{R}^{n_j \times 1}$, $i \in [r]$, $j \in [d]$. The rank of a tensor $\boldsymbol{\mathcal{X}} \in \mathcal{R}^{n_1 \times n_2 \ldots \times n_d}$ is defined as the minimum number of rank-one tensors that sum to $\boldsymbol{\mathcal{X}}$:*

$$\operatorname{rank}(\boldsymbol{\mathcal{X}}) = \min \Big\{ r \in \mathbb{N} : \ \boldsymbol{\mathcal{X}} = \sum_{i=1}^{r} \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \cdots \circ \boldsymbol{x}_i^{(d)} \Big\}.$$

Definition 2. *The nuclear norm (Friedland & Lim, 2018) of a tensor* $\boldsymbol{\mathcal{X}}$ *is defined as* $\|\boldsymbol{\mathcal{X}}\|_* = \inf\left\{ \sum_{i=1}^r |s_i| : \ \boldsymbol{\mathcal{X}} = \sum_{i=1}^r s_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \cdots \circ \boldsymbol{u}_i^{(d)}, \|\boldsymbol{u}_i^{(j)}\| = 1, \ r \in \mathbb{N} \right\}$.

Using Definition 2, we see the tensor nuclear norm is a convex relaxation of tensor rank, though it is NP-hard to compute the tensor rank and tensor nuclear norm (Friedland & Lim, 2018). Similarly, we may define a tensor Schatten-p quasi-norm that extends matrix Schatten-p quasi-norm. The p-th power of the tensor Schatten-p quasi-norm is a nonconvex relaxation of tensor rank.

Definition 3. *The Schatten-p quasi-norm* $(0 < p < 1)$ *of a tensor* $\boldsymbol{\mathcal{X}}$ *is defined as* $\|\boldsymbol{\mathcal{X}}\|_{S_p} = \inf\left\{ \left(\sum_{i=1}^r |s_i|^p\right)^{1/p} : \ \boldsymbol{\mathcal{X}} = \sum_{i=1}^r s_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \ldots \circ \boldsymbol{u}_i^{(d)}, \|\boldsymbol{u}_i^{(j)}\| = 1, \ r \in \mathbb{N}, \ 0 < p < 1 \right\}$.

For convenience, we define an operator that forms a tensor $\boldsymbol{\mathcal{X}}$ from a collection of vectors $\boldsymbol{x}$.

Definition 4. *Define the inverse CP operator as*

$$\mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)}) = \sum_{i=1}^k \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \cdots \circ \boldsymbol{x}_i^{(d)}.$$

We provide the following variational form of the tensor Schatten-p (quasi) norm $(0 < p \leq 1)$:

Theorem 1 (Symmetric Regularizer[1])**.** *The tensor Schatten-p (quasi) norm* $(0 < p \leq 1)$ *can be represented as a function of the Euclidean norms of the CP components:*

$$\|\boldsymbol{\mathcal{X}}\|_{S_p}^p = \inf\left\{ \frac{1}{d} \sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^{pd} : \boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)}) \right\}.$$

Compared to the definition (Definition 3) of the tensor Schatten-p quasi-norm, $\|\boldsymbol{\mathcal{X}}\|_{S_p}$ given by Theorem 1 internalizes the weight factors $\boldsymbol{s}$ into the vectors $\boldsymbol{u}$ and removes the constraints on $\boldsymbol{u}$. This reduces decision variables and the computational cost of calculating the gradient in optimization. In addition, for some choice of p, there is no nonsmooth functions on the factors in Theorem 1, while there are always nonsmooth functions in Definition 3. Hence, the formulation of Schatten-p quasi-norm in Theorem 1 is more tractable than that in Definition 3. For instance, in Theorem 1, letting $p = 1$, $\frac{2}{d}$, or $\frac{1}{d}$, we have $\|\boldsymbol{\mathcal{X}}\|_* = \inf_{\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})} \ \frac{1}{d}\sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^d$, $\|\boldsymbol{\mathcal{X}}\|_{S_{2/d}}^{2/d} = \inf_{\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})} \ \frac{1}{d}\sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^2$, and $\|\boldsymbol{\mathcal{X}}\|_{S_{1/d}}^{1/d} = \inf_{\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})} \ \frac{1}{d}\sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|$. These three special cases are based on convex functions of Euclidean norms of component vectors, and hence are possibly easier to handle in optimization.

According to Theorem 1, for a relatively low-order tensor, when we want to obtain a sharp enough regularizer, we have to use a sufficiently small p for all i and j. Thus, every term in $\{\|\boldsymbol{x}_i^{(j)}\|^{pd}\}_{i\in[k]}^{j\in[d]}$ is nonconvex and nonsmooth, making it difficult to solve the optimization problem of low-rank tensor recovery. The following theorem provides a class of asymmetric regularizers that have fewer nonconvex and nonsmooth terms than those in Theorem 1.

Theorem 2 (Asymmetric Regularizer)**.** *Suppose* $q \in \{1, 1/2, 1/3, 1/4, \ldots\}$. *Let* $p_1 = q/(1 + qd - q)$ *and* $p_2 = 2q/(2 + qd - q)$. *We have*

$$\text{(a)} \ \|\boldsymbol{\mathcal{X}}\|_{S_{p_1}}^{p_1} = \inf_{\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})} \ p_1 \sum_{i=1}^k \left(\frac{1}{q}\|\boldsymbol{x}_i^{(1)}\|^q + \sum_{j=2}^d \|\boldsymbol{x}_i^{(j)}\| \right);$$

$$\text{(b)} \ \|\boldsymbol{\mathcal{X}}\|_{S_{p_2}}^{p_2} = \inf_{\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})} \ p_2 \sum_{i=1}^k \left(\frac{2}{q}\|\boldsymbol{x}_i^{(1)}\|^q + \sum_{j=2}^d \|\boldsymbol{x}_i^{(j)}\|^2 \right).$$

In Theorem 2 (b), the terms of $j \geq 2$ are convex and smooth while those in Theorem 2 (a) are nonsmooth. Therefore, the optimization related to Theorem 2 (b) is easier. Since 3rd-order tensors (e.g. color images

[1]It is worth mentioning that the Theorem 2 of (Cheng et al., 2016) and the Proposition 1 of (Lacroix et al., 2018) are two special cases of our Theorem 1 (when $p = 1$ and $d = 3$ or $p = 2/3$ and $d = 3$).

and videos) are more prevalent than tensors with other orders, we list their symmetric and asymmetric regularizers with only convex terms in Table 1 for convenience. Note that the last two regularizers in the table are not the consequences of Theorem 2. The derivations are in Appendix D.3.

Table 1: Regularizers ($\mathcal{R}(\boldsymbol{\mathcal{X}})$) with only convex terms for 3rd-order tensor ($\boldsymbol{\mathcal{X}} = \mathcal{CP}_k^3(\boldsymbol{x})$).

	$\mathcal{R}(\boldsymbol{\mathcal{X}})$)	Characterization based on Euclidean norm
Symmetric	$\|\boldsymbol{\mathcal{X}}\|_*$	$\frac{1}{3}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\|^3 + \|\boldsymbol{x}_i^{(2)}\|^3 + \|\boldsymbol{x}_i^{(3)}\|^3\right)$
	$\|\boldsymbol{\mathcal{X}}\|_{S_{2/3}}^{2/3}$	$\frac{1}{3}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\|^2 + \|\boldsymbol{x}_i^{(2)}\|^2 + \|\boldsymbol{x}_i^{(3)}\|^2\right)$
	$\|\boldsymbol{\mathcal{X}}\|_{S_{1/3}}^{1/3}$	$\frac{1}{3}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\| + \|\boldsymbol{x}_i^{(2)}\| + \|\boldsymbol{x}_i^{(3)}\|\right)$
Asymmetric	$\|\boldsymbol{\mathcal{X}}\|_{S_{1/2}}^{1/2}$	$\frac{\sqrt{2}}{4}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\|^2 + \|\boldsymbol{x}_i^{(2)}\|^2 + \|\boldsymbol{x}_i^{(3)}\|\right)$
	$\|\boldsymbol{\mathcal{X}}\|_{S_{2/5}}^{2/5}$	$\frac{16^{1/5}}{5}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\|^2 + \|\boldsymbol{x}_i^{(2)}\| + \|\boldsymbol{x}_i^{(3)}\|\right)$
	$\|\boldsymbol{\mathcal{X}}\|_{S_{3/7}}^{3/7}$	$\frac{81^{1/7}}{7}\sum_{i=1}^k \left(\|\boldsymbol{x}_i^{(1)}\|^3 + \|\boldsymbol{x}_i^{(2)}\| + \|\boldsymbol{x}_i^{(3)}\|\right)$

In sum, the regularizers we proposed in this section covers all Schatten-p (quasi) norms with any $0 < p \leq 1$ for any-order tensors. Some of the regularizers, especially those asymmetric ones, are based on convex or/and smooth functions on the tensor factors, which provide convenience for application and optimization. These regularizers can be applied to LRTC and TRPCA that enjoy a variety of real applications in machine learning and computer vision.

The tensor regularizers we presented in this paper are closely related to the variational forms of matrix nuclear norm and Schatten-p quasi-norms (Shang et al., 2016; 2017; Fan et al., 2019; Giampouras et al., 2020). For instance, the squared sum of Frobenius norms of two factors of a matrix is lower bounded by the nuclear norm of the matrix, which is a special case of our Theorem 1 with $d = 2$ and $p = 1$. In (Fan et al., 2019), the authors provided a class of SVD-free variational forms of the matrix Schatten-p quasi-norm with discrete p that can be arbitrarily small. Our tensor regularizers can be regarded as a generalization of the variational form of Schatten-p quasi-norm from matrix to tensor. Nevertheless, theoretical guarantees about these tensor regularizers in low-rank tensor recovery are more difficult to derive than their matrix counterparts.

3 Low-Rank Tensor Completion

3.1 LRTC-ENR algorithm

Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{n_1 \times n_2 \ldots \times n_d}$ be a rank-r tensor. Suppose we observed a few noisy entries of $\boldsymbol{\mathcal{T}}$ randomly (without replacement):

$$[\boldsymbol{\mathcal{D}}]_{j_1 j_2 \ldots j_d} = [\boldsymbol{\mathcal{T}}]_{j_1 j_2 \ldots j_d} + [\boldsymbol{\mathcal{N}}]_{j_1 j_2 \ldots j_d}, \quad (j_1, \ldots, j_d) \in \Omega$$

where Ω consists of the locations of the observed entries and each entry of the noise tensor $\boldsymbol{\mathcal{N}}$ is drawn from $\mathcal{N}(0, \sigma^2)$. To recover $\boldsymbol{\mathcal{T}}$ from $\boldsymbol{\mathcal{D}}$, one may solve

$$\underset{\boldsymbol{\mathcal{X}}}{\text{minimize}} \, \frac{1}{2}\|\mathcal{P}_\Omega\left(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\right)\|_F^2 + \lambda \|\boldsymbol{\mathcal{X}}\|_{S_p}^p, \tag{1}$$

of which the solution is an estimate of $\boldsymbol{\mathcal{T}}$. However, Problem (1) is intractable. Instead, based on the analysis in Section 2, we propose to solve

$$\underset{\{\boldsymbol{x}_i^{(j)}\}}{\text{minimize}} \, \frac{1}{2}\left\|\boldsymbol{\mathcal{M}} * \left(\boldsymbol{\mathcal{D}} - \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)})\right)\right\|_F^2 + \lambda \mathcal{R}\left(\{\boldsymbol{x}_i^{(j)}\}\right), \tag{2}$$

where $\boldsymbol{\mathcal{M}}$ is a binary tensor with 0 for missing entries and 1 for observed entries and the sign "$*$" denotes the element-wise product of tensors. $\mathcal{R}\left(\{\boldsymbol{x}_i^{(j)}\}\right)$ denotes a certain regularizer in Theorem 1, Theorem 2, or Table 1, e.g.,

$$\mathcal{R}\left(\{\boldsymbol{x}_i^{(j)}\}\right) = \sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^{pd}, \tag{3}$$

in which we have omitted the constant for simplicity. Note that even finding a stationary point of (1) is hard due to computation of the Schatten-p quasi-norm while finding a stationary point of (2) is computationally feasible. For convenience, we call problem (2) Low-Rank Tensor Completion with Euclidean Norm Regularization (**LRTC-ENR**). We propose to solve (2) by Block Coordinate Descent (BCD) with Extrapolation (BCDE for short) (Xu & Yin, 2013), which is more efficient than BCD. Note that when $p < 1/d$, we integrate BCDE with iteratively reweighted method (Lu, 2014). We may also use quasi-Newton methods such as L-BFGS (Liu & Nocedal, 1989) to solve problem (2) even when the objective function is nonsmooth. The details of the optimization for (2) are in Appendix A.

It is worth mentioning that according to Definition 3, one may consider the following LRTC problem

$$\begin{aligned} \underset{\{\boldsymbol{x}_i^{(j)}\},\boldsymbol{s}}{\text{minimize}} \quad & \frac{1}{2}\left\|\boldsymbol{\mathcal{M}} * \left(\boldsymbol{\mathcal{D}} - \widetilde{\mathcal{CP}}_k^d(\{\boldsymbol{x}_i^{(j)}\}, \boldsymbol{s})\right)\right\|_F^2 + \lambda \sum_{i=1}^k |s_i|^p, \\ \text{subject to} \quad & \|\boldsymbol{x}_i^{(j)}\| = 1, \forall i \in [k], j \in [d]. \end{aligned} \tag{LRTC-Schatten-p}$$

where $\widetilde{\mathcal{CP}}_k^d(\{\boldsymbol{x}_i^{(j)}\}, \boldsymbol{s}) = \sum_{i=1}^k s_i \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \ldots \circ \boldsymbol{x}_i^{(d)}$. Nevertheless, it is more difficult to solve LRTC-Schatten-p than LRTC-ENR due to the following reasons. First, LRTC-Schatten-p has one more block of variables $\boldsymbol{s}$, of which the gradient computation is costly. Second, constrained optimization LRTC-Schatten-p is generally harder than unconstrained optimization. Heuristically, we suggest using BCDE with iteratively reweighted minimization embedded to solve LRTC-Schatten-p. It is compared with LRTC-ENR in Figure 3 of Section 5.1.

3.2 Generalization Error Bound of LRTC

In this section, we study the generalization ability of LRTC-Schatten-p and LRTC-ENR. It should be pointed out that, compared to LRMC, it is much more difficult to analyze the generalization ability of LRTC. The main reason is that we may not have orthogonal factors in a CP decomposition. Thus, we first consider the case of orthogonal CP decomposition. We will also provide generalization error bounds for the general cases (without the restriction of orthogonality).

Without loss of generality, we consider the case of hyper-cubic tensors, denoted by $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n^{\otimes d}}$. We define

Definition 5 (Orthogonal CP tensor set)**.**

(a) $\mathcal{S}_{d,n}^{\perp} := \left\{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n^{\otimes d}} : \ \boldsymbol{\mathcal{X}} = \sum_{i=1}^r s_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \ldots \circ \boldsymbol{u}_i^{(d)}; \forall j \in [d], {\boldsymbol{u}_i^{(j)}}^\top \boldsymbol{u}_l^{(j)} = 1 \text{ if } i = l, {\boldsymbol{u}_i^{(j)}}^\top \boldsymbol{u}_l^{(j)} = 0 \text{ if } i \neq l\right\}$.

(b) $\mathcal{S}_{d,n,p}^{\perp} := \{\boldsymbol{\mathcal{X}} \in \mathcal{S}_{d,n}^{\perp} : \ \|\boldsymbol{\mathcal{X}}\|_{S_p^{\perp}} \leq \psi_p\}$.

Accordingly, we denote the Schatten-p quasi-norm of $\boldsymbol{\mathcal{X}}$ with orthogonal CP decomposition by $\|\boldsymbol{\mathcal{X}}\|_{S_p^{\perp}}$. We have the following covering number result.

Theorem 3. *The covering numbers of $\mathcal{S}_{d,n,p}^{\perp}$ with respect to the Frobenius norm satisfy*

$$\log \mathcal{N}(\mathcal{S}_{d,n,p}^{\perp}, \|\cdot\|_F, \epsilon) \leq \left(\frac{1}{2} + \frac{1}{p}\right) nd \left(\log(d+1)\right) \left(\frac{c\psi_p}{\epsilon}\right)^{\frac{2p}{2-p}},$$

where $c > 0$ is a universal constant and $0 < p < 2$.

Note that although Theorem 3 does not include the case $p = 2$, it is much easier to obtain the covering number of $\mathcal{S}_{d,n,2}^{\perp}$, which will not be detailed in this paper because $\|\boldsymbol{\mathcal{X}}\|_{S_2^{\perp}}$ is useless in regularizing the tensor rank. Based on Theorem 3, we derive the following generalization error bound for Schatten-p quasi-norm regularized orthogonal tensor completion.

Theorem 4. *Suppose* $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{n^{\otimes d}}$, $\boldsymbol{\mathcal{X}} \in \mathcal{S}_{d,n}^{\perp}$, $\max\{\|\boldsymbol{\mathcal{D}}\|_\infty, \|\boldsymbol{\mathcal{X}}\|_\infty\} \leq \varepsilon$, *and* $0 < p < 2$. *Then there exists a numerical constant* c *such that with probability at least* $1 - 2n^{-d}$,

$$\frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F - \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F \leq c\varepsilon \left(\frac{(\frac{1}{2} + \frac{1}{p})nd\log(d+1)}{|\Omega|} \left(\frac{\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}}{\varepsilon\sqrt{dn}} \right)^{\frac{2p}{2-p}} \right)^{1/4}. \tag{4}$$

In the theorem, since $\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}/(\varepsilon\sqrt{dn}) > 1$ and $2p/(2-p)$ is an exponent and decreases when p decreases, a smaller p but not too small will lead to a tighter error bound, even though there is a term $1/p$ and $\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}$ is also associated with p. Here we use a toy example to illustrate the result of Theorem 4. We generate six $\boldsymbol{s}$ (defined in Definition 3) of dimension 20 with different decay rates corresponding to different low-rankness. They are used to form random tensors with $d = 3$ and $n = 100$. We let $|\Omega| = 0.1n^3$ and compute $\left((\frac{1}{2} + \frac{1}{p})nd\log(d+1)|\Omega|^{-1}(\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}/(\varepsilon\sqrt{dn}))^{\frac{2p}{2-p}}\right)^{1/4} \triangleq \Delta$ for each tensor. The six $\boldsymbol{s}$ and average Δ (corresponding to each $\boldsymbol{s}$) of 100 repeated trials are shown in Figure 1. We see that in each case, a smaller p but not too small will lead to a smaller Δ. When the decay rate is higher, the approximate rank is lower, which leads to a smaller Δ. On the other hand, reducing p is more useful when the decay rate is low.

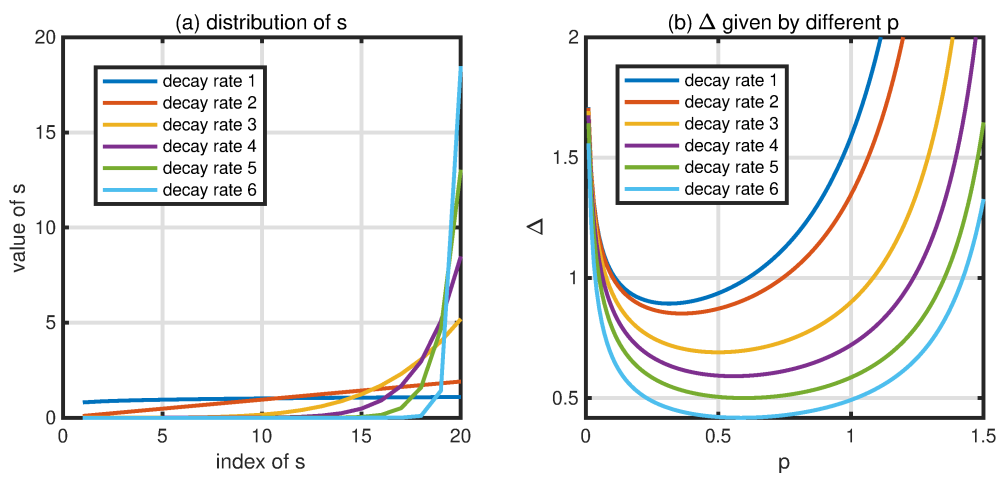

Figure 1: An intuitive example of the error bound of Theorem 4

Now we study the generalization error of LRTC-ENR (2). For convenience, let $\|\boldsymbol{Y}\|_{2,q} := (\sum_i \|\boldsymbol{y}_i\|^q)^{1/q}$ and define

$$\mathcal{S}_{d,n,q}^k = \left\{ \boldsymbol{\mathcal{X}} \in \mathbb{R}^{n^{\otimes d}} : \ \boldsymbol{\mathcal{X}} = \mathcal{CP}_k^d(\boldsymbol{x}_i^{(j)}); \|\boldsymbol{X}^{(j)}\|_{2,q} \leq \alpha_q^{(j)}, \ \|\boldsymbol{X}^{(j)}\|_{op} \leq \gamma_j, \ j \in [d] \right\}, \tag{5}$$

where $\boldsymbol{X}^{(j)} = [\boldsymbol{x}_1^{(j)}, \boldsymbol{x}_2^{(j)}, \ldots, \boldsymbol{x}_k^{(j)}]$. We have

Theorem 5. *The covering numbers of* $\mathcal{S}_{d,n,q}^k$ *with respect to the Frobenius norm satisfy*

$$\log \mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq \frac{c(n + \log(ek))}{q} \sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j \epsilon} \right)^{\frac{2q}{2-q}},$$

for any $0 < q < 2$, *and*

$$\log \mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq c' nk\log(2nk) \sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j \epsilon} \right)^2$$

for $q = 2$, *where* $\phi = \prod_{j=1}^d \gamma_j$ *and* c, c' *are universal constants.*

Based on Theorem 5, we get the generalization error bound of (2) as follows.

Theorem 6. *Suppose* $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{n^{\otimes d}}$, $\boldsymbol{\mathcal{X}} \in \mathcal{S}^k_{d,n,q}$, $\phi = \prod_{j=1}^d \gamma_j$, *and* $\max\{\|\boldsymbol{\mathcal{D}}\|_\infty, \|\boldsymbol{\mathcal{X}}\|_\infty\} \leq \varepsilon$. *Then there exists a numerical constant* c *such that with probability at least* $1 - 2n^{-d}$,

$$\frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F \leq \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F + c\varepsilon\left(\frac{B_q}{|\Omega|}\right)^{1/4},$$

where $B_q = \frac{n+\log(ek)}{q}\sum_{j=1}^d \left(\frac{\sqrt{d}\phi\alpha_q^{(j)}/\gamma_j}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)}$ *when* $0 < q < 2$ *and* $B_q = nk\log(2nk)\sum_{j=1}^d \left(\frac{\sqrt{d}\phi\alpha_q^{(j)}/\gamma_j}{\varepsilon\sqrt{n}}\right)^2$ *when* $q = 2$.

The theorem indicates that a smaller p but not too small in (2) leads to a tighter error bound, which can also be verified by a toy example similar to Figure 1. Based on Theorem 6, we can obtain the generalization error bound for (1), which is shown in the following corollary.

Corollary 1. *Suppose* $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{n^{\otimes d}}$, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n^{\otimes d}}$, $\max\{\|\boldsymbol{\mathcal{D}}\|_\infty, \|\boldsymbol{\mathcal{X}}\|_\infty\} \leq \varepsilon$, $0 < p < 2/d$, $\|\boldsymbol{\mathcal{X}}\|_{S_p}$ *is attained at* $\boldsymbol{\mathcal{X}} = \mathcal{CP}^d_k(\boldsymbol{x}_i^{(j)})$, *and* $\max_{j\in[d]}\|\boldsymbol{X}^{(j)}\|_{op} \leq \bar{\gamma}$. *Then there exists a numerical constant* c *such that with probability at least* $1 - 2n^{-d}$,

$$\frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F \leq \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F + c\varepsilon\left(\frac{B_p}{|\Omega|}\right)^{1/4},$$

where $B_p = \frac{n+\log(ek)}{p}\left(\frac{\bar{\gamma}^{d-1}\sqrt{d}}{\varepsilon\sqrt{n}}\|\boldsymbol{\mathcal{X}}\|_{S_p}^{1/d}\right)^{\frac{2pd}{2-pd}}$ *when* $0 < p < 2/d$ *and* $B_p = d^2k\log(2nk)\left(\frac{\bar{\gamma}^{d-1}\|\boldsymbol{\mathcal{X}}\|_{S_p}^{1/d}}{\varepsilon}\right)^2$ *when* $p = 2/d$.

Compared to Theorem 3, Corollary 1 as well as Theorem 6 does not require orthogonality.

Actually, compared to Theorem 4, Theorem 6, and Corollary 1, we can obtain tighter bounds via using the covering number bounds, Dudley entropy integral bound, and the transductive sample complexity bound of (El-Yaniv & Pechyony, 2009), at a price of breaking the continuity at $p = 1$ or $q = 1$. For instance, for (2), we have

Theorem 7. *Suppose* $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{n^{\otimes d}}$, $\boldsymbol{\mathcal{X}} \in \mathcal{S}^k_{d,n,q}$, $\phi = \prod_{j=1}^d \gamma_j$, *and* $\max\{\|\boldsymbol{\mathcal{D}}\|_\infty, \|\boldsymbol{\mathcal{X}}\|_\infty\} \leq \varepsilon$. *Denote* $\bar{\Omega}$ *the index set of the missing entries of* $\boldsymbol{\mathcal{D}}$ *and suppose* $|\bar{\Omega}| > |\Omega| > 50$. *Then with probability at least* $1 - \delta$ *over the random sampling of* Ω, *we have*

$$\frac{1}{|\bar{\Omega}|}\|\mathcal{P}_{\bar{\Omega}}(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F^2 \leq \frac{1}{|\Omega|}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F^2 + \varepsilon^2(B_{\mathcal{R}} + B_\delta)$$

where $B_\delta = \dfrac{44n^d}{\sqrt{|\Omega||\bar{\Omega}|}} + 12\sqrt{\dfrac{n^d}{|\Omega||\bar{\Omega}|}\log\dfrac{1}{\delta}}$ *and*

$$B_{\mathcal{R}} = \begin{cases} \frac{c_1}{|\Omega|}\sqrt{nk\log(2nk)\sum_{j=1}^d\left(\frac{d\phi\alpha_q^{(j)}}{\varepsilon\gamma_j}\right)^2}\log|\Omega| & \text{if } q = 2 \\ \frac{c_2}{|\Omega|}\sqrt{(n+\log(ek))\sum_{j=1}^d\left(\frac{d\phi\alpha_q^{(j)}}{\varepsilon\gamma_j}\right)^2}\log|\Omega| & \text{if } q = 1 \\ \frac{c_3}{|\Omega|}\sqrt{\frac{(n+\log(ek))(2-q)^2|\Omega|^{\frac{1-q}{2-q}}}{q(2-2q)^2}\sum_{j=1}^d\left(\frac{d\phi\alpha_q^{(j)}}{\varepsilon\gamma_j}\right)^{\frac{2q}{2-q}}} & \text{if } 0 < q < 1 \text{ or } 1 < q < 2 \end{cases}$$

with constants c_1, c_2, *and* c_3.

We see that in Theorem 7 the dominant part of squared error bound is almost linear with $1/|\Omega|$, while in Theorem6 squared error bound is linear with $1/\sqrt{|\Omega|}$. However, in Theorem 7, the bound is not continuous

around $q = 1$. In fact, Theorem 7 indicates that a smaller q but not too small leads to a tighter error bound and for a smaller $|\Omega|$ reducing the value of q becomes more useful.

We show an intuitive example by Figure 2 to illustrate the role of q and $|\Omega|$ in the error bound of Theorem 7. In the example, we use $d = 3$, $n = 100$, and $k = 20$ to generate synthetic random tensors, where we have let. $\|\boldsymbol{x}_j^{(i)}\| = \cdots = \|\boldsymbol{x}_j^{(d)}\| = \delta_j$ for convenience. We consider six different decay rates for $\delta_1, \ldots, \delta_k$ corresponding to different levels of low-rankness. In general, a larger decay rate means an easier tensor recovery problem. We let $\rho = \frac{|\Omega|}{n^d}$ and

$$\Delta = \frac{1}{|\Omega|}\sqrt{\frac{(n+\log(ek))(2-q)^2|\Omega|^{\frac{1-q}{2-q}}}{q(2-2q)^2}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\varepsilon\gamma_j}\right)^{\frac{2q}{2-q}}}, \tag{6}$$

where $0 < q < 1$. In the sub-figures (b,c,d), we see that a smaller q but not too small provides a tighter error bound. When the decay rate is smaller, reducing the value of q becomes more useful, which is consistent with the result of Figure 1. In addition, when ρ is smaller, namely the problem is harder, reducing the value of q becomes much more effective.

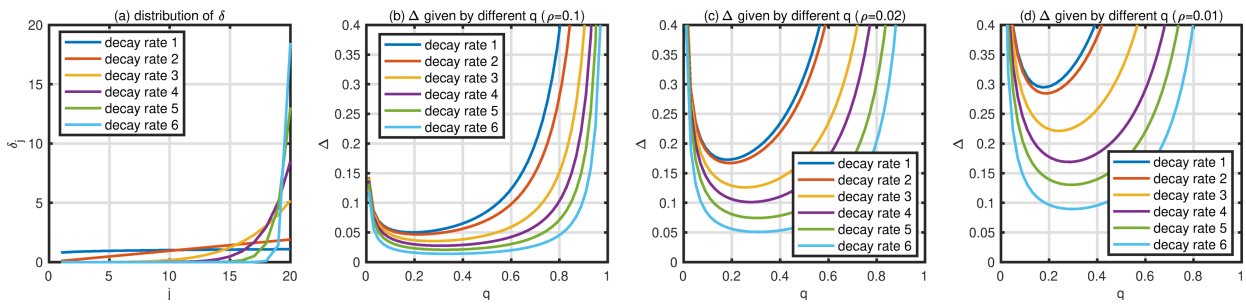

Figure 2: An intuitive exxample for the error bound of Theorem 7.

4 Tensor Robust PCA

4.1 TRPCA-ENR algorithm

Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ be a rank-r tensor. Suppose $\boldsymbol{\mathcal{T}}$ is corrupted as

$$\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{T}} + \boldsymbol{\mathcal{N}} + \boldsymbol{\mathcal{E}}, \tag{7}$$

where $\boldsymbol{\mathcal{N}}$ is a dense noise tensor drawn from $\mathcal{N}(0, \sigma^2)$ and $\boldsymbol{\mathcal{E}}$ is a sparse noise tensor with randomly distributed nonzero entries. To recover $\boldsymbol{\mathcal{T}}$ from $\boldsymbol{\mathcal{D}}$, we wish to solve

$$\underset{\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{E}}}{\text{minimize}}\ \frac{1}{2}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{E}}\|_F^2 + \lambda_x\|\boldsymbol{\mathcal{X}}\|_{S_p}^p + \lambda_e\|\boldsymbol{\mathcal{E}}\|_1, \tag{8}$$

but the problem is intractable. Then, based on the analysis in Section 2, we propose to solve

$$\underset{\{\boldsymbol{x}_i^{(j)}\},\ \boldsymbol{\mathcal{E}}}{\text{minimize}}\ \frac{1}{2}\left\|\boldsymbol{\mathcal{D}} - \sum_{i=1}^{k}\boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \ldots \circ \boldsymbol{x}_i^{(d)} - \boldsymbol{\mathcal{E}}\right\|_F^2 + \lambda_x\mathcal{R}\left(\{\boldsymbol{x}_i^{(j)}\}\right) + \lambda_e\|\boldsymbol{\mathcal{E}}\|_1, \tag{9}$$

where λ_e is a penalty parameter for the sparse tensor $\boldsymbol{\mathcal{E}}$. For convenience, we call (9) **TRPCA-ENR**.

When $p = 2/d$, we use alternating minimization to solve the optimization of TRPCA-ENR because every subproblem has a closed-form solution. When $p = 1/d$, we may, like Appendix A.1, use block coordinate descent with extrapolation (Xu & Yin, 2013). However, the additional variable $\boldsymbol{\mathcal{E}}$ further slows down the convergence. We hence propose to solve (9) by the (nonconvex) alternating direction method of multipliers (ADMM) (Wang et al., 2015). The optimization for (9) is detailed in Appendix B.

5 Experiments of LRTC

5.1 Synthetic data

We generate noisy low-rank synthetic tensors of size $50 \times 50 \times 50$ by $\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{T}} + \boldsymbol{\mathcal{N}}$, where $\boldsymbol{\mathcal{T}} = \sum_{i=1}^{r} \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \circ \boldsymbol{x}_i^{(3)}$. The entries of $\boldsymbol{x}_i^{(j)} \in \mathbb{R}^n$ ($i \in [r]$, $j \in [3]$) are drawn from $\mathcal{N}(0,1)$. The entries of the noise tensor $\boldsymbol{\mathcal{N}}$ are drawn from $\mathcal{N}(0, (0.1\sigma_{\boldsymbol{\mathcal{T}}})^2)$, where $\sigma_{\boldsymbol{\mathcal{T}}}$ denotes the standard deviation of the entries of $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{n \times n \times n}$. We set $n = 50$ and randomly remove a fraction (which we call missing rate) of the entries to test the performance of tensor completion, measured by

$$\text{Relative recovery error} = \|\mathcal{P}_{\bar{\Omega}}(\boldsymbol{\mathcal{T}} - \hat{\boldsymbol{\mathcal{T}}})\|_F / \|\mathcal{P}_{\bar{\Omega}}(\boldsymbol{\mathcal{T}})\|_F,$$

where $\bar{\Omega}$ denotes the locations of unknown entries and $\hat{\boldsymbol{\mathcal{T}}}$ denotes the recovered tensor.

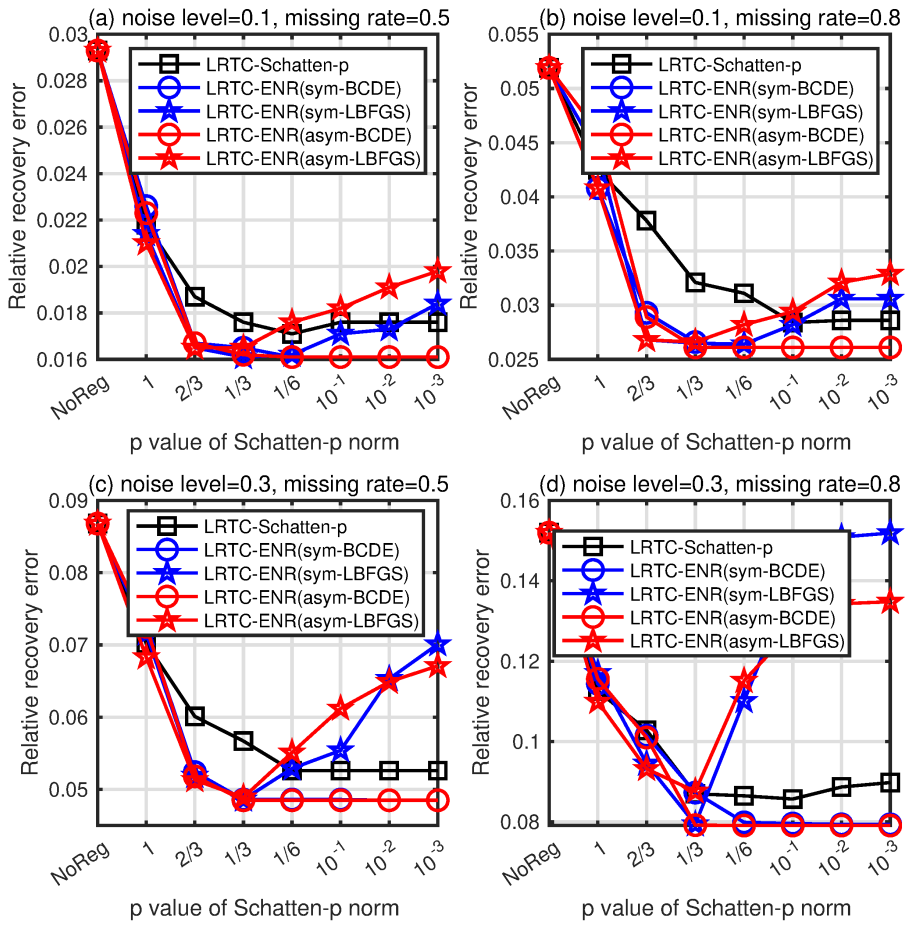

Figure 3: Performance (average of 20 repeated trials) of Schatten-p (quasi) norm with different p in the case of different noise level v for the noise distribution $\mathcal{N}(0, (v\sigma_{\boldsymbol{\mathcal{T}}})^2)$; "NoReg" means LRTC without regularization (Acar et al., 2011); the time costs (500 iterations) of the five methods are 8.6s, 3.0s, 2.1s, 2.8s, and 2.3s respectively).

We compare the performance of LRTC with Schatten-p (quasi) norm regularization (LRTC-Schatten-p) and Euclidean norm regularization (solved by BCDE and LBFGS) in Figure 3, in which we set $r = 10$ and considered different noise levels, missing rates, and p values. In all methods, we set $k = 2r = 20$, $t_{\max} = 500$, and use grid search (within $[0.01, 500]$) to determine λ. We have the following results.

a. *Regularization is helpful* In Figure 3 (a-d), the recovery error when there is no regularization (i.e. $\lambda = 0$) is higher than those of the regularized methods.

b. *Smaller p leads to lower recovery error?* In almost all cases, $p < 1$ outperforms $p = 1$, verifying the superiority of Schatten-p quasi-norm over nuclear norm in LRTC (both LRTC-ENR and LRTC-Schatten-p). In LRTC-ENR solved by BCDE and LBFGS, smaller p provides lower recovery error provided that p is not too small (e.g. $p \geq 1/6$), which matches Theorem 6 and Corollary 1.

c. *LRTC-ENR v.s. LRTC-Schatten-p* Compared to Euclidean norm minimization, direct Schatten-p norm minimization has higher recovery error and time cost because the optimization problem is more difficult to solve.

d. *BCDE v.s. LBFGS* When p is too small, the recovery error of LRTC-ENR solved by BCDE does not change but the recovery error of LRTC-ENR solved by LBFGS increases. One possible reason is that LBFGS is not effective in handling nonsmooth objective function and is often stuck in bad local minima or even saddle points, especially when p is very small. When $p < 1/3$, LRTC-ENR has at least one nonsmooth nonconvex subproblem, which brings difficulty to LBFGS. LBFGS is more efficient than BCDE because the latter is in the manner alternating update scheme.

e. *Symmetric regularizer v.s. asymmetric regularizer* For LRTC-ENR, the asymmetric regularization slightly outperforms the symmetric regularization in a few cases because the former has only one nonsmooth subproblem

Based on the above results, we suggest using BCDE to solve LRTC-ENR on small datasets and using LBFGS to solve LRTC-ENR on large datasets. In Schatten-p quasi norm, we suggest using $p = 1/3$ or $p = 1/6$.

We compare our tensor completion method LRTC-ENR with HaLRTC (Liu et al., 2012), TenALS (Jain & Oh, 2014), TMac (Xu et al., 2013), BCPF (Zhao et al., 2015), Rprecon (Kasai & Mishra, 2016), KBR-TC (Xie et al., 2018), TRLRF (Yuan et al., 2019). In TenALS, TMac, BCPF, Rprencon, TRLRF, and our LRTC-ENR, we need to determine the factorization size (or the initial rank in other word) beforehand. These methods, compared to HaLRTC and KBR-TC, may be applicable to big tensors efficiently provided that the factorization sizes are sufficiently small. Particularly, in TMac, BCPF, and our LRTC-ENR, the rank is adjusted adaptively. The tensor operation and computation are based on the MATLAB Tensor Tool Box (Bader et al., 2019).

Figure 4 shows the recovery of LRTC-ENR ($p = 1/3$, solved by LBFGS) and seven baselines on the synthetic data with $r = 10$. In all methods except HaLRTC and KBR-TC, one has to determine the initial rank beforehand. We have set the initial rank to $2r$ because in practice it is difficult to know the true rank. Though r is very small compared to the size of the tensor, the recovery errors of HaLRTC and Rprecon increase quickly when the missing rate is high. TMac, BCPF, and LRTC-ENR are able to adjust rank adaptively and the latter two perform much better than TMac.

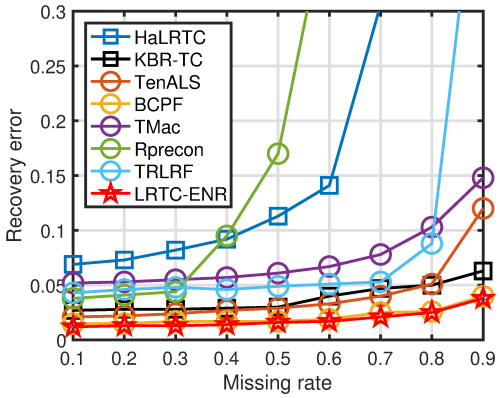

Figure 4: Low-rank tensor completion on synthetic data ($r = 10$): LRTC-ENR ($p = 1/3$) v.s. seven baseline methods

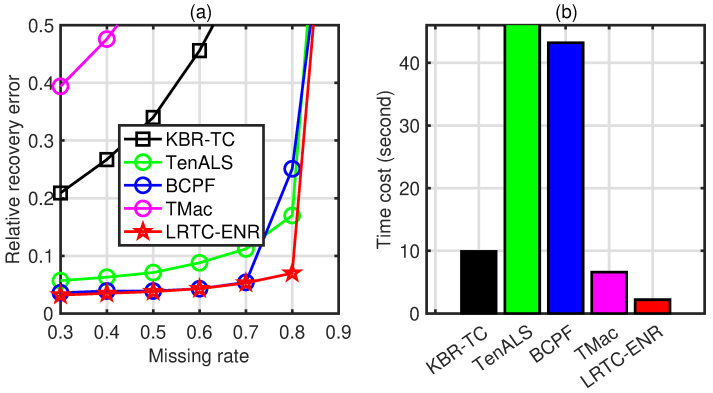

Figure 5: Low-rank tensor completion on synthetic data: LRTC-ENR ($p = 1/3$) v.s. baseline methods, $r = 50$; (a) recovery error with different missing rates; (b) average time cost in all cases of missing rates.

In Figure 4, when the missing rate is 0.9, we observed $12,500$ entries, much more than the minimum number (1500) of freedom parameters required to determine the tensor uniquely. It means the tensor completion problem is too easy. Hence we increase r to 50 and report the recovery error and computational time of KBR-TC, TenALS, BCPF, TMac, and LRTC-ENR in Figure 5. We see that the recovery error of BCPF and LRTC-ENR are lower than those of other methods. When the missing rate is sufficiently high, LRTC-ENR has smaller recovery error than BCPF. Moreover, LRTC-ENR (200 iterations) is at least 15 times faster than BCPF and 5 times faster than KBR-TC.

5.2 Multi-spectral image inpainting

We use the Columbia multi-spectral image database (Yasuma et al., 2008) to show the effectiveness of our tensor completion method in image inpainting problem. The dataset consists of the multi-spectral images of 32 real-world scenes of a variety of real-world materials and objects. The images have a spatial resolution of 512×512 and the number of spectral bands as 31. Thus the data of each image is a tensor of size $512 \times 512 \times 31$.

In our experiments, we pre-scale all images of every band to $[0, 1]$. Since TRLRF (Yuan et al., 2019) and Rprecon (Kasai & Mishra, 2016) have extremely high computational costs on these tensors, we omit their implementations here. In (Xie et al., 2018), it has been shown that BCPF (Zhao et al., 2015) significantly outperformed HaLRTC (Liu et al., 2012) as well as a few other baselines on this dataset, so we will not compare HaLRTC. The initial rank of TMac (Xu et al., 2013) and our LRTC-ENR ($\lambda = 0.1$) are set to 100. When the initial rank is 100, it takes TenALS (Jain & Oh, 2014) and BCPF (Zhao et al., 2015) more than 1500 seconds on each image, thus we set the initial rank to 50. Since the tensor rank is very small compared to the size and the images are noiseless, we here consider highly incomplete images of missing rate no less than 0.95.

As an example, Figure 6 shows the recovery errors of LRTC-ENR with different p and the compared methods on the first image of the dataset when the missing rate is 0.95, 0.97, and 0.99. It can be found that when the missing rate is 0.95, KBR-TC outperformed our LRTC-ENR. When the missing rate is 0.99, our LRTC-ENR outperformed all other methods. In Figure 6 (a), for LRTC-ENR, $p = 1/6$ is the best choice, while in Figure 6 (b) and (c), $p = 1/3$ has least recovery error. These results are consistent with the theorems.

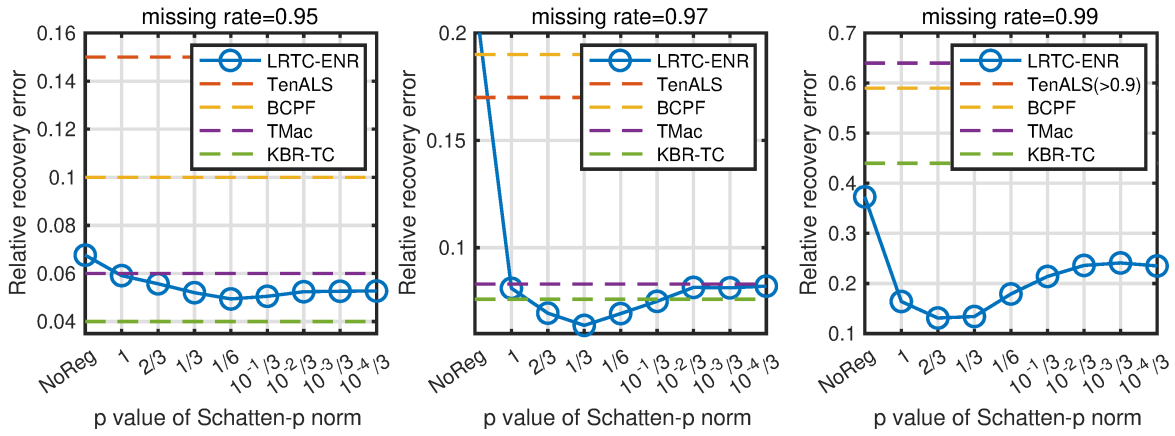

Figure 6: Recovery error on the first image of the MSI dataset.

Figure 7 and 8 visualize the recovery performance on the first image when the missing rates are 0.97 and 0.99 respectively. Visually, the recovery performance of our LRTC-ENR ($p = 1/3$) is much better than those of other methods. The average results (PSNR) on the 32 images are reported in Table 2. KBR-TC and LRTC-ENR ($p = 1/3$) outperformed other methods significantly. When the missing rate is 0.97 or 0.99, LRTC-ENR outperformed KBR-TC, and the improvement is significant according to the paired t-test. In addition, the time cost of LRTC-ENR is only 15% of that of KBR-TC.

6 Experiments of Tensor Robust PCA

6.1 Synthetic data

We generate synthetic tensors by $\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{T}} + \boldsymbol{\mathcal{N}} + \boldsymbol{\mathcal{E}}$. The low-rank tensor $\boldsymbol{\mathcal{T}}$ is given by $\boldsymbol{\mathcal{T}} = \sum_{i=1}^{r} w_i \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \circ \boldsymbol{x}_i^{(3)}$, where $w_i = i/r$ and the entries of $\boldsymbol{x}_i^{(j)} \in \mathbb{R}^n$ ($i \in [r]$, $j \in [3]$) are drawn from $\mathcal{N}(0, 1)$. $\boldsymbol{\mathcal{N}}$ is a dense noise tensor drawn from $\mathcal{N}(0, (0.1\sigma_{\boldsymbol{\mathcal{T}}})^2)$ and $\boldsymbol{\mathcal{E}}$ is a sparse noise tensor drawn from $\mathcal{N}(0, \sigma_{\boldsymbol{\mathcal{T}}}^2)$. We set $n = 50$, $r = 25$ and evaluate the TRPCA performance in recovering $\boldsymbol{\mathcal{T}}$ in the case of different sparsity (noise

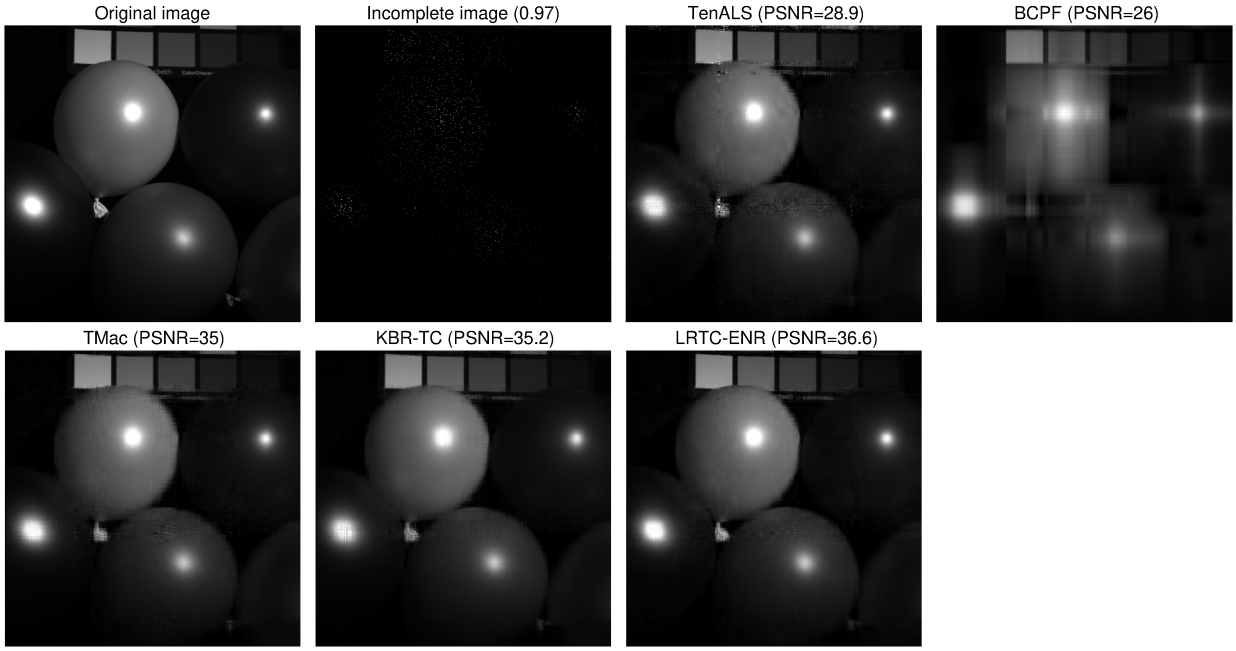

Figure 7: Inpainting performance on the first image (band 16) of the MSI dataset when the missing rate is 0.97.

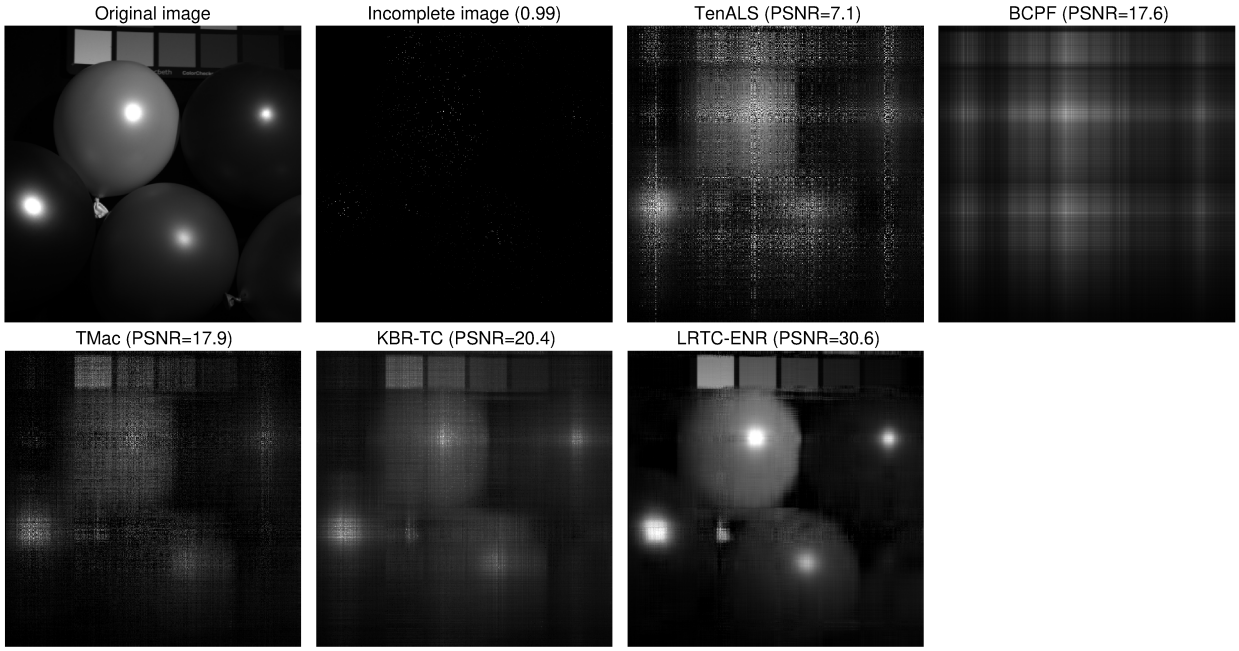

Figure 8: Inpainting performance on the first image (band 16) of the MSI dataset when the missing rate is 0.99.

density) of $\boldsymbol{\mathcal{E}}$. The evaluation metric is the relative recovery error defined by

$$\text{Relative recovery error} = \|\boldsymbol{\mathcal{T}} - \hat{\boldsymbol{\mathcal{T}}}\|_F / \|\boldsymbol{\mathcal{T}}\|_F.$$

Table 2: Average recovery performance (PSNR) on the Columbia MSI dataset of 32 images (the p-value is from the paired t-test between KBR-TC and LRTC-ENR)

Missing rate	0.95	0.97	0.99	Time
TenALS	30.1±4.4	25.9±3.22	13.3±4.9	500s
BCPF	30.4±5.8	26.8±5.6	20.2±3.3	450s
TMac	33.3±5.4	30.6±4.9	17.9±3.1	190s
KBR-TC	**37.1**±4.9	31.4±4.9	20.5±2.8	730s
LRTC-ENR	35.4±4.4	**32.4**±5.1	**26.1**±5.4	**110**s
p-value (t-test)	5×10^{-11}	5×10^{-4}	2×10^{-8}	0

We compare our TRPCA-ENR with KBR-PCA (Xie et al., 2018). The TRPCA (denoted by T-TRPCA) proposed by (Lu et al., 2019) and the robust tensor decomposition method OITNN-L proposed by (Wang et al., 2020) are under the assumption of t-product and hence not suitable in our setting of synthetic data. We will compare them on real data. The hyper-parameters of KBR-RPCA and our TRPCA-ENR[2] are sufficiently tuned to provide their best performance. Figure 9 (a) shows the recovery errors of KBR-RPCA and TRPCA-ENR (with $p = 2/3$, $1/3$, and $1/6$) when the noise density increased from 0.1 to 0.7. We see that TRPCA-ENR outperformed KBR-RPCA consistently. In TRPCA-ENR, $p = 1/6$ is better than $p = 2/3$ and $1/3$ but the difference is not obvious. In Figure 9 (b), we show the performance of different p's. It can be found that a smaller p (but not too small, e.g. larger than 10^{-2}) yields less recovery error.

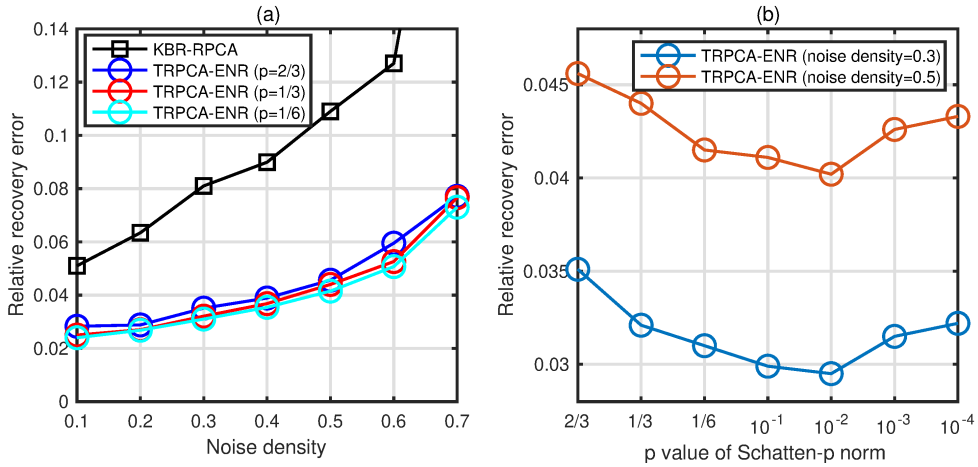

Figure 9: Performance of TRPCA on synthetic data (average of 20 repeated trials): (a) KBR-RPCA v.s. TRPCA-ENR (initial rank=2r); (b) relative recovery error of TRPCA-ENR with different initial rank (noise density=0.5).

6.2 Color image denoising

We compare our TRPCA-ENR with KBR-PCA (Xie et al., 2018), T-TRPCA (Lu et al., 2019), and OITNN-L (Wang et al., 2020) in the task of color image denoising on nine color images (shown in (10)) of size $256 \times 256 \times 3$ used in (Wang et al., 2020). For each image, we randomly set 10% of the tensor elements to random values in $[0, 1]$. We tuned all hyper-parameters carefully to provided the best denoising performance of the methods. As a result, in KBR-RPCA, we set $\lambda = 2$ or 3. In OITNN-L, we set $\lambda_L = 4$ or 5 and $\lambda_S = 0.16$. In T-TRPCA, we set $\lambda = 1/\sqrt{768}$ or $2/\sqrt{768}$. In our TRPCA-ENR, we set the initial rank to a relative large value 200 because the approximate rank of the image tensors is not too small and our methods can adjust the rank adaptively; we set $\lambda_e = 0.02$ and consider the cases of $p = 2/3$, $1/3$, $1/6$, and $1/10$, for which $\lambda_x = 0.1$, 0.1, 0.02, and 0.015.

[2]Throughout this paper, we set $\mu = 10$ and $t_{\max} = 500$ for Algorithms 4 and 5.

Figure 10: Nine color images for denoising

The average PSNR with standard deviation of 20 repeated trials are reported in Tables 3. We see that the PSNRs of our TRPCA-ENR methods are much lower than those of other methods in all cases. In addition, TRPCA-ENR with smaller p has higher PSNR and TRPCA-ENR with $p = 1/10$ performs the best. Figure 11 visualizes the denoising performance on Image 1, showing that TRPCA-ENR methods are better than other methods. These results provided evidence that tensor Schatten-p quasi-norm minimization is able to provide higher recovery accuracy in TRPCA when a relatively smaller p is used.

Table 3: PSNR of color image denoising

	Image 1	Image 2	Image 3	Image 4	Image5
KBR-RPCA (Xie et al., 2018)	30.25±0.06	33.36±0.22	31.93±0.12	31.87±0.08	35.08±0.12
T-TRPCA (Lu et al., 2019)	31.16±0.05	32.75±0.06	32.87±0.11	32.52±0.06	33.91±0.10
OITNN-O (Wang et al., 2020)	28.46±0.04	29.99±0.03	29.16±0.05	28.13±0.04	29.86±0.03
TRPCA-ENR (p=$\frac{2}{3}$)	34.23±0.12	35.44±0.09	34.82±0.18	34.23±0.14	37.14±0.06
TRPCA-ENR (p=$\frac{1}{3}$)	34.68±0.18	36.88±0.16	37.52±0.12	34.53±0.31	37.78±0.04
TRPCA-ENR (p=$\frac{1}{6}$)	34.96±0.40	36.04±1.08	37.57±0.33	34.46±1.21	38.14±0.11
TRPCA-ENR (p=$\frac{1}{10}$)	**35.45**±0.12	**37.14**±0.10	**37.76**±0.14	**35.38**±0.46	**38.28**±0.13
	Image 6	Image 7	Image 8	Image 9	
KBR-RPCA (Xie et al., 2018)	31.36±0.09	30.08±0.10	32.41±0.05	34.28±0.12	
T-TRPCA (Lu et al., 2019)	28.96±0.06	29.36±0.09	31.49±0.11	33.67±0.14	
OITNN-O (Wang et al., 2020)	27.39±0.05	27.46±0.05	28.61±0.07	30.06±0.06	
TRPCA-ENR (p=$\frac{2}{3}$)	35.09±0.12	33.67±0.10	35.66±0.16	36.12±0.05	
TRPCA-ENR (p=$\frac{1}{3}$)	33.21±0.54	34.10±0.23	36.09±0.14	37.95±0.16	
TRPCA-ENR (p=$\frac{1}{6}$)	31.93±0.98	34.44±0.12	36.49±0.08	37.96±0.10	
TRPCA-ENR (p=$\frac{1}{10}$)	**35.18**±0.35	**34.57**±0.14	**36.71**±0.06	**38.01**±0.16	

7 Conclusion

We have proposed a new class of nonconvex rank regularizers based on the Euclidean norms of component vectors in CP decomposition. These regularizers are monotonic transformations of tensor nuclear norm and Schatten-p quasi-norms. We proved that Schattern-p quasi-norm with a smaller p but not too small provides tighter error bounds for LRTC. Numerical results on synthetic data and natural images demonstrated the advantages of our methods in LRTC and TRPCA against state-of-the-art methods. Future work may focus on the theoretical guarantee of TRPCA-ENR.

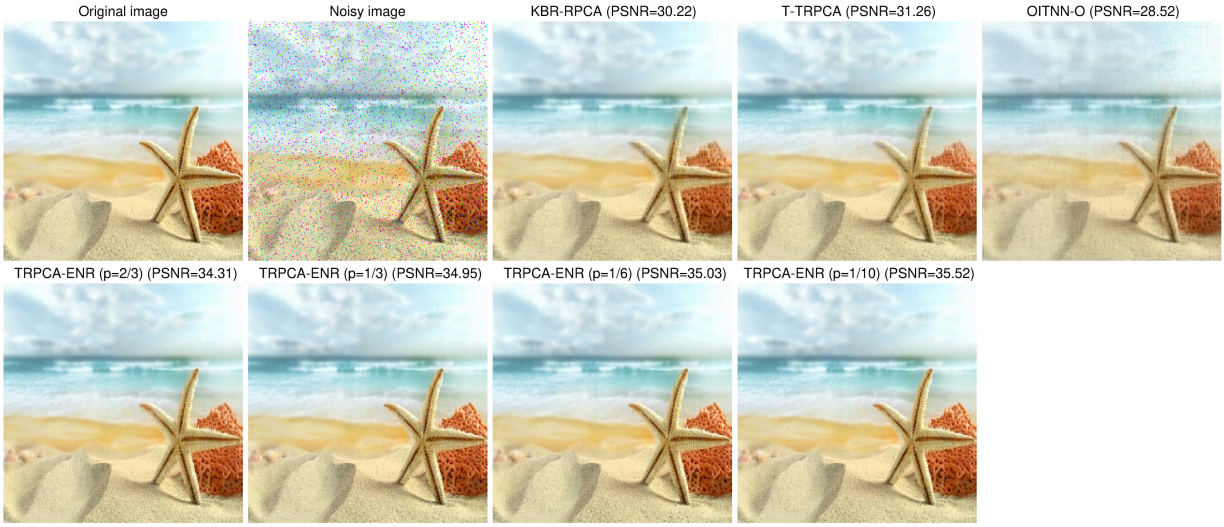

Figure 11: Denoising performance on Image 1.

References

Evrim Acar, Daniel M Dunlavy, Tamara G Kolda, and Morten Mørup. Scalable tensor factorizations for incomplete data. *Chemometrics and Intelligent Laboratory Systems*, 106(1):41–56, 2011.

Anima Anandkumar, Prateek Jain, Yang Shi, and Uma Naresh Niranjan. Tensor vs. matrix methods: Robust tensor decomposition under block sparse perturbations. In *Artificial Intelligence and Statistics*, pp. 268–276, 2016.

Brett W. Bader, Tamara G. Kolda, et al. Matlab tensor toolbox version 3.1. Available online, June 2019. URL `https://www.tensortoolbox.org`.

Boaz Barak and Ankur Moitra. Noisy tensor completion via the sum-of-squares hierarchy. In *Conference on Learning Theory*, pp. 417–445, 2016.

Peter Bartlett, Dylan J Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1706.08498*, 2017.

Juan Andrés Bazerque, Gonzalo Mateos, and Georgios B Giannakis. Rank regularization and bayesian inference for tensor completion and extrapolation. *IEEE transactions on signal processing*, 61(22):5689–5703, 2013.

Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

Emmanuel J. Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009. ISSN 1615-3383. doi: 10.1007/s10208-009-9045-5.

Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):1–37, 2011. ISSN 0004-5411. doi: 10.1145/1970392.1970395.

Hao Cheng, Yaoliang Yu, Xinhua Zhang, Eric Xing, and Dale Schuurmans. Scalable and sound low-rank tensor learning. In Arthur Gretton and Christian C. Robert (eds.), *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 1114–1123, Cadiz, Spain, 09–11 May 2016. PMLR.

Fernando De la Torre and Michael J Black. Robust principal component analysis for computer vision. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 1, pp. 362–369. IEEE, 2001.

Chris Ding, Ding Zhou, Xiaofeng He, and Hongyuan Zha. R 1-pca: rotational invariant l 1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd international conference on Machine learning*, pp. 281–288, 2006.

Ran El-Yaniv and Dmitry Pechyony. Transductive rademacher complexity and its applications. *Journal of Artificial Intelligence Research*, 35:193–234, 2009.

Jicong Fan. Multi-mode deep matrix and tensor factorization. In *International Conference on Learning Representations*, 2022.

Jicong Fan, Lijun Ding, Yudong Chen, and Madeleine Udell. Factor group-sparse regularization for efficient low-rank matrix recovery. In *Advances in Neural Information Processing Systems 32*, pp. 5104–5114. Curran Associates, Inc., 2019.

Dylan J. Foster and Andrej Risteski. Sum-of-squares meets square loss: Fast rates for agnostic tensor completion. In Alina Beygelzimer and Daniel Hsu (eds.), *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pp. 1280–1318. PMLR, 25–28 Jun 2019. URL `https://proceedings.mlr.press/v99/foster19a.html`.

Shmuel Friedland and Lek-Heng Lim. Nuclear norm of higher-order tensors. *Mathematics of Computation*, 87(311):1255–1281, 2018.

Silvia Gandy, Benjamin Recht, and Isao Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010, 2011.

Navid Ghadermarzy, Yaniv Plan, and Özgür Yilmaz. Near-optimal sample complexity for convex tensor completion. *Information and Inference: A Journal of the IMA*, 8(3):577–619, 2019.

Paris Giampouras, René Vidal, Athanasios Rontogiannis, and Benjamin Haeffele. A novel variational form of the schatten-p quasi-norm. *arXiv preprint arXiv:2010.13927*, 2020.

Donald Goldfarb and Zhiwei Qin. Robust low-rank tensor recovery: Models and algorithms. *SIAM Journal on Matrix Analysis and Applications*, 35(1):225–253, 2014.

Benjamin D Haeffele and René Vidal. Structured low-rank matrix factorization: Global optimality, algorithms, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1468–1482, 2019.

Moritz Hardt. Understanding alternating minimization for matrix completion. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS),*, pp. 651–660. IEEE, 2014.

Aicke Hinrichs, Joscha Prochno, and Jan Vybíral. Entropy numbers of embeddings of schatten classes. *Journal of Functional Analysis*, 273(10):3241–3261, 2017. ISSN 0022-1236. doi: https://doi.org/10.1016/j.jfa.2017.08.008. URL `https://www.sciencedirect.com/science/article/pii/S0022123617303221`.

Bo Huang, Cun Mu, Donald Goldfarb, and John Wright. Provable models for robust low-rank tensor completion. *Pacific Journal of Optimization*, 11(2):339–364, 2015.

Prateek Jain and Sewoong Oh. Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems*, pp. 1431–1439, 2014.

Hiroyuki Kasai and Bamdev Mishra. Low-rank tensor completion: a riemannian manifold preconditioning approach. In *International Conference on Machine Learning*, pp. 1012–1021, 2016.

Misha E Kilmer and Carla D Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 2011.

Hao Kong, Xingyu Xie, and Zhouchen Lin. t-schatten-p norm for low-rank tensor recovery. *IEEE Journal of Selected Topics in Signal Processing*, 12(6):1405–1419, 2018.

Daniel Kressner, Michael Steinlechner, and Bart Vandereycken. Low-rank tensor completion by riemannian optimization. *BIT Numerical Mathematics*, 54(2):447–468, 2014.

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. Canonical tensor decomposition for knowledge base completion. In *International Conference on Machine Learning*, pp. 2863–2872. PMLR, 2018.

Allen Liu and Ankur Moitra. Tensor completion made practical. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18905–18916. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/dab1263d1e6a88c9ba5e7e294def5e8b-Paper.pdf`.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):208–220, 2012.

Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5249–5257, 2016.

Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE transactions on pattern analysis and machine intelligence*, 42(4):925–938, 2019.

Zhaosong Lu. Iterative reweighted minimization methods for l_p regularized unconstrained nonlinear programming. *Mathematical Programming*, 147(1-2):277–307, 2014.

Sebastian Mayer and Tino Ullrich. Entropy numbers of finite dimensional mixed-norm balls and function space embeddings with small mixed smoothness. *Constructive Approximation*, 53(2):249–279, 2021.

Feiping Nie, Heng Huang, and Chris Ding. Low-rank matrix recovery via efficient Schatten p-norm minimization. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pp. 655–661. AAAI Press, 2012.

Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3): 127–239, 2014.

Aaron Potechin and David Steurer. Exact tensor completion with sum-of-squares. In Satyen Kale and Ohad Shamir (eds.), *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pp. 1619–1673. PMLR, 07–10 Jul 2017.

Bernardino Romera-Paredes and Massimiliano Pontil. A new convex relaxation for tensor completion. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, pp. 2967–2975, 2013.

Nicolas Schreuder. Bounding the expectation of the supremum of empirical processes indexed by h\" older classes. *arXiv preprint arXiv:2003.13530*, 2020.

Fanhua Shang, Yuanyuan Liu, and James Cheng. Tractable and scalable Schatten quasi-norm approximations for rank minimization. In *Artificial Intelligence and Statistics*, pp. 620–629, 2016.

Fanhua Shang, James Cheng, Yuanyuan Liu, Zhi-Quan Luo, and Zhouchen Lin. Bilinear factor matrix norm minimization for robust pca: Algorithms and applications. *IEEE transactions on pattern analysis and machine intelligence*, 40(9):2066–2080, 2017.

Qiquan Shi, Haiping Lu, and Yiu-ming Cheung. Tensor rank estimation and completion via cp-based nuclear norm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 949–958, 2017.

Nathan Srebro and Adi Shraibman. Rank, trace-norm and max-norm. In *International Conference on Computational Learning Theory*, pp. 545–560. Springer, 2005.

Andong Wang, Chao Li, Zhong Jin, and Qibin Zhao. Robust tensor decomposition via orientation invariant tubal nuclear norms. In *AAAI*, pp. 6102–6109, 2020.

Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, pp. 1–35, 2015.

Q. Xie, Q. Zhao, D. Meng, and Z. Xu. Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(8): 1888–1902, 2018.

Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.

Yangyang Xu, Ruru Hao, Wotao Yin, and Zhixun Su. Parallel matrix factorization for low-rank tensor completion. *arXiv preprint arXiv:1312.1254*, 2013.

Bo Yang, Gang Wang, and Nicholas D Sidiropoulos. Tensor completion via group-sparse regularization. In *2016 50th Asilomar Conference on Signals, Systems and Computers*, pp. 1750–1754. IEEE, 2016.

F. Yasuma, T. Mitsunaga, D. Iso, and S.K. Nayar. Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum. Technical report, Nov 2008.

Longhao Yuan, Chao Li, Danilo Mandic, Jianting Cao, and Qibin Zhao. Tensor ring decomposition with rank minimization on latent space: An efficient approach for tensor completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 9151–9158, 2019.

Ming Yuan and Cun-Hui Zhang. On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*, 16(4):1031–1068, 2016.

Zemin Zhang and Shuchin Aeron. Exact tensor completion using t-svd. *IEEE Transactions on Signal Processing*, 65(6):1511–1526, 2016.

Qibin Zhao, Liqing Zhang, and Andrzej Cichocki. Bayesian cp factorization of incomplete tensors with automatic rank determination. *IEEE transactions on pattern analysis and machine intelligence*, 37(9): 1751–1763, 2015.

Yu-Bang Zheng, Ting-Zhu Huang, Xi-Le Zhao, Tai-Xiang Jiang, Tian-Hui Ma, and Teng-Yu Ji. Mixed noise removal in hyperspectral image via low-fibered-rank regularization. *IEEE Transactions on Geoscience and Remote Sensing*, 58(1):734–749, 2019.

Pan Zhou, Canyi Lu, Zhouchen Lin, and Chao Zhang. Tensor factorization for low-rank tensor completion. *IEEE Transactions on Image Processing*, 27(3):1152–1163, 2017.

A Optimization for LRTC-ENR

A.1 Block Coordinate Descent with Extrapolation

There are d blocks of decision variables in Problem (2), i.e. $\{\boldsymbol{X}^{(j)}\}^{j\in[d]}$, where $\boldsymbol{X}^{(j)} = [\boldsymbol{x}_1^{(j)}, \boldsymbol{x}_2^{(j)}, \ldots, \boldsymbol{x}_k^{(j)}]$. We propose to find a critical point of (2) by Block Coordinate Descent (BCD) with Extrapolation (BCDE for short) (Xu & Yin, 2013), which is more efficient than BCD. For simplicity, here we only present the optimization of (2) with the regularizer shown in (3). The optimization can be easily extended to (2) with other regularizers we proposed in Section 2.

Let

$$\begin{aligned}\mathcal{L}\left(\{\boldsymbol{X}^{(j)}\}^{j\in[d]}\right) :=& \frac{1}{2}\left\|\boldsymbol{\mathcal{M}} * \left(\boldsymbol{\mathcal{D}} - \sum_{i=1}^{k} \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \ldots \circ \boldsymbol{x}_i^{(d)}\right)\right\|_F^2 \\ =& \frac{1}{2}\left\|\boldsymbol{M}_{(j)} * \left(\boldsymbol{D}_{(j)} - \boldsymbol{X}^{(j)}[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^\top\right)\right\|_F^2\end{aligned}$$

and

$$\mathcal{R}_{pd}(\boldsymbol{X}^{(j)}) := \sum_{j=1}^{d} \|\boldsymbol{x}_i^{(j)}\|^{pd},$$

where $[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}] = \boldsymbol{X}^{(d)} \odot \ldots \odot \boldsymbol{X}^{(j+1)} \odot \boldsymbol{X}^{(j-1)} \odot \ldots \odot \boldsymbol{X}^{(1)}$. We initialize $\{\boldsymbol{X}^{(j)}\}^{j\in[d]}$ randomly and rescale $\boldsymbol{x}$ to have unit Euclidean norm:

$$\boldsymbol{x}_i^{(j)} \sim \mathcal{N}(0,1), \quad \boldsymbol{x}_i^{(j)} \leftarrow \boldsymbol{x}_i^{(j)}/\|\boldsymbol{x}_i^{(j)}\|, \quad i \in [k],\ j \in [d]. \tag{10}$$

Then at iteration t, for $j = 1, 2, \ldots, d$, we first perform an extrapolation

$$\hat{\boldsymbol{X}}_{t-1}^{(j)} = \boldsymbol{X}_{t-1}^{(j)} + \omega_{j,t-1}(\boldsymbol{X}_{t-1}^{(j)} - \boldsymbol{X}_{t-2}^{(j)}), \tag{11}$$

where $\omega_{j,t-1} \geq 0$ controls the size of the extrapolation at iteration t. Then we perform a proximal step as

$$\begin{aligned}\boldsymbol{X}_t^{(j)} =& \operatorname*{argmin}_{\boldsymbol{X}^{(j)}} \left\langle \boldsymbol{\nabla}_{\hat{\boldsymbol{X}}_{t-1}^{(j)}} \mathcal{L},\ \boldsymbol{X}^{(j)} - \hat{\boldsymbol{X}}_{t-1}^{(j)} \right\rangle + \frac{\hat{L}_{j,t}}{2}\|\boldsymbol{X}^{(j)} - \hat{\boldsymbol{X}}_{t-1}^{(j)}\|_F^2 + \lambda \mathcal{R}_{pd}(\boldsymbol{X}^{(j)}) \\ \triangleq& \operatorname{prox}_{pd}^{\lambda}\Big(\hat{\boldsymbol{X}}_{t-1}^{(j)} - \hat{L}_{j,t}^{-1}\boldsymbol{\nabla}_{\hat{\boldsymbol{X}}_{t-1}^{(j)}}\mathcal{L}\Big).\end{aligned} \tag{12}$$

In (12), $\boldsymbol{\nabla}_{\hat{\boldsymbol{X}}_{t-1}^{(j)}}\mathcal{L} = \left(\boldsymbol{M}_{(j)} * \left(\boldsymbol{D}_{(j)} - \hat{\boldsymbol{X}}_{t-1}^{(j)}[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top\right)\right)\left(-[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\right)$, $[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}] = \boldsymbol{X}_{t-1}^{(d)} \odot \cdots \odot \boldsymbol{X}_{t-1}^{(j+1)} \odot \boldsymbol{X}_t^{(j-1)} \odot \cdots \odot \boldsymbol{X}_t^{(1)}$, and $\hat{L}_{j,t}$ is larger than the Lipschitz constant of $\boldsymbol{\nabla}_{\hat{\boldsymbol{X}}_{t-1}^{(j)}}\mathcal{L}$. We approximate it by

$$\hat{L}_{j,t} = \varrho\sqrt{\frac{|\Omega|}{\prod_{i=1}^{d} n_i}}\left\|[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\right\|_2^2, \tag{13}$$

where $\varrho = 0.5$, 1, or 2. The derivation of (13) is as follows. For convenience, we denote a perturbed copy of $\hat{\boldsymbol{X}}_{t-1}^{(j)}$ by $\hat{\boldsymbol{Z}}_{t-1}^{(j)}$. We have

$$\begin{aligned}
&\|\boldsymbol{\nabla}_{\hat{\boldsymbol{X}}_{t-1}^{(j)}}\mathcal{L} - \boldsymbol{\nabla}_{\hat{\boldsymbol{Z}}_{t-1}^{(j)}}\mathcal{L}\|_F \\
=&\| \Big(\boldsymbol{M}_{(j)} * \big(\boldsymbol{D}_{(j)} - \hat{\boldsymbol{X}}_{t-1}^{(j)}[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top\big)\Big)\big(-[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\big) \\
&- \Big(\boldsymbol{M}_{(j)} * \big(\boldsymbol{D}_{(j)} - \hat{\boldsymbol{Z}}_{t-1}^{(j)}[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top\big)\Big)\big(-[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\big)\|_F \\
\leq&\|[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\|_2\|\boldsymbol{M}_{(j)} * \big(\hat{\boldsymbol{X}}_{t-1}^{(j)}[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top - \hat{\boldsymbol{Z}}_{t-1}^{(j)}[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top\big)\|_F \\
\leq&\|[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\|_2\varrho\sqrt{\frac{|\Omega|}{\prod_{i=1}^d n_i}}\|\big(\hat{\boldsymbol{X}}_{t-1}^{(j)} - \hat{\boldsymbol{Z}}_{t-1}^{(j)}\big)[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]^\top\|_F \\
\leq&\varrho\sqrt{\frac{|\Omega|}{\prod_{i=1}^d n_i}}\|[(\boldsymbol{X}^{(i)})_t^{\odot_{i\neq j}}]\|_2^2\|\hat{\boldsymbol{X}}_{t-1}^{(j)} - \hat{\boldsymbol{Z}}_{t-1}^{(j)}\|_F.
\end{aligned}$$

Here $0 < \varrho \leq \sqrt{\dfrac{\prod_{i=1}^d n_i}{|\Omega|}}$ is some suitable constant.

In (12), when $p = 1/d$, $\mathrm{prox}_1^\lambda\big(\boldsymbol{Y}\big) = \Phi_\lambda(\boldsymbol{Y})$, where Φ is the column-wise soft-thresholding operator (Parikh et al., 2014) defined by

$$\Phi_\lambda(\boldsymbol{y}) = \begin{cases} \frac{(\|\boldsymbol{y}\|-\lambda)\boldsymbol{y}}{\|\boldsymbol{y}\|}, & \text{if } \|\boldsymbol{y}\| > \lambda; \\ \boldsymbol{0}, & \text{otherwise.} \end{cases}$$

When $p = 2/d$, $\mathrm{prox}_2^\lambda\big(\boldsymbol{Y}\big) = \dfrac{\hat{L}_{j,t}}{\hat{L}_{j,t} + 2\lambda}\boldsymbol{Y}$. When $p > 2/d$, we can estimate $\boldsymbol{X}_t^{(j)}$ by gradient descent. When $p < 1/d$, (12) is nonconvex and nonsmooth. Then we update $\boldsymbol{X}^{(j)}$ with iteratively reweighted method (Lu, 2014), which is given by Algorithm 1.

Algorithm 1 $\min_{\boldsymbol{Y}} \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{G}\|_F^2 + \tilde{\lambda}\sum_{i=1}^k \|\boldsymbol{y}_i\|^q$

Require: $\boldsymbol{G}$, q, $\tilde{\lambda}$, t_q, ϵ.
1: $\boldsymbol{Y} \leftarrow \boldsymbol{G}$.
2: **for** $t = 1, 2, \ldots, t_q$ **do**
3: $\quad\boldsymbol{W} = \mathrm{diag}\big((\|\boldsymbol{y}_1\| + \epsilon)^{\frac{q-2}{2}}, \ldots, (\|\boldsymbol{y}_k\| + \epsilon)^{\frac{q-2}{2}}\big)$.
4: $\quad\boldsymbol{Y} = \boldsymbol{G}(\boldsymbol{I} + 2\tilde{\lambda}\boldsymbol{W}\boldsymbol{W}^T)^{-1}$.
5: **end for**
Ensure: $\boldsymbol{Y}$.

In (11), the parameter ω_{jt} is determined as

$$\omega_{j,t-1} = \delta\sqrt{\hat{L}_{j,t-2}/\hat{L}_{j,t-1}}, \tag{14}$$

where $\delta < 1$. We set $\delta = 0.95$ for simplicity. The whole procedure is summarized in Algorithm 2. The convergence analysis when $p \geq 1/d$ can be found in (Xu & Yin, 2013). When $p < 1/d$, suppose Algorithm 1 returns the optimal solution, we can get similar convergence result as the case $p = 1/d$. Empirically, we found that there is no need to find the exact solution for the subproblem and instead we just perform Algorithm 1 for a few iterations, which can still provide satisfactory result. Currently, proving the convergence is out of the scope of our paper.

A.2 L-BFGS

Though faster than BCD, the computational cost per iteration of BCDE is high when d is not small because we need to compute $\boldsymbol{X}^{(j)}[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^\top$ for d times in every iteration. One may consider the Jacobi-type

Algorithm 2 solve LRTC-ENR by BCDE

Input: $\boldsymbol{\mathcal{D}}$, $\boldsymbol{\mathcal{M}}$, k, λ, $t_{\max}$.

Initialize $\{\boldsymbol{x}_i^{(j)}\}_{i\in[k]}^{j\in[d]}$ with (10), let $t=0$.
repeat
  $t \leftarrow t+1$.
  for $j=1,2,\ldots,d$ **do**
    if $t <= 2$ **then**
      $\omega_{j,t-1}=0$.
    else
      Compute $\omega_{j,t-1}$ by (14).
    end if
    Compute $\hat{\boldsymbol{X}}_{t-1}^{(j)}$ by (11).
    Compute $L_{j,t-1}$ by (13).
    Compute $\boldsymbol{X}_t^{(j)}$ by (12).
  end for
  Remove the zero columns of $\boldsymbol{X}_t^{(j)}$, $j\in[d]$.
until converged or $t=t_{\max}$

Output: $\hat{\boldsymbol{\mathcal{T}}}=\boldsymbol{\mathcal{I}}\times_1\boldsymbol{X}_t^{(1)}\times_2\boldsymbol{X}_t^{(2)}\ldots\times_d\boldsymbol{X}_t^{(d)}$.

iteration (cheaper computation but slower convergence) rather than the Gauss-Seidel iteration in BCD and BCDE. In practice, we can use quasi-Newton methods such as L-BFGS (Liu & Nocedal, 1989) to solve problem (2) even though the objective function is nonsmooth. Particularly, we can drop the columns of $\boldsymbol{X}_t^{(j)}$ ($j\in[d]$) with nearly-zero Euclidean norms for acceleration. The corresponding algorithm[3] is shown in Algorithm 3.

Algorithm 3 solve LRTC-ENR by LBFGS

Input: $\boldsymbol{\mathcal{D}}$, $\boldsymbol{\mathcal{M}}$, k, λ, $t_{\max}$.

Initialize $\{\boldsymbol{x}_i^{(j)}\}_{i\in[k]}^{j\in[d]}$ with (10), let $t=0$.
repeat
  $t \leftarrow t+1$.
  Compute the search directions by LBFGS.
  Use line search to determine the step size.
  Update $\boldsymbol{X}_t^{(j)}$, $j\in[d]$.
  For $j\in[d]$, remove the columns of $\boldsymbol{X}_t^{(j)}$ with Euclidean norms less than a small threshold, e.g. 10^{-5}.
until converged or $t=t_{\max}$

Output: $\hat{\boldsymbol{\mathcal{T}}}=\boldsymbol{\mathcal{I}}\times_1\boldsymbol{X}_t^{(1)}\times_2\boldsymbol{X}_t^{(2)}\ldots\times_d\boldsymbol{X}_t^{(d)}$.

B Optimization for TRPCA-ENR

When $p=2/d$, we use alternating minimization to solve the optimization of TRPCA-ENR because every subproblem has a closed-form solution. When $p=1/d$, we may, like Section A.1, use block coordinate descent with extrapolation (Xu & Yin, 2013). However, the additional variable $\boldsymbol{\mathcal{E}}$ further slows down the convergence. We hence propose to solve (9) by the (nonconvex) alternating direction method of multipliers (ADMM) (Wang et al., 2015). Specifically, by adding auxiliary variables $\{\boldsymbol{Y}^{(j)}\}_{j=1}^d$, we reformulate (9) as

$$\begin{aligned}\underset{\{\boldsymbol{X}^{(j)},\boldsymbol{Y}^{(j)}\}_{j=1}^d,\boldsymbol{\mathcal{E}}}{\text{minimize}} \quad & \frac{1}{2}\left\|\boldsymbol{D}_{(j)}-\boldsymbol{X}^{(j)}[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^\top-\boldsymbol{E}_{(j)}\right\|_F^2+\lambda_x\mathcal{R}\left(\{\boldsymbol{y}_i^{(j)}\}_{i\in[k]}^{j\in[d]}\right)+\lambda_e\|\boldsymbol{\mathcal{E}}\|_1\\ \text{subject to} \quad & \boldsymbol{Y}^{(j)}=\boldsymbol{X}^{(j)}, j=1,2,\ldots,d.\end{aligned}$$

[3]The implementation in this paper is based on the *minFunc* MATLAB toolbox of M. Schmidt: `http://www.cs.ubc.ca/~schmidtm/Software/minFunc.html`.

Let $\{\boldsymbol{Z}^{(j)}\}_{j=1}^d$ be Lagrange multipliers and solve

$$\begin{aligned}\underset{\{\boldsymbol{X}^{(j)},\boldsymbol{Y}^{(j)}\}_{j=1}^d,\boldsymbol{\mathcal{E}}}{\text{minimize}} \quad & \frac{1}{2}\left\|\boldsymbol{D}_{(j)} - \boldsymbol{X}^{(j)}[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^\top - \boldsymbol{E}_{(j)}\right\|_F^2 \\ & + \lambda_x \sum_{i=1}^k \sum_{j=1}^d \|\boldsymbol{y}_i^{(j)}\|^{pd} + \lambda_e \|\boldsymbol{\mathcal{E}}\|_1 + \sum_{j=1}^d \left\langle \boldsymbol{Y}^{(j)} - \boldsymbol{X}^{(j)}, \boldsymbol{Z}^{(j)} \right\rangle + \frac{\mu}{2}\|\boldsymbol{Y}^{(j)} - \boldsymbol{X}^{(j)}\|_F^2,\end{aligned} \tag{15}$$

where μ is the augmented Lagrange penalty parameter. Then update $\{\boldsymbol{X}^{(j)}, \boldsymbol{Y}^{(j)}\}_{j=1}^d$ and $\boldsymbol{\mathcal{E}}$ sequentially to minimize (15) and update $\{\boldsymbol{Z}^{(j)}\}_{j=1}^d$ lastly. The procedures are summarized into Algorithm 4, where

$$\boldsymbol{X}_t^{(j)} = \Big((\boldsymbol{D}_{(j)} - \boldsymbol{E}_{(j)})[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}] + \mu \boldsymbol{Y}_{t-1}^{(j)} - \boldsymbol{Z}^{(j)}\Big)\Big([(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^T[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}] + \mu \boldsymbol{I}\Big)^{-1}, \tag{16}$$

$$\boldsymbol{Y}_t^{(j)} = \Phi_{\lambda_x/\mu}\left(\boldsymbol{X}_t^{(j)} - \boldsymbol{Z}^{(j)}/\mu\right), \tag{17}$$

$$\boldsymbol{Z}^{(j)} \longleftarrow \boldsymbol{Z}^{(j)} + \mu(\boldsymbol{Y}_t^{(j)} - \boldsymbol{X}_t^{(j)}), \tag{18}$$

$$\boldsymbol{\mathcal{E}}_t = \Psi_{\lambda_e}\left(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}_t^{(1)} \times_2 \boldsymbol{X}_t^{(2)} \ldots \times_d \boldsymbol{X}_t^{(d)}\right). \tag{19}$$

Ψ is the element-wise soft-thresholding operator (Parikh et al., 2014) defined by

$$\Psi_{\lambda_e}(v) = \text{sign}(v)\max(0, |v| - \lambda_e).$$

Algorithm 4 RTPCA-ENR ($p = 1/d$) solved by ADMM

Input: $\boldsymbol{\mathcal{D}}$, k, λ_x, λ_e, μ, $t_{\max}$.

Initialize $\{\boldsymbol{x}_i^{(j)}\}_{i\in[k]}^{j\in[d]}$ with (10); for $j = 1, \ldots, d$, let $\boldsymbol{Y}_0^{(j)} = \boldsymbol{X}_0^{(j)}$ and $\boldsymbol{Z}^{(j)} = \boldsymbol{0}$; let $t = 0$ and $\boldsymbol{\mathcal{E}} = \boldsymbol{0}$.
repeat
  $t \leftarrow t + 1$.
  for $j = 1, 2, \ldots, d$ **do**
    Compute $\boldsymbol{X}_t^{(j)}$ by (16).
    Compute $\boldsymbol{Y}_t^{(j)}$ by (17).
    Update $\boldsymbol{Z}^{(j)}$ by (18).
  end for
  Compute $\boldsymbol{\mathcal{E}}_t$ by (19).
until converged or $t = t_{\max}$

Output: $\hat{\boldsymbol{\mathcal{T}}} = \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}_t^{(1)} \times_2 \boldsymbol{X}_t^{(2)} \ldots \times_d \boldsymbol{X}_t^{(d)}$.

In (9), when $p \notin \{1/d, 2/d\}$, the optimization becomes more difficult because all d groups of the regularizers are nonconvex and nonsmooth. Thanks to Theorem 2, especially its (b), we can obtain arbitrarily sharp Schatten-p quasi-norm regularization by using only one group of nonconvex and nonsmooth regularizers on the component vectors. The corresponding problem is

$$\underset{\{\boldsymbol{x}_i^{(j)}\}_{i\in[k]}^{j\in[d]},\, \boldsymbol{\mathcal{E}}}{\text{minimize}} \ \frac{1}{2}\left\|\boldsymbol{\mathcal{D}} - \sum_{i=1}^k \boldsymbol{x}_i^{(1)} \circ \boldsymbol{x}_i^{(2)} \ldots \circ \boldsymbol{x}_i^{(d)} - \boldsymbol{\mathcal{E}}\right\|_F^2 + \lambda_x \sum_{i=1}^k \left(\frac{1}{q}\|\boldsymbol{x}_i^{(1)}\|^q + \frac{1}{2}\sum_{j=2}^d \|\boldsymbol{x}_i^{(j)}\|^2\right) + \lambda_e \|\boldsymbol{\mathcal{E}}\|_1. \tag{20}$$

We propose to solve (20) by ADMM with iteratively reweighted update (Algorithm 1) embedded, which is shown in Algorithm 5. In the algorithm, for $j = 2, \ldots, d$, the subproblem of $\boldsymbol{X}_t^{(j)}$ has a closed-form solution, which makes the algorithm more efficient than Algorithm 4, especially when d is large.

Algorithm 5 RTPCA-ENR when $p \notin \{1/d, 2/d\}$

Input: $\mathcal{D}$, k, q, λ_x, λ_e, μ, $t_{\max}$.

Initialize $\{\boldsymbol{x}_i^{(j)}\}_{i\in[k]}^{j\in[d]}$ with (10); let $t = 0$ and $\mathcal{E} = \mathbf{0}$; let $\boldsymbol{Y}_0^{(1)} = \boldsymbol{X}_0^{(1)}$ and $\boldsymbol{Z}^{(1)} = \mathbf{0}$.
repeat
  $t \leftarrow t + 1$.
  Compute $\boldsymbol{X}_t^{(1)}$ by (16).
  Compute $\boldsymbol{Y}_t^{(1)}$ by Algorithm 1.
  Update $\boldsymbol{Z}^{(1)}$ by (18).
  for $j = 2, 3, \ldots, d$ **do**
    Compute $\boldsymbol{X}_t^{(j)}$ by $\boldsymbol{X}_t^{(j)} = (\boldsymbol{D}_{(j)} - \boldsymbol{E}_{(j)})[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]\Big([(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}]^T[(\boldsymbol{X}^{(i)})^{\odot_{i\neq j}}] + \lambda_x \boldsymbol{I}\Big)^{-1}$.
  end for
  Compute $\mathcal{E}_t$ by (19).
until converged or $t = t_{\max}$

Output: $\hat{\mathcal{T}} = \mathcal{I} \times_1 \boldsymbol{X}_t^{(1)} \times_2 \boldsymbol{X}_t^{(2)} \ldots \times_d \boldsymbol{X}_t^{(d)}$.

C More experimental results

C.1 Theoretical bound v.s. empirical error

We then apply our method to the synthetic tensors used in Figure 1 and Figure 2, where we let the noise-signal ratio be 0.2. We let $\tilde{\Delta} = \left(\frac{1}{|\bar{\Omega}|}\|\mathcal{P}_{\bar{\Omega}}(\mathcal{D} - \mathcal{X})\|_F^2 - \frac{1}{|\Omega|}\|\mathcal{P}_{\Omega}(\mathcal{D} - \mathcal{X})\|_F^2\right)/\varepsilon^2$, where $\varepsilon = \max\{\|\mathcal{D}\|_\infty, \|\mathcal{X}\|_\infty\}$. Now in Figure 12, we compare $\tilde{\Delta}$ with the theoretical error upper bound Δ defined by (6). $\tilde{\Delta}$ is in log scale for better visualization. We see the empirical reconstruction error is much less than the theoretical error (upper) bound. One reason is that the bound involves ε^2, which is much larger (often 100 times) than the average of square entries of $\mathcal{D}$ and $\mathcal{X}$.

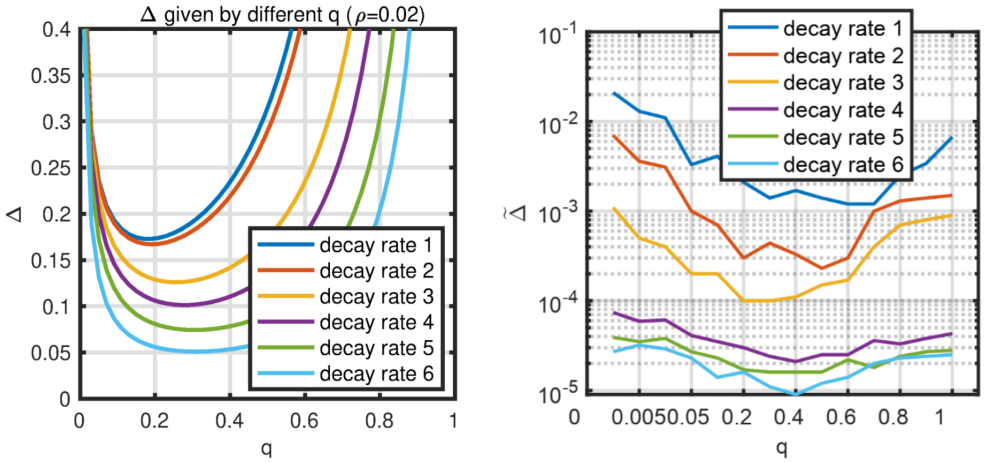

Figure 12: Left: theoretical upper bound Δ (Theorem 7). Right: empirical reconstruction error $\tilde{\Delta}$. The sampling rate ρ is 0.02.

C.2 Iterative performance of LRTC-ENR

Figure 13 shows the value of the objective function and relative recovery error of LRTC-ENR in each iteration on the synthetic data used in Section 5.1. We see that the optimization converged quickly and the relative recovery error decreased when the iteration number increased.

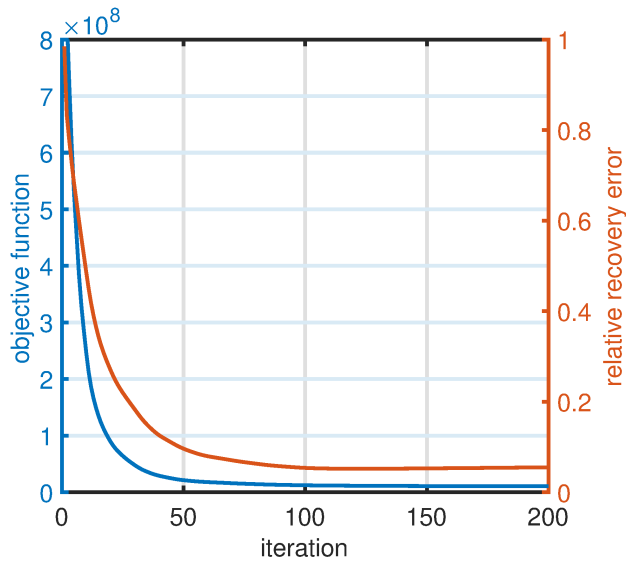

Figure 13: Iterative performance of LRTC-ENR

Relative recovery error on 4-order tensor	missing rate	
	0.7	0.9
HaLRTC	0.98	0.99
KBR-TC	0.34	0.67
BCPF	0.11	0.21
LRTC-ENR	**0.03**	**0.05**

Figure 14: Tensor completion on 4-order tensor of size 30×30×30×30. The data generating model is similar to the one used in Section 5.1.

C.3 LRTC-ENR on higher-order tensors

We compare our methods with HaLRTC, KBR-TC, and BCPF on a synthetic 4-order tensor of size 30×30×30×30 and rank 50. The relative recovery errors (average of ten trials) are reported in Figure 14. Our method outperformed the baselines.

D Proof of the theorems

D.1 Proof of Theorem 1

Proof. Let $\boldsymbol{x}_i^{(j)} = \alpha_{ij}\bar{\boldsymbol{x}}_i^{(j)}$, where $\|\bar{\boldsymbol{x}}_i^{(j)}\| = 1$. Then $\prod_{j=1}^d \|\boldsymbol{x}_i^{(j)}\| = \prod_{j=1}^d \alpha_{ij} = |\lambda_i|$. We have

$$\begin{aligned}\frac{1}{d}\sum_{i=1}^k\sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^q &\geq \sum_{i=1}^k \left(\prod_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^q\right)^{1/d} \\ &= \sum_{i=1}^k\prod_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^{q/d} = \sum_{i=1}^k |\lambda_i|^{q/d},\end{aligned}$$

in which the inequality holds according to the AM-GM inequality. When $\alpha_{i1} = \alpha_{i2} = \ldots \alpha_{id} = |\lambda_i|^{1/d}$, the equality of AM-GM inequality is true. Replacing q with pd, we have

$$\frac{1}{d}\sum_{i=1}^k\sum_{j=1}^d \|\boldsymbol{x}_i^{(j)}\|^{pd} \geq \sum_{i=1}^k |\lambda_i|^p \geq \|\boldsymbol{\mathcal{X}}\|_{S_p}^p.$$

□

D.2 Proof of Theorem 2

Proof. (1) We have

$$
\begin{aligned}
&\sum_{i=1}^{k}\Big(\frac{1}{q}\|\boldsymbol{x}_i^{(1)}\|^q+\sum_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|\Big)=\sum_{i=1}^{k}\Big(\sum_{l=1}^{1/q}\|\boldsymbol{x}_i^{(1)}\|^q+\sum_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|\Big)\\
\geq&\sum_{i=1}^{k}(\frac{1}{q}+d-1)\left(\prod_{l=1}^{1/q}\|\boldsymbol{x}_i^{(1)}\|^q\prod_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|\right)^{\frac{q}{1+qd-q}}\\
=&\sum_{i=1}^{k}(\frac{1}{q}+d-1)\left(\prod_{j=1}^{d}\|\boldsymbol{x}_i^{(j)}\|\right)^{\frac{q}{1+qd-q}}\\
=&\frac{1+qd-q}{q}\sum_{i=1}^{k}|\lambda_i|^{\frac{q}{1+qd-q}}\geq\frac{1+qd-q}{q}\|\boldsymbol{\mathcal{X}}\|_{S_{q/(1+qd-q)}}^{q/(1+qd-q)}.
\end{aligned}
$$

(2) We have

$$
\begin{aligned}
&\sum_{i=1}^{k}\Big(\frac{2}{q}\|\boldsymbol{x}_i^{(1)}\|^q+\sum_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|^2\Big)=\sum_{i=1}^{k}\Big(\sum_{l=1}^{2/q}\|\boldsymbol{x}_i^{(1)}\|^q+\sum_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|^2\Big)\\
\geq&\sum_{i=1}^{k}(\frac{2}{q}+d-1)\left(\prod_{l=1}^{2/q}\|\boldsymbol{x}_i^{(1)}\|^q\prod_{j=2}^{d}\|\boldsymbol{x}_i^{(j)}\|^2\right)^{\frac{q}{2+qd-q}}\\
=&\sum_{i=1}^{k}(\frac{2}{q}+d-1)\left(\prod_{j=1}^{d}\|\boldsymbol{x}_i^{(j)}\|^2\right)^{\frac{q}{2+qd-q}}\\
=&\frac{2+qd-q}{q}\sum_{i=1}^{k}|\lambda_i|^{\frac{2q}{2+qd-q}}\geq\frac{2+qd-q}{q}\|\boldsymbol{\mathcal{X}}\|_{S_{2q/(2+qd-q)}}^{2q/(2+qd-q)}.
\end{aligned}
$$

The equality in the second row of formula (1) (also (2)) holds when all terms in the parentheses are equal, for each i. □

D.3 Proof of the last two in Table 1

Proof. Recall the definition of $\lambda_1, \dots, \lambda_k$. We have

$$\begin{aligned}
&\frac{16^{1/5}}{5}\sum_{i=1}^{k}\left(\|\boldsymbol{x}_i^{(1)}\|^2+\|\boldsymbol{x}_i^{(2)}\|+\|\boldsymbol{x}_i^{(3)}\|\right)\\
=&\frac{16^{1/5}}{5}\sum_{i=1}^{k}\Bigg(\|\boldsymbol{x}_i^{(1)}\|^2+\frac{1}{2}\|\boldsymbol{x}_i^{(2)}\|+\frac{1}{2}\|\boldsymbol{x}_i^{(2)}\|\\
&\qquad+\frac{1}{2}\|\boldsymbol{x}_i^{(3)}\|+\frac{1}{2}\|\boldsymbol{x}_i^{(3)}\|\Bigg)\\
\geq&\frac{16^{1/5}}{5}\sum_{i=1}^{k}5\left(\frac{1}{16}\|\boldsymbol{x}_i^{(1)}\|^2\|\boldsymbol{x}_i^{(2)}\|\|\boldsymbol{x}_i^{(2)}\|\|\boldsymbol{x}_i^{(3)}\|\|\boldsymbol{x}_i^{(3)}\|\right)^{1/5}\\
=&\sum_{i=1}^{k}\left(\|\boldsymbol{x}_i^{(1)}\|^2\|\boldsymbol{x}_i^{(2)}\|^2\|\boldsymbol{x}_i^{(3)}\|^2\right)^{1/5}\\
=&\sum_{i=1}^{k}|\lambda_i|^{2/5}\geq\|\boldsymbol{\mathcal{X}}\|_{S_{2/5}}^{2/5}.
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
&\frac{81^{1/7}}{7}\sum_{i=1}^{k}\left(\|\boldsymbol{x}_i^{(1)}\|^3+\|\boldsymbol{x}_i^{(2)}\|+\|\boldsymbol{x}_i^{(3)}\|\right)\\
=&\frac{81^{1/7}}{7}\sum_{i=1}^{k}\left(\|\boldsymbol{x}_i^{(1)}\|^3+\sum_{j=1}^{3}\left(\frac{1}{3}\|\boldsymbol{x}_i^{(2)}\|+\frac{1}{3}\|\boldsymbol{x}_i^{(3)}\|\right)\right)\\
\geq&\frac{81^{1/7}}{7}\sum_{i=1}^{k}7\left(\frac{1}{3^6}\|\boldsymbol{x}_i^{(1)}\|^3\|\boldsymbol{x}_i^{(2)}\|^3\|\boldsymbol{x}_i^{(3)}\|^3\right)^{1/7}\\
=&\sum_{i=1}^{k}\left(\|\boldsymbol{x}_i^{(1)}\|\|\boldsymbol{x}_i^{(2)}\|\|\boldsymbol{x}_i^{(3)}\|\right)^{2/7}\\
=&\sum_{i=1}^{k}|\lambda_i|^{3/7}\geq\|\boldsymbol{\mathcal{X}}\|_{S_{3/7}}^{3/7}.
\end{aligned}$$

□

D.4 Proof of Theorem 3

First, we give the following lemma, which is proved in Appendix E.1.

Lemma 1. *Let* $\mathcal{S}_{d,n,2}^{k,\perp}=\{\boldsymbol{\mathcal{X}}\in\mathcal{S}_{d,n}^{\perp}:\ \mathrm{rank}(\boldsymbol{\mathcal{X}})\leq k,\ \|\boldsymbol{\mathcal{X}}\|_{S_2^{\perp}}\leq 1\}$. *Then the covering numbers of* $\mathcal{S}_{d,n,2}^{k,\perp}$ *with respect to* $\|\cdot\|_{S_2^{\perp}}$ *satisfy*

$$\log\mathcal{N}(\mathcal{S}_{d,n,2}^{k,\perp},\|\cdot\|_{S_2^{\perp}},\epsilon)\leq\left(\frac{c(d+1)}{\epsilon}\right)^{dk(n-(k+1)/2)+k},$$

where $c>0$ *is a universal constant.*

Proof. We follow the idea of Theorem 4.3 of (Hinrichs et al., 2017) and analyze the entropy number first. According to the monotonicity of dyadic entropy numbers, it is enough to let $n=2^{\alpha}$ and $\eta=2^{\alpha}\cdot 2^{\beta}$, where

$1 \leq \alpha < \beta$ are natural numbers. We have $n/\eta = 2^{-\beta}$. Let $\boldsymbol{\mathcal{X}} \in \mathcal{S}_{d,n,p}^{\perp}$ with $\psi = 1$. The orthogonal CP decomposition of $\boldsymbol{\mathcal{X}}$ is denoted by $\boldsymbol{\mathcal{X}} = \sum_{i=1}^{n} s_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \cdots \circ \boldsymbol{u}_i^{(d)}$ and $s_1 \geq s_2 \geq \cdots \geq s_n$. We define

$$\boldsymbol{\mathcal{X}}_1 := \sum_{i=1}^{n} \bar{s}_{1i} \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \cdots \circ \boldsymbol{u}_i^{(d)}, \quad \bar{\boldsymbol{s}}_1 = (s_1, 0, \ldots, 0)^\top, \tag{21}$$

and, for $j = 2, 3, \ldots, \beta$,

$$\boldsymbol{\mathcal{X}}_j = \sum_{i=1}^{n} \bar{s}_{ji} \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \cdots \circ \boldsymbol{u}_i^{(d)}, \quad \bar{\boldsymbol{s}}_j = (0, \ldots, 0, \overbrace{s_{2^{j-1}}, \ldots, s_{2^j-1}}^{2^{j-1}}, 0, \ldots, 0)^\top. \tag{22}$$

It follows that $\operatorname{rank}(\boldsymbol{\mathcal{X}}_j) \leq 2^{j-1}$, $j \in [\beta]$. Then we have $\|\boldsymbol{\mathcal{X}}_1\|_{S_p^\perp} \leq 1$, and $j = 2, 3, \ldots, \beta$,

$$\begin{aligned} \|\boldsymbol{\mathcal{X}}_j\|_{S_q^\perp} = \left(\sum_{t=2^{j-1}}^{2^j-1} s_t^q \right)^{1/q} &\leq \left(2^{j-1} s_{2^{j-1}}^q\right)^{1/q} = 2^{(j-1)/q} \left(s_{2^{j-1}}^p\right)^{1/p} \\ &\leq 2^{(j-1)/q} \left(\frac{1}{2^{j-1}} \sum_{t=1}^{2^{j-1}} s_t^p \right)^{1/p} \leq 2^{(j-1)(1/q-1/p)}, \end{aligned} \tag{23}$$

where the last inequality holds because of $\sum_{t=1}^{2^{j-1}} s_t^p \leq 1$. Now we can decompose $\boldsymbol{\mathcal{X}}$ as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_1 + \boldsymbol{\mathcal{X}}_2 + \cdots + \boldsymbol{\mathcal{X}}_\beta + \bar{\boldsymbol{\mathcal{X}}}. \tag{24}$$

Suppose $q \geq p$, we have

$$\begin{aligned} \|\bar{\boldsymbol{\mathcal{X}}}\|_{S_q^\perp}^q = \sum_{t=2^\beta}^{\tilde{n}} s_t^q &= \sum_{t=2^\beta}^{\tilde{n}} s_t^p s_t^{q-p} \leq s_{2^\beta}^{q-p} \sum_{t=2^\beta}^{\tilde{n}} s_t^p \\ &\leq \left(\frac{1}{2^\beta} \sum_{t=1}^{2^\beta} s_{2^\beta}^p \right)^{\frac{q-p}{p}} \left(\sum_{t=2^\beta}^{\tilde{n}} s_t^p \right) \\ &\leq 2^{-\beta(q-p)/p} \|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}^{q-p} \|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}^p = 2^{-\beta(q-p)/p} \|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}^q \leq 2^{-\beta(q-p)/p}. \end{aligned} \tag{25}$$

It follows that

$$\left\| \boldsymbol{\mathcal{X}} - \sum_{j=1}^{\beta} \boldsymbol{\mathcal{X}}_j \right\|_{S_q^\perp} = \|\bar{\boldsymbol{\mathcal{X}}}\|_{S_q^\perp} \leq 2^{\beta(1/q-1/p)}.$$

We focus on the case $q = 2$. We see that $2^{(j-1)(1/p-1/2)} \boldsymbol{\mathcal{X}}_j \in \mathcal{S}_{d,n,2}^{2^{j-1},\perp}$. Let $\mathcal{N}_j \subseteq \mathcal{S}_{d,n,2}^{2^{j-1},\perp}$ be an ϵ_j-net, $j \in [\beta]$, and define

$$\mathcal{N} := \left\{ \sum_{j=1}^{\beta} 2^{-(j-1)(1/p-1/2)} \boldsymbol{\mathcal{Z}}_j : \ \boldsymbol{\mathcal{Z}}_j \in \mathcal{N}_j, \ j \in [\beta] \right\}.$$

Then $\mathcal{N}$ is an ϵ-net of $\mathcal{S}_{d,n,p}^{\perp}$ with $\psi = 1$ in $\|\cdot\|_{\mathcal{S}_2^\perp}$, where

$$\epsilon = \sum_{j=1}^{\beta} 2^{-(j-1)(1/p-1/2)} \epsilon_j + 2^{\beta(1/2-1/p)}$$

and $|\mathcal{N}| = \prod_{j=1}^{\beta} |\mathcal{N}_j|$.

Now let $\epsilon_j = c2^{(j-\beta)(1/p-1/2+1)}$, where c is the constant in Lemma 1. We have

$$\begin{aligned}\epsilon =& \sum_{j=1}^{\beta} c2^{(1-\beta)(1/p-1/2)}2^{(j-\beta)} + 2^{\beta(1/2-1/p)}\\ =& c2^{(1-\beta)(1/p-1/2)}\sum_{j=1}^{\beta} 2^{(j-\beta)} + 2^{\beta(1/2-1/p)}\\ \leq& c'2^{-\beta(1/p-1/2)}\\ =& c'\left(\frac{n}{\eta}\right)^{1/p-1/2},\end{aligned}$$

where c' is a constant. Then we have

$$\eta \leq n\left(\frac{c'}{\epsilon}\right)^{2p/(2-p)}. \tag{26}$$

Using Lemma 1 and the definition of $\mathcal{N}_j$, we obtain

$$\begin{aligned}\log|\mathcal{N}| = \log\prod_{j=1}^{\beta}|\mathcal{N}_j| \leq& \log\prod_{j=1}^{\beta}\left(\frac{c(d+1)}{c2^{(j-\beta)(1/p-1/2+1)}}\right)^{d2^{j-1}(n-(2^{j-1}+1)/2)+2^{j-1}}\\ =& \sum_{j=1}^{\beta}\left(d2^{j-1}\left(n-\frac{2^{j-1}+1}{2}\right)+2^{j-1}\right)\log\left((d+1)2^{(\beta-j)(1/p-1/2+1)}\right)\\ \leq& \sum_{j=1}^{\beta} dn2^{j-1}\left((\beta-j)(1/p-1/2+1)+\log(d+1)\right)\\ \leq& (1/p-1/2+1)d\log(d+1)n\sum_{j=1}^{\beta}2^{j-1}(\beta-j)\\ =& (1/p-1/2+1)d\log(d+1)n2^{\beta}\sum_{j=1}^{\beta}2^{j-\beta-1}(\beta-j)\\ \leq& (1/p-1/2+1)d\log(d+1)\eta\\ \leq& (1/p-1/2+1)nd\log(d+1)\left(\frac{c'}{\epsilon}\right)^{2p/(2-p)}.\end{aligned} \tag{27}$$

Now using a general ψ_p instead of 1, we finish the proof. □

D.5 Proof of Theorem 4

Before proving the theorem, we give the following lemma.

Lemma 2. *Let $\mathcal{S}$ be a set of d-order hyper-cubic tensors of side length n. Suppose the ϵ-covering numbers of $\mathcal{S}$ with respect to the Frobenius norm are upper-bounded by B. Suppose $\max\{\|\boldsymbol{\mathcal{D}}\|_\infty, \|\boldsymbol{\mathcal{X}}\|_\infty\} \leq \varepsilon$. Then the following inequality holds with probability at least $1-2n^{-d}$,*

$$\sup_{\boldsymbol{\mathcal{X}}\in\mathcal{S}}\left|\frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}}\|_F - \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}})\|_F\right| \leq \frac{2\epsilon}{\sqrt{|\Omega|}} + 2\varepsilon\left(\frac{d\log n+\log|B|)}{2|\Omega|}\right)^{1/4}.$$

The proof of the lemma can be found in Appendix E.2. Now we prove Theorem 4 as follows.

Proof. According to Theorem 3, the covering numbers of $\mathcal{S}_p^{n,\perp}$ with respect to the Frobenius norm satisfy

$$\log|\mathcal{S}_p^{n,\perp}| \leq (1/2+1/p)nd\left(\log(d+1)\right)\left(\frac{c\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}}{\epsilon}\right)^{2p/(2-p)} \triangleq \log B, \tag{28}$$

where $c > 0$ is a universal constant. Now substituting $\log B$ into Lemma 2 and letting $\epsilon = c\tau\varepsilon\sqrt{dn}$, we get

$$
\begin{aligned}
&\sup_{\boldsymbol{\mathcal{X}}\in\mathcal{S}} \left| \frac{1}{\sqrt{n^d}} \|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F - \frac{1}{\sqrt{|\Omega|}} \|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F \right| \\
\leq & \frac{2\epsilon}{\sqrt{|\Omega|}} + 2\varepsilon \left(\frac{d\log n + (\frac{1}{2} + \frac{1}{p}) nd \left(\log(d+1)\right) \left(\frac{c\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}}{\epsilon} \right)^{2p/(2-p)}}{2|\Omega|} \right)^{1/4} \\
= & \frac{2c\tau\varepsilon\sqrt{dn}}{\sqrt{|\Omega|}} + 2\varepsilon \left(\frac{d\log n + (\frac{1}{2} + \frac{1}{p}) nd \left(\log(d+1)\right) \left(\frac{\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}}{\varepsilon\tau\sqrt{dn}} \right)^{2p/(2-p)}}{2|\Omega|} \right)^{1/4} \\
\leq & c'\varepsilon \left(\frac{(\frac{1}{2} + \frac{1}{p}) nd \left(\log(d+1)\right) \left(\frac{\|\boldsymbol{\mathcal{X}}\|_{S_p^\perp}}{\varepsilon\sqrt{dn}} \right)^{2p/(2-p)}}{|\Omega|} \right)^{1/4},
\end{aligned}
\tag{29}
$$

where we have let $\tau = 1$ and c' is a universal constant. This finished the proof. □

D.6 Proof of Theorem 5

The following lemma (proved in Appendix E.3) will be used in the proof of the theorem.

Lemma 3. *Define $B_{2,q}^{n,k} := \{\boldsymbol{X} \in \mathbb{R}^{n\times k} : \|\boldsymbol{X}\|_{2,q} \leq 1\}$. Then the covering numbers of $B_{2,q}^{n,k}$ with respect to the Frobenius norm satisfy:*
(a) $\log \mathcal{N}(\mathcal{B}_{2,q}^{n,k}, \|\cdot\|_F, \epsilon) \leq c_q(n + \log(ek))\epsilon^{-2q/(2-q)}$, *when* $0 < q < 2$,
(b) $\log \mathcal{N}(\mathcal{B}_{2,q}^{n,k}, \|\cdot\|_F, \epsilon) \leq \lceil nk\epsilon^{-2} \rceil \log(2nk)$, *when* $q = 2$,
where $c_q = O\left(\frac{1}{q}\right)$.

Proof. Let $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d \boldsymbol{X}^{(d)}$ be the CP (or Tucker equivalently) decomposition of $\boldsymbol{\mathcal{X}} \in \mathcal{S}_{k,p}^n$, where $\boldsymbol{\mathcal{I}}$ is super-diagonal. tensor of 1s.

Let $\bar{\boldsymbol{\mathcal{X}}} = \boldsymbol{\mathcal{I}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \cdots \times_d \bar{\boldsymbol{X}}^{(d)}$ and denote $\varsigma = \|\boldsymbol{\mathcal{C}}\|_F$. Let $\|\boldsymbol{X}^{(j)} - \bar{\boldsymbol{X}}^{(j)}\|_F \leq \frac{\epsilon}{d} \prod_{i \neq j} \gamma_j^{-1}$, $j \in [d]$. We have

$$
\begin{aligned}
&\|\boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{X}}}\|_F \\
=&\|\boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_d \boldsymbol{X}^{(d)} - \boldsymbol{\mathcal{I}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \ldots \times_d \bar{\boldsymbol{X}}^{(d)}\|_F \\
=&\|\boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_d \boldsymbol{X}^{(d)} \pm \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_d \bar{\boldsymbol{X}}^{(d)} \\
&\pm \boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_{d-1} \bar{\boldsymbol{X}}^{(d-1)} \times_d \bar{\boldsymbol{X}}^{(d)} \\
&\pm \ldots \pm \boldsymbol{\mathcal{I}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \ldots \times_{d-1} \bar{\boldsymbol{X}}^{(d-1)} \times_d \bar{\boldsymbol{X}}^{(d)} \\
&- \boldsymbol{\mathcal{I}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \ldots \times_d \bar{\boldsymbol{X}}^{(d)}\|_F \\
\leq&\|\boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_d (\boldsymbol{X}^{(d)} - \bar{\boldsymbol{X}}^{(d)})\|_F \\
&+ \|\boldsymbol{\mathcal{I}} \times_1 \boldsymbol{X}^{(1)} \times_2 \ldots \times_{d-1} (\boldsymbol{X}^{(d-1)} - \bar{\boldsymbol{X}}^{(d-1)}) \times_d \bar{\boldsymbol{X}}^{(d)}\|_F \\
&+ \ldots + \|\boldsymbol{\mathcal{I}} \times_1 (\boldsymbol{X}^{(1)} - \bar{\boldsymbol{X}}^{(1)}) \times_2 \bar{\boldsymbol{X}}^{(2)} \ldots \times_d \bar{\boldsymbol{X}}^{(d)}\|_F \\
\leq&\|\boldsymbol{\mathcal{I}}\|_{op} \|\boldsymbol{X}^{(1)}\|_{op} \ldots \|\boldsymbol{X}^{(d-1)}\|_{op} \|\boldsymbol{X}^{(d)} - \bar{\boldsymbol{X}}^{(d)}\|_F \\
&+ \|\boldsymbol{\mathcal{I}}\|_{op} \|\boldsymbol{X}^{(1)}\|_{op} \ldots \|\boldsymbol{X}^{(d-1)} - \bar{\boldsymbol{X}}^{(d-1)}\|_F \|\bar{\boldsymbol{X}}^{(d)}\|_{op} \\
&+ \ldots + \|\boldsymbol{\mathcal{I}}\|_{op} \|\boldsymbol{X}^{(1)} - \bar{\boldsymbol{X}}^{(1)}\|_F \|\bar{\boldsymbol{X}}^{(2)}\|_{op} \ldots \|\bar{\boldsymbol{X}}^{(d-1)}\|_{op} \|\bar{\boldsymbol{X}}^{(d)}\|_{op} \\
\leq&\frac{\epsilon}{d} + \frac{\epsilon}{d} + \cdots + \frac{\epsilon}{d} \\
=&\epsilon.
\end{aligned}
$$

Denote $\phi = \prod_{j=1}^d \gamma_j$. When $0 < q < 2$, the the covering number of $\mathcal{S}_{d,n,q}^k$ can be bounded as

$$
\mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq \prod_{i=1}^d \exp\left(c_q(n + \log(ek)) \left(\frac{d\phi\alpha_q^{(j)}\gamma_j^{-1}}{\epsilon} \right)^{2q/(2-q)} \right),
$$

where c_q is a universal constant. It follows that

$$
\log \mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq \frac{c(n + \log(ek))}{q} \sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}\gamma_j^{-1}}{\epsilon} \right)^{2q/(2-q)},
$$

where c is a constant.

When $q = 2$, the the covering number of $\mathcal{S}_{d,n,q}^k$ can be bounded as

$$
\mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq \prod_{i=1}^d \exp\left(c' nk \log(2nk) \left(\frac{d\phi\alpha_q^{(j)}\gamma_j^{-1}}{\epsilon} \right)^2 \right),
$$

where we have converted the *ceil* operation into multiplying a constant c' for simplicity. It follows that

$$
\log \mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq c' nk \log(2nk) \sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}\gamma_j^{-1}}{\epsilon} \right)^2.
$$

□

D.7 Proof of Theorem 6

Proof. According to Theorem 5, when $) < q < 2$, the covering numbers of $\mathcal{S}_{d,n,q}^k$ with respect to the Frobenius norm satisfy

$$
\log \mathcal{N}(\mathcal{S}_{d,n,q}^k, \|\cdot\|_F, \epsilon) \leq \frac{c(n + \log(ek))}{q} \sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j \epsilon} \right)^{\frac{2q}{2-q}} \triangleq \log B, \tag{30}
$$

where c is a constant. Now substituting $\log B$ into Lemma 2 and letting $\epsilon = \varepsilon\tau\sqrt{dn}$, we get

$$\begin{aligned}
&\sup_{\boldsymbol{\mathcal{X}}\in\mathcal{S}^n_{k,q}} \left| \frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}}\|_F - \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}})\|_F \right| \\
\leq& \frac{2\epsilon}{\sqrt{|\Omega|}} + 2\varepsilon \left(\frac{d\log n + \frac{c(n+\log(ek))}{q}\sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}/\gamma_j}{\epsilon}\right)^{2q/(2-q)}}{2|\Omega|} \right)^{1/4} \\
=& \frac{2\varepsilon\tau\sqrt{dn}}{\sqrt{|\Omega|}} + 2\varepsilon \left(\frac{d\log n + \frac{c(n+\log(ek))}{q}\sum_{j=1}^d \left(\frac{d\phi\alpha_q^{(j)}/\gamma_j}{\varepsilon\tau\sqrt{dn}}\right)^{2q/(2-q)}}{2|\Omega|} \right)^{1/4} \\
\leq& c'\varepsilon \left(\frac{\frac{n+\log(ek)}{q}\sum_{j=1}^d \left(\frac{\sqrt{d}\phi\alpha_q^{(j)}/\gamma_j}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)}}{|\Omega|} \right)^{1/4},
\end{aligned} \tag{31}$$

where we have let $\tau = 1$ and c' is a universal constant. The case of $q = 2$ can be proved similarly.

□

D.8 Proof of Corollary 1

Proof. In Theorem 6, when $0 < q < 2$, letting $\theta = \sum_{j=1}^d \left(\frac{\sqrt{d}\phi\alpha_q^{(j)}/\gamma_j}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)}$, we have

$$\theta \leq \left(\frac{\sqrt{d}\bar{\gamma}^{d-1}}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)} \sum_{j=1}^d (\alpha_q^{(j)})^{2q/(2-q)}. \tag{32}$$

Recall that in the proof for Theorem 1, the equality holds only when $\alpha_{i1} = \alpha_{i2} = \ldots \alpha_{id} = |\lambda_i|^{1/d}$, for all $i \in [k]$. Here we have $\alpha_q^{(j)} = (\sum_{i\in[k]} \alpha_{ij}^p)^{1/p}$. That means we can get $\|\boldsymbol{\mathcal{X}}\|_{S_{p/d}}$ only when $\alpha_q^{(1)} = \alpha_q^{(2)} = \cdots = \alpha_q^{(d)} = \|\boldsymbol{\mathcal{X}}\|_{S_{p/d}}^{1/d}$. Then we have

$$\theta \leq d\left(\frac{\sqrt{d}\bar{\gamma}^{d-1}}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)} \|\boldsymbol{\mathcal{X}}\|_{S_{q/d}}^{2q/(2-q)/d}. \tag{33}$$

Letting $q = pd$ and applying the upper bound of θ to Theorem 6, we obtain

$$\begin{aligned}
&\frac{1}{\sqrt{n^d}}\|\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}}\|_F - \frac{1}{\sqrt{|\Omega|}}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}}-\boldsymbol{\mathcal{X}})\|_F \\
\leq& \frac{2\varepsilon\sqrt{dn}}{\sqrt{|\Omega|}} + c\varepsilon \left(\frac{\frac{n+\log(ek)}{p}\left(\frac{\bar{\gamma}^{d-1}\sqrt{d}}{\varepsilon\sqrt{n}}\|\boldsymbol{\mathcal{X}}\|_{S_p}^{1/d}\right)^{2pd/(2-pd)}}{|\Omega|} \right)^{1/4}.
\end{aligned}$$

In Theorem 6, when $q = 2$, we have the following results (similar to the case $0 < q < 2$)

$$\theta \leq \left(\frac{\sqrt{d}\bar{\gamma}^{d-1}}{\varepsilon\sqrt{n}}\right)^{2q/(2-q)} \sum_{j=1}^d (\alpha_2^{(j)})^2 = d\left(\frac{\sqrt{d}\bar{\gamma}^{d-1}}{\varepsilon\sqrt{n}}\right)^2 \|\boldsymbol{\mathcal{X}}\|_{S_{q/d}}^{2/d}. \tag{34}$$

Letting $q = pd$ and applying the upper bound of θ to Theorem 6, we can get the second bound accordingly. This finished the proof. □

D.9 Proof for Theorem 7

The following lemma provides a sample complexity bound for transductive learning, which is consistent with the objective function and evaluation metric (RMSE) widely used in collaborative filtering.

Lemma 4 (Corollary 1 of (El-Yaniv & Pechyony, 2009), reformulated)**.** *Let $\mathcal{F}$ be a fixed hypothesis set and suppose $\sup_{i,j|\boldsymbol{X}\in\mathcal{H}} |\ell(Y_{ij}, X_{ij})| \leq \tau_\ell$. Suppose a fixed set S of distinct indices is uniformly and randomly split to two subsets S_{train} and S_{test}, where*[4] *$|S_{\text{test}}| > |S_{\text{train}}| > 50$. Then with probability at least $1-\delta$ over the random split, we have*

$$\begin{aligned}\frac{1}{|S_{\text{test}}|}\sum_{(i,j)\in S_{\text{test}}} \ell(Y_{ij}, X_{ij}) \leq & \frac{1}{|S_{\text{train}}|}\sum_{(i,j)\in S_{\text{train}}} \ell(Y_{ij}, X_{ij}) + 4\mathcal{R}_S(\ell\circ\mathcal{F}) \\ & + \frac{11\tau_\ell(|S_{\text{train}}|+|S_{\text{test}}|)}{\sqrt{|S_{\text{train}}|}|S_{\text{test}}|} + 3\tau_\ell\sqrt{\frac{(|S_{\text{train}}|+|S_{\text{test}}|)}{|S_{\text{train}}||S_{\text{test}}|}\log\frac{1}{\delta}}\end{aligned} \tag{35}$$

Before proof, we give the following lemma, which is a variant of the Dudley entropy integral bound on Rademacher complexity.

Lemma 5 (Theorem 3 of (Schreuder, 2020))**.** *Let $\mathcal{F} \subset \{f : \mathcal{X} \mapsto \mathbb{R}\}$ be any class of measurable functions containing the uniformly zero function and let $S_n(\mathcal{F}) = \sup_{f\in\mathcal{F}} \|f\|_{L_2(P_n)}$. Then*

$$\mathcal{R}_n(\mathcal{F}) \leq \inf_{\tau>0}\left(4\tau + \frac{12}{\sqrt{n}}\int_\tau^{S_n(\mathcal{F})}\sqrt{\log\mathcal{N}(\mathcal{F}, L_2(P_n), \zeta)}\mathrm{d}\zeta\right). \tag{36}$$

In the lemma, $\|f\|_{L_2(P_n)}$ is defined as $\|f\|^2_{L_2(P_n)} = \frac{1}{n}\sum_{i=1}^n f(X_i)^2$, which means

$$\mathcal{N}(\mathcal{F}, L_2(P_n), \zeta) = \mathcal{N}(\mathcal{F}, \|\cdot\|_F, \sqrt{n}\zeta). \tag{37}$$

Then we have

$$\begin{aligned}\mathcal{R}_n(\mathcal{F}) &\leq \inf_{\tau>0}\left(4\tau + \frac{12}{\sqrt{n}}\int_\tau^{S_n(\mathcal{F})}\sqrt{\log\mathcal{N}(\mathcal{F}, \|\cdot\|_F, \sqrt{n}\zeta)}\mathrm{d}\zeta\right) \\ &= \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{n}} + \frac{12}{n}\int_\tau^{S_n(\mathcal{F})\sqrt{n}}\sqrt{\log\mathcal{N}(\mathcal{F}, \|\cdot\|_F, \epsilon)}\mathrm{d}\epsilon\right).\end{aligned} \tag{38}$$

Now we use Theorem 5 and (38) to obtain the Rademacher complexity of the tensor decomposition model in LRTC-ENR. When $0 < q < 2$, we have

$$\begin{aligned}\mathcal{R}_{|\Omega|}(\mathcal{F}) &\leq \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\int_\tau^{\varepsilon\sqrt{|\Omega|}}\sqrt{\frac{c(n+\log(ek))}{q}\sum_{j=1}^d\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j\epsilon}\right)^{\frac{2q}{2-q}}}\mathrm{d}\epsilon\right) \\ &= \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\sqrt{\frac{c(n+\log(ek))}{q}\sum_{j=1}^d\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^{\frac{2q}{2-q}}}\int_\tau^{\varepsilon\sqrt{|\Omega|}}\epsilon^{-\frac{q}{2-q}}\mathrm{d}\epsilon\right).\end{aligned} \tag{39}$$

[4] We use these assumptions to simplify the theorem.

When $q = 1$, we obtain

$$\begin{aligned}\mathcal{R}_{|\Omega|}(\mathcal{F}) \leq & \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\sqrt{\frac{c(n+\log(ek))}{q}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\log\frac{\varepsilon\sqrt{|\Omega|}}{\tau}\right)\\ \leq & \frac{4\varepsilon}{|\Omega|} + \frac{12}{|\Omega|}\sqrt{c(n+\log(ek))\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\log|\Omega|\\ \leq & \frac{c'}{|\Omega|}\sqrt{(n+\log(ek))\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\log|\Omega|,\end{aligned} \tag{40}$$

where we have let $\tau = \varepsilon/\sqrt{|\Omega|}$ and c' is a suitable numerical constant.

When $0 < q < 2$ and $q \neq 1$, we have

$$\begin{aligned}\mathcal{R}_{|\Omega|}(\mathcal{F}) \leq & \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\sqrt{\frac{c(n+\log(ek))}{q}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^{\frac{2q}{2-q}}}\left(\frac{2-q}{2-2q}\right)\left((\varepsilon\sqrt{|\Omega|})^{\frac{2-2q}{2-q}} - \tau^{\frac{2-2q}{2-q}}\right)\right)\\ \leq & \frac{4\varepsilon}{|\Omega|} + \frac{12}{|\Omega|}\sqrt{\frac{c(n+\log(ek))}{q}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^{\frac{2q}{2-q}}}\left(\frac{2-q}{2-2q}\right)(\varepsilon\sqrt{|\Omega|})^{\frac{2-2q}{2-q}}\\ \leq & \frac{c'}{|\Omega|}\sqrt{\frac{(n+\log(ek))(2-q)^2(\varepsilon\sqrt{|\Omega|})^{\frac{2-2q}{2-q}}}{q(2-2q)^2}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^{\frac{2q}{2-q}}}\\ = & \frac{c'}{|\Omega|}\sqrt{\frac{(n+\log(ek))(2-q)^2\varepsilon^2|\Omega|^{\frac{1-q}{2-q}}}{q(2-2q)^2}\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\varepsilon\gamma_j}\right)^{\frac{2q}{2-q}}}\end{aligned} \tag{41}$$

where c' is a suitable numerical constant.

When $q = 2$, we have

$$\begin{aligned}\mathcal{R}_{|\Omega|}(\mathcal{F}) \leq & \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\int_{\tau}^{\varepsilon\sqrt{|\Omega|}}\sqrt{cnk\log(2nk)\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}q^{(j)}}{\gamma_j\epsilon}\right)^2}\mathrm{d}\epsilon\right)\\ = & \inf_{\tau>0}\left(\frac{4\tau}{\sqrt{|\Omega|}} + \frac{12}{|\Omega|}\sqrt{cnk\log(2nk)\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\int_{\tau}^{\varepsilon\sqrt{|\Omega|}}\epsilon^{-2}\mathrm{d}\epsilon\right)\\ \leq & \inf_{\tau>0}\left(\frac{4\tau}{|\Omega|} + \frac{12}{|\Omega|}\sqrt{cnk\log(2nk)\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\log\frac{\varepsilon\sqrt{|\Omega|}}{\tau}\right)\\ \leq & \frac{c'}{|\Omega|}\sqrt{nk\log(2nk)\sum_{j=1}^{d}\left(\frac{d\phi\alpha_q^{(j)}}{\gamma_j}\right)^2}\log|\Omega|,\end{aligned} \tag{42}$$

where c' is a suitable numerical constant.

Note that

$$\mathcal{R}_{|\Omega|}(\ell\circ\mathcal{F}) \leq \eta_\ell\mathcal{R}_{|\Omega|}(\mathcal{F}), \tag{43}$$

where η_ℓ is the Lipschitz constant of function ℓ. In this work, ℓ is the square loss, which means $\eta_\ell = 4\varepsilon$. Finally, integrating (40), (41), (42), and (43) with Lemma 4 and renaming the constants, we get the desired results.

E Proof of the lemmas

E.1 Proof of Lemma 1

Before proof, we restate Lemma 4.1 of (Hinrichs et al., 2017) here.

Lemma 6 (Lemma 4.1 of (Hinrichs et al., 2017))**.** *Define* $V_k^n = \{\boldsymbol{U} \in \mathbb{R}^{n\times k} : \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_k,\ k, n \in \mathbb{N},\ k \le n\}$. *Let* $0 < \epsilon < 1$. *Then*

$$\mathcal{N}(V_k^n, \|\cdot\|_{op}, \epsilon) \le \left(\frac{c}{\epsilon}\right)^{k(n-(k+1)/2)},$$

where $c > 0$ *is a universal constant.*

Proof. Let $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d \boldsymbol{X}^{(d)}$ be the orthogonal CP (or Tucker equivalently) decomposition of $\boldsymbol{\mathcal{X}} \in \mathcal{S}_{d,n,2}^{k,\perp}$, where $\boldsymbol{\mathcal{C}}$ is super-diagonal. We have $\|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} = \|\mathrm{diag}(\boldsymbol{\mathcal{C}})\|_2 \le 1$. Then the covering number of $\boldsymbol{\mathcal{C}}$ with respective to $\|\cdot\|_{S_2^\perp}$ is equal to the covering number of $\mathrm{diag}(\boldsymbol{\mathcal{C}})$ with respect to $\|\cdot\|_2$, i.e., $|\mathcal{N}_{\boldsymbol{\mathcal{C}}}| \le \left(3\epsilon^{-1}\right)^k$. According to Lemma 6, the covering number of $\boldsymbol{X}^{(j)}$ with respect to $\|\cdot\|_{op}$ satisfies $|\mathcal{N}_{\boldsymbol{X}^{(j)}}| \le \left(c\epsilon^{-1}\right)^{k(n-(k+1)/2)}$, $j \in [d]$. We have

$$\begin{aligned}
\|\boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{X}}}\|_{S_2^\perp} =& \|\boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d \boldsymbol{X}^{(d)} - \bar{\boldsymbol{\mathcal{C}}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \cdots \times_d \bar{\boldsymbol{X}}^{(d)}\|_{S_2^\perp} \\
=& \|\boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d \boldsymbol{X}^{(d)} \pm \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d \bar{\boldsymbol{X}}^{(d)} \\
&\pm \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_{d-1} \bar{\boldsymbol{X}}^{(d-1)} \times_d \bar{\boldsymbol{X}}^{(d)} \\
&\pm \cdots \pm \boldsymbol{\mathcal{C}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \cdots \times_{d-1} \bar{\boldsymbol{X}}^{(d-1)} \times_d \bar{\boldsymbol{X}}^{(d)} - \bar{\boldsymbol{\mathcal{C}}} \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \cdots \times_d \bar{\boldsymbol{X}}^{(d)}\|_{S_2^\perp} \\
\overset{(a)}{\le}& \|\boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_d (\boldsymbol{X}^{(d)} - \bar{\boldsymbol{X}}^{(d)})\|_{S_2^\perp} \\
&+ \|\boldsymbol{\mathcal{C}} \times_1 \boldsymbol{X}^{(1)} \times_2 \cdots \times_{d-1} (\boldsymbol{X}^{(d-1)} - \bar{\boldsymbol{X}}^{(d-1)}) \times_d \bar{\boldsymbol{X}}^{(d)}\|_{S_2^\perp} \\
&+ \cdots + \|(\boldsymbol{\mathcal{C}} - \bar{\boldsymbol{\mathcal{C}}}) \times_1 \bar{\boldsymbol{X}}^{(1)} \times_2 \bar{\boldsymbol{X}}^{(2)} \cdots \times_d \bar{\boldsymbol{X}}^{(d)}\|_{S_2^\perp} \\
\overset{(b)}{\le}& \|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} \|\boldsymbol{X}^{(1)}\|_{op} \cdots \|\boldsymbol{X}^{(d-1)}\|_{op} \|\boldsymbol{X}^{(d)} - \bar{\boldsymbol{X}}^{(d)}\|_{op} \\
&+ \|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} \|\boldsymbol{X}^{(1)}\|_{op} \cdots \|\boldsymbol{X}^{(d-1)} - \bar{\boldsymbol{X}}^{(d-1)}\|_{op} \|\bar{\boldsymbol{X}}^{(d)}\|_{op} \\
&+ \cdots + \|\boldsymbol{\mathcal{C}} - \bar{\boldsymbol{\mathcal{C}}}\|_{S_2^\perp} \|\bar{\boldsymbol{X}}^{(1)}\|_{op} \|\bar{\boldsymbol{X}}^{(2)}\|_{op} \cdots \|\bar{\boldsymbol{X}}^{(d-1)}\|_{op} \|\bar{\boldsymbol{X}}^{(d)}\|_{op} \\
=& \|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} \|\boldsymbol{X}^{(d)} - \bar{\boldsymbol{X}}^{(d)}\|_{op} + \|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} \|\boldsymbol{X}^{(d-1)} - \bar{\boldsymbol{X}}^{(d-1)}\|_{op} + \cdots \\
&+ \|\boldsymbol{\mathcal{C}}\|_{S_2^\perp} \|\boldsymbol{X}^{(1)} - \bar{\boldsymbol{X}}^{(1)}\|_{op} + \|\boldsymbol{\mathcal{C}} - \bar{\boldsymbol{\mathcal{C}}}\|_{S_2^\perp}.
\end{aligned}$$

Note that (a) holds owing to the triangle inequality of norms and (b) holds because the inequality $\|\boldsymbol{AB}\|_{S_q} \le \|\boldsymbol{A}\|_{op}\|\boldsymbol{B}\|_{S_q}$ (Bhatia, 2013) can be easily extended to orthogonally decomposable tensors. Now let $\|\boldsymbol{\mathcal{C}} - \bar{\boldsymbol{\mathcal{C}}}\|_{S_2^\perp} \le \epsilon/(d+1)$, and $\|\boldsymbol{X}^{(j)} - \bar{\boldsymbol{X}}^{(j)}\|_{op} \le \epsilon/(d+1)$, $j \in [d]$. We arrive at

$$\|\boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{X}}}\|_{S_2^\perp} \le \frac{\epsilon}{d+1} + \frac{\epsilon}{d+1} + \cdots + \frac{\epsilon}{d+1} \le \epsilon.$$

Then the covering number of $\mathcal{S}_{d,n,2}^{k,\perp}$ can be bounded as

$$\begin{aligned}
\mathcal{N}(\mathcal{S}_{d,n,2}^{k,\perp}, \|\cdot\|_{S_2^\perp}, \epsilon) &\le \left(\frac{3(d+1)}{\epsilon}\right)^k \prod_{i=1}^d \left(\frac{c(d+1)}{\epsilon}\right)^{k(n-(k+1)/2)} \\
&\le \left(\frac{c'(d+1)}{\epsilon}\right)^{dk(n-(k+1)/2)+k},
\end{aligned}$$

where c' is a universal constant. This finished the proof. □

E.2 Proof of Lemma 2

Proof. For convenience, we define

$$\hat{h}(\boldsymbol{\mathcal{X}}) = \frac{1}{|\Omega|}\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F^2, \qquad h(\boldsymbol{\mathcal{X}}) = \frac{1}{n^d}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F^2.$$

According to the following lemma

Lemma 7 ((Hoeffding inequality for sampling without replacement)**.** *Let $X_1, X_2, \ldots, Xs$ be a set of samples taken without replacement from a distribution $\{x_1, x_2, \ldots, x_N\}$ of mean u and variance σ^2. Denote $a = \min_i x_i$ and $b = \max_i x_i$. Then*

$$P\left[\left|\frac{1}{s}\sum_{i=1}^{s} X_i - u\right| \geq t\right] \leq 2\exp\left(-\frac{2st^2}{(1-(s-1)/N)(b-a)^2}\right).$$

we have

$$P\left[|\hat{h} - h| \geq t\right] \leq 2\exp\left(-\frac{2|\Omega|t^2}{(1-(|\Omega|-1)/n^d)\varsigma^2}\right),$$

where $\varsigma = 4\varepsilon^2$. Using union bound for all $\bar{\boldsymbol{\mathcal{X}}} \in \mathcal{S}$ yields

$$P\left[\sup_{\bar{\boldsymbol{\mathcal{X}}}\in\mathcal{S}} |\hat{h}(\bar{\boldsymbol{\mathcal{X}}}) - h(\bar{\boldsymbol{\mathcal{X}}})| \geq t\right] \leq 2|\mathcal{S}|\exp\left(-\frac{2|\Omega|t^2}{(1-(|\Omega|-1)/n^d)\varsigma^2}\right).$$

Or equivalently, with probability at least $1 - 2n^{-d}$,

$$\sup_{\bar{\boldsymbol{\mathcal{X}}}\in\mathcal{S}} |\hat{h}(\bar{\boldsymbol{\mathcal{X}}}) - h(\bar{\boldsymbol{\mathcal{X}}})| \leq \sqrt{\frac{\varsigma^2\log\left(|\mathcal{S}|n^d\right)}{2}\left(\frac{1}{|\Omega|} - \frac{1}{n^d} + \frac{1}{n^d|\Omega|}\right)}.$$

Then we have

$$\begin{aligned} g(\Omega) &\triangleq \sup_{\bar{\boldsymbol{\mathcal{X}}}\in\mathcal{S}} |\hat{h}(\bar{\boldsymbol{\mathcal{X}}}) - h(\bar{\boldsymbol{\mathcal{X}}})| \\ &\leq \sqrt{\frac{\varsigma^2}{2}\left(d\log n + \log|\mathcal{S}|\right)\left(\frac{1}{|\Omega|} - \frac{1}{n^d} + \frac{1}{n^d|\Omega|}\right)}. \end{aligned}$$

Since $|\sqrt{u} - \sqrt{v}| \leq \sqrt{|u-v|}$ holds for any non-negative u and v, we have

$$\sup_{\bar{\boldsymbol{\mathcal{X}}}\in\mathcal{S}} \left|\sqrt{\hat{h}(\bar{\boldsymbol{\mathcal{X}}})} - \sqrt{h(\bar{\boldsymbol{\mathcal{X}}})}\right| \leq \sqrt{g(\Omega)}.$$

Recall that $\epsilon \geq \|\boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{X}}}\|_F \geq \|\mathcal{P}(\boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{X}}})\|_F$, we have

$$\left|\sqrt{h(\boldsymbol{\mathcal{X}})} - \sqrt{h(\bar{\boldsymbol{\mathcal{X}}})}\right| = \frac{1}{\sqrt{n^d}}\left|\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}}\|_F - \|\boldsymbol{\mathcal{D}} - \bar{\boldsymbol{\mathcal{X}}}\|_F\right| \leq \frac{\epsilon}{\sqrt{n^d}}$$

and

$$\left|\sqrt{\hat{h}(\boldsymbol{\mathcal{X}})} - \sqrt{\hat{h}(\bar{\boldsymbol{\mathcal{X}}})}\right| = \frac{1}{\sqrt{|\Omega|}}\left|\|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{X}})\|_F - \|\mathcal{P}_\Omega(\boldsymbol{\mathcal{D}} - \bar{\boldsymbol{\mathcal{X}}})\|_F\right| \leq \frac{\epsilon}{\sqrt{|\Omega|}}.$$

It follows that

$$\begin{aligned} &\sup_{\boldsymbol{\mathcal{X}}\in\mathcal{S}} \left|\sqrt{\hat{h}(\boldsymbol{\mathcal{X}})} - \sqrt{h(\boldsymbol{\mathcal{X}})}\right| \\ \leq &\sup_{\boldsymbol{\mathcal{X}}\in\mathcal{S}} \left|\sqrt{\hat{h}(\boldsymbol{\mathcal{X}})} - \sqrt{\hat{h}(\bar{\boldsymbol{\mathcal{X}}})}\right| + \left|\sqrt{\hat{h}(\bar{\boldsymbol{\mathcal{X}}})} - \sqrt{h(\bar{\boldsymbol{\mathcal{X}}})}\right| + \left|\sqrt{h(\bar{\boldsymbol{\mathcal{X}}})} - \sqrt{h(\boldsymbol{\mathcal{X}})}\right| \\ \leq &\frac{\epsilon}{\sqrt{|\Omega|}} + \sqrt{g(\Omega)} + \frac{\epsilon}{\sqrt{n^d}}. \end{aligned}$$

Using the definition of $g(\Omega)$ and letting $\epsilon = 3d$, we have

$$\begin{aligned}
&\sup_{\boldsymbol{\mathcal{X}} \in \mathcal{S}} \left| \sqrt{\hat{h}(\boldsymbol{\mathcal{X}})} - \sqrt{h(\boldsymbol{\mathcal{X}})} \right| \\
\leq & \frac{2\epsilon}{\sqrt{|\Omega|}} + \left(\frac{\varsigma^2}{2} \left(d \log n + \log |\mathcal{S}| \right) \left(\frac{1}{|\Omega|} - \frac{1}{n^d} + \frac{1}{n^d |\Omega|} \right) \right)^{1/4} \\
\leq & \frac{2\epsilon}{\sqrt{|\Omega|}} + 2\varepsilon \left(\frac{d \log n + \log |B|)}{2|\Omega|} \right)^{1/4}.
\end{aligned}$$

□

E.3 Proof of Lemma 3

Proof. Case (a) can be easily obtained by transforming the entropy number result of the special case $p = u = r = 2$ (we have exchanged p and q) of Theorem 13 in (Mayer & Ullrich, 2021) to covering number. Specifically, let

$$e_\eta \leq c_q \left(\frac{\log(ek/\eta) + n}{\eta} \right)^{1/q - 1/2} \leq c_q \left(\frac{\log(ek) + n}{\eta} \right)^{1/q - 1/2},$$

where c_q is a constant depending only on q and $c_q = O(1/q)$. It follows that

$$\eta \leq (n + \log(ek)) \left(\frac{c_q}{e_\eta} \right)^{2q/(2-q)}.$$

Then the covering number is bounded as

$$\begin{aligned}
\log \mathcal{N}(\mathcal{B}_{2,q}^{n,k}, \| \cdot \|_F, \epsilon) \leq & (n + \log(ek)) \left(\frac{c_q}{\epsilon} \right)^{2q/(2-q)} \log 2 \\
= & c_q' (n + \log(ek)) \epsilon^{-2q/(2-q)},
\end{aligned}$$

where $c_q' = O(1/q)$.

Case (b) is a special case of Lemma 3.2 of (Bartlett et al., 2017). Namely, in the lemma, letting $\boldsymbol{X}$ be an identity matrix and $p = q = r = s = 2$, we have $\log \mathcal{N}(\mathcal{B}_{2,q}^{n,k}, \| \cdot \|_F, \epsilon) \leq \lceil nk\epsilon^{-2} \rceil \log(2nk)$.

□