TenIPS: Inverse Propensity Sampling for Tensor Completion

Chengrun Yang
Cornell University

Lijun Ding
Cornell University

Ziyang Wu
Cornell University

Madeleine Udell
Cornell University

Abstract

Tensors are widely used to represent multiway arrays of data. The recovery of missing entries in a tensor has been extensively studied, generally under the assumption that entries are missing completely at random (MCAR). However, in most practical settings, observations are missing not at random (MNAR): the probability that a given entry is observed (also called the propensity) may depend on other entries in the tensor or even on the value of the missing entry. In this paper, we study the problem of completing a partially observed tensor with MNAR observations, without prior information about the propensities. To complete the tensor, we assume that both the original tensor and the tensor of propensities have low multilinear rank. The algorithm first estimates the propensities using a convex relaxation and then predicts missing values using a higher-order SVD approach, reweighting the observed tensor by the inverse propensities. We provide finite-sample error bounds on the resulting complete tensor. Numerical experiments demonstrate the effectiveness of our approach.

1 Introduction

Tensor completion is gaining increasing popularity and is one of the major tensor-related research topics. The literature we survey here is by no means exhaustive. A straightforward method is to flatten a tensor along one of its dimensions to a matrix and then pick one of the extensively studied matrix completion algorithms [1, 2, 3]. However, this method neglects the multiway structure along all other dimensions and does not make full use of the combinatorial relationships. Instead, it is common to assume that the tensor is low rank along each mode. Tensors differ from matrices in having many incompatible notions of rank and low rank decompositions, including CANDECOMP/PARAFAC (CP) [4, 5], Tucker [6] and tensor-train [7]. Each of the decompositions exploits a different definition of tensor rank, and can be used to recover tensors that are low rank in that sense, including CP [8, 9, 10, 11, 12, 13], Tucker [14, 15, 16, 17, 18, 19] and tensor-train [20, 21]. In this paper, we assume the tensor has approximately low multilinear rank, corresponding to a tensor that can be approximated by a low rank Tucker decomposition.

Existing techniques used for tensor completion include subspace projection onto unfoldings [8], alternating minimization [9, 13, 20], gradient descent [21] and expectation-maximization [17]; different surrogates for the rank penalty have been used, including convex surrogates like nuclear norm on unfoldings [22, 14, 23] or specific flattenings [15] and the maximum norm on factors [12], and nonconvex surrogates such as the minimum number of rank-1 sign tensor components [12]. We use a higher-order SVD (HOSVD) approach that does not require rank surrogates. The two methods closest to ours are HOSVD_w [19], which computes a weighted HOSVD by reweighting the tensor twice (before and after the HOSVD) by the inverse square root propensity, and the method in [16], which computes a HOSVD on a second-order estimator of a missing completely at random tensor, and we call it SO-HOSVD. We compare our method with HOSVD_w and SO-HOSVD theoretically in Section 5.2 and numerically in Section 6.1.

Most previous works on tensor completion have used the assumption that entries are missing completely at random (MCAR). The missing not at random (MNAR) setting is less studied, especially for tensors. A missingness pattern is MNAR when the observation probabilities (also called propensities) of different entries are not equal and may depend on the entry values themselves. In the matrix MNAR setting, a popular observation model is 1-bit observations [24]: each entry is observed with a probability that comes from applying a differ-

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

entiable function $\sigma : \mathbb{R} \to [0, 1]$ to the corresponding entry in a parameter matrix, which is assumed to be low rank. Two popular convex surrogates for the rank have been used to estimate the parameter matrix from an entrywise binary observation pattern using a regularized likelihood approach: nuclear norm [24, 25, 23] and max-norm [3, 12]. We show that we can achieve a small propensity estimation error by solving for a parameter tensor with low multilinear rank using a (roughly) square flattening. This approach outperforms simple slicing or flattening methods.

In this paper, we study the problem of provably completing a MNAR tensor with (approximately) low multilinear rank. We use a two-step procedure to first estimate the propensities in this tensor and then predict missing values by HOSVD on the inverse propensity reweighted tensor. We give the error bound on final estimation as Theorem 4.

This paper is organized as follows. Section 2 sets up our notations. Section 3 formally describes the problem we tackle in this paper. Section 4 gives an overview of our algorithms. Section 5 gives further clarification and finite sample error bounds for the algorithms. Section 6 shows numerical experiments.

2 Notations

Basics We define $[N] = \{1, \ldots, N\}$ for a positive integer N. Given a set S, we denote its cardinality by $|S|$. $\subset$ denotes strict subset. We denote $f(n) = O(g(n))$ if there exists $C > 0$ and N such that $|f(n)| \leq Cg(n)$ for all $n \geq N$. The indicator function $\mathbf{1}(X)$ has value 1 if the condition X is true, and 0 otherwise.

Matrices and tensors We denote *vector*, *matrix*, and *tensor* variables respectively by lowercase letters (x), capital letters (X) and Euler script letters ($\mathcal{X}$). For a matrix $X \in \mathbb{R}^{m\times n}$, $\sigma_1(X) \geq \sigma_2(X) \geq \cdots \geq \sigma_{\min\{m,n\}}(X)$ denote its singular values, $\|X\|$ denotes its 2-norm, $\|X\|_\star$ denotes its nuclear norm, $\mathrm{tr}(X)$ denotes its trace, and with another matrix $Y \in \mathbb{R}^{m\times n}$, $\langle X, Y\rangle := \mathrm{tr}(X^\top Y)$ denotes the matrix inner product. For a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$, $\|\mathcal{X}\|_\infty$ denotes its entrywise maximum absolute value. The *order* of a tensor is the number of dimensions; matrices are order-two tensors. Each dimension is called a *mode*. To denote a part of matrix or tensor, we use a colon to denote the mode that is not fixed: given a matrix $A \in \mathbb{R}^{I\times J}$, $A_{i,:}$ and $A_{:,j}$ denote the ith row and jth column of A, respectively. A *fiber* is a one-dimensional section of a tensor $\mathcal{X}$, defined by fixing every index but one; for example, a fiber of the order-3 tensor $\mathcal{X}$ is $X_{:,j,k}$. A *slice* is an $(N-1)$-dimensional section of an order-N tensor $\mathcal{X}$: a slice of the order-3 tensor $\mathcal{X}$ is $X_{:,:,k}$. The *size* of a mode is the number of slices along that mode: the n-th mode of $\mathcal{X}$ has size I_n. A tensor is *cubical* if every mode is the same size: $\mathcal{X} \in \mathbb{R}^{I\times I\times\cdots\times I}$. The *mode-$n$ unfolding* of $\mathcal{X}$, denoted as $\mathcal{X}^{(n)}$, is a matrix whose columns are the mode-n fibers of $\mathcal{X}$. For example, given an order-3 tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$, $\mathcal{X}^{(1)} \in \mathbb{R}^{I\times(J\times K)}$.

Products We denote the *n-mode product* of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ with a matrix $U \in \mathbb{R}^{J\times I_n}$ by $\mathcal{X} \times_n U \in \mathbb{R}^{I_1\times\cdots\times I_{n-1}\times J\times I_{n+1}\times\cdots\times I_N}$; the $(i_1, i_2, \ldots, i_{n-1}, j, i_{n+1}, \ldots, i_N)$-th entry is $\Sigma_{i_n=1}^{I_n} x_{i_1 i_2 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} u_{j i_n}$. $\otimes$ denotes the Kronecker product. Given two tensors with the same shape, we use $\odot$ to denote their entrywise product.

Missingness Given a partially observed order-N tensor $\mathcal{X} \in \mathbb{R}^{I_1\times\cdots\times I_N}$, we denote its observation pattern by $\Omega \in \{0,1\}^{I_1\times\cdots\times I_N}$: the *mask tensor* of $\mathcal{X}$. It is a binary tensor that denotes whether each entry of $\mathcal{X}$ is observed or not. Ω has the same shape as $\mathcal{X}$, with entry value 1 if the corresponding entry of $\mathcal{X}$ is observed, and 0 otherwise. With an abuse of notation, we call $\Omega := \{(i_1, i_2, \ldots, i_N) | \Omega_{i_1,i_2,\ldots,i_N} = 1\}$ the *mask set* of $\mathcal{X}$. Given a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$, we use $\mathcal{E}(i_1, i_2, \ldots, i_N)$ to denote a binary tensor with the same shape as $\mathcal{X}$, with value 1 at the $(i_1, i_2, \ldots, i_N)$-th entry and 0 elsewhere.

Square unfoldings Extending the notation of [15], with a matrix $A \in \mathbb{R}^{I\times J}$ and integers I', J' satisfying $IJ = I'J'$, $\texttt{reshape}(A, I', J')$ gives an $I' \times J'$ matrix A' with entries taken columnwise from A. Given a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$, we can partition the indices of its N modes into two sets, S and $S^C = [N] \setminus S$, and permute the order of the N modes by permutation $\pi_S = (S_1, \ldots, S_{|S|}, S_1^C, \ldots, S_{N-|S|}^C)$, so that the modes in set S appear first, followed by modes in S^C:

$$\mathcal{X}_{i_1\cdots i_N} = \pi_S(\mathcal{X})_{i_{S_1}\cdots i_{S_{|S|}} i_{S_1^C}\cdots i_{S_{N-|S|}^C}}.$$

Denote the *S-unfolding* of $\mathcal{X}$ as

$$\mathcal{X}_S := \texttt{reshape}(\pi_S(\mathcal{X})^{(1)}, \prod_{n\in S} I_n, \prod_{n\in S^C} I_n).$$

Our methods for tensor completion rely on methods for matrix completion that work best for square matrices. To make $\mathcal{X}_S$ as square as possible, $\Big|\prod_{n\in S} I_n - \prod_{n\in S^C} I_n\Big|$ should be as small as possible. Hence we define the *square set* of $\mathcal{X}$ as

$$S_\square = \operatorname*{arg\,min}_{S\subset[N]} \Big|\prod_{n\in S} I_n - \prod_{n\in S^C} I_n\Big|,$$

the *square unfolding* of $\mathcal{X}$ as

$$\mathcal{X}_\square := \texttt{reshape}(\pi_{S_\square}(\mathcal{X})^{(1)}, \prod_{n\in S_\square} I_n, \prod_{n\in S_\square^C} I_n),$$

and the *square norm* $\|\mathcal{X}\|_\square := \|\mathcal{X}_\square\|_\star$ of $\mathcal{X}$.

Dimensions of unfoldings For brevity, we denote $I_S := \prod_{n\in S} I_n$, $I_{S^C} := \prod_{n\in S^C} I_n$, $I_\square := \prod_{n\in S_\square} I_n$, $I_{\square^C} := \prod_{n\in S_\square^C} I_n$, $I_{(-n)} := \prod_{m\in[N], m\neq n} I_m$. Thus $I_{[N]} = \prod_{n\in[N]} I_n = I_S \cdot I_{S^C} = I_\square \cdot I_{\square^C} = I_n \cdot I_{(-n)}$.

3 Problem setting

In this paper, we study the following problem: given a partially observed tensor $\mathcal{B}_{\text{obs}} \in \mathbb{R}^{I_1\times\cdots\times I_N}$ with MNAR entries, how can we recover its missing values?

Throughout the paper, we denote the true order-N tensor we want to complete as $\mathcal{B} \in \mathbb{R}^{I_1\times\cdots\times I_N}$. For each $n \in [N]$, we suppose $I_n \leq I_{(-n)} := \prod_{m\in[N], m\neq n} I_m$ for cleanliness. We assume there exists a propensity tensor $\mathcal{P} \in \mathbb{R}^{I_1\times I_2\times\cdots I_N}$, such that $\mathcal{B}_{i_1 i_2\cdots i_N}$ is observed with probability $\mathcal{P}_{i_1 i_2\cdots i_N}$. We observe the entries without noise: with the observation pattern Ω, $\mathcal{B}_{\text{obs}} = \mathcal{B} \odot \Omega$.

A tensor $\mathcal{B}$ has *multilinear rank* $(r_1^{\text{true}}, r_2^{\text{true}}, \ldots, r_N^{\text{true}})$ if r_n^{true} is the rank of $\mathcal{B}^{(n)}$. For any $n \in [N]$, $r_n^{\text{true}} \leq I_n$. We can write the *Tucker decomposition* of the tensor $\mathcal{B}$ as $\mathcal{B} = \mathcal{G}^{\text{true}} \times_1 U_1^{\text{true}} \times \cdots \times_N U_N^{\text{true}}$, with *core tensor* $\mathcal{G}^{\text{true}} \in \mathbb{R}^{r_1^{\text{true}}\times\cdots\times r_N^{\text{true}}}$ and column orthonormal *factor matrices* $U_n^{\text{true}} \in \mathbb{R}^{I_n\times r_n^{\text{true}}}$ for $n \in [N]$.

We seek a *fixed-rank approximation* of $\mathcal{B}$ by a tensor with multilinear rank $(r_1, r_2, \cdots, r_N)$: we want to find a core tensor $\mathcal{W} \in \mathbb{R}^{r_1\times r_2\times\cdots\times r_N}$ and N factor matrices $Q_n \in \mathbb{R}^{I_n\times r_n}$, $n \in [N]$ with orthonormal columns, such that $\mathcal{B} \approx \mathcal{W} \times_1 Q_1 \times_2 \cdots \times_N Q_N$. We generally seek a low multilinear rank decomposition with $r_n < I_n$.

4 Methodology

Our algorithm proceeds in two steps. First, we estimate the propensities by ConvexPE (Algorithm 1) or NonconvexPE (Algorithm 2), with an overview in Table 1. Both of these algorithms use a Bernoulli maximum likelihood estimator for 1-bit matrix completion [24] to estimate the propensities from the mask tensor Ω, aiming to recover propensities that come from the low rank parameters. ConvexPE explicitly requires the propensities to be neither too large or too small. NonconvexPE does not require the associated tuning parameters, but empirically returns a good solution if the true propensity tensor $\mathcal{P}$ has this property. With the estimated propensity tensor $\widehat{\mathcal{P}}$, we estimate the data tensor $\mathcal{B}$ by TenIPS (Algorithm 3), a procedure that only requires a Tucker decomposition on the propensity-reweighted observations.

Our propensity estimation uses the observation model of 1-bit matrix completion. Each entry of $\mathcal{P}$ comes from applying a differentiable link function $\sigma : \mathbb{R} \to [0, 1]$ to the corresponding entry of a parameter tensor $\mathcal{A}$, which we are trying to solve. An instance is the logistic function $\sigma(x) = 1/(1 + e^{-x})$. We assume $\mathcal{A}$ has low multilinear rank. In ConvexPE (Algorithm 1), $\mathcal{A}_\square$ is low-rank from Lemma 1. We also assume an upper bound on the nuclear norm of $\mathcal{A}_\square$, a convex surrogate for its low-rank property. ConvexPE can be implemented by the proximal-proximal-gradient method (PPG) [26] or the proximal alternating gradient descent. In Section 5, we will show that on a square tensor, the square unfolding achieves the smallest upper bound for propensity estimation among all possible unfoldings.

In practice, the square unfolding of a tensor is often a large matrix: $I^{N/2}$-by-$I^{N/2}$ for a cubical tensor with order N. Since each iteration of the PPG subroutine in ConvexPE requires the computation of a truncated SVD, this algorithm becomes too expensive in such case. Also, it does not make full use of the low multilinear rank property of $\mathcal{A}$. As a substitute, we propose NonconvexPE (Algorithm 2), which uses gradient descent (GD) on the core tensor $\mathcal{G}^{\mathcal{A}}$ and factor matrices $\{U_n^{\mathcal{A}}\}_{n\in[N]}$ to minimize the objective function $f(\mathcal{G}^{\mathcal{A}}, \{U_n^{\mathcal{A}}\}_{n\in[N]})$ defined in Line 4. It achieves a feasible solution with similar quality as ConvexPE, and does not require the tuning of thresholds τ and γ. This can be attributed to the fact that the objective function f is multi-convex with respect $(\mathcal{G}^{\mathcal{A}}, U_1^{\mathcal{A}}, \ldots, U_N^{\mathcal{A}})$. The gradient computation can be found in Appendix C.

TenIPS (Algorithm 3) completes the observed data tensor $\mathcal{B}_{\text{obs}}$ by HOSVD on its entrywise inverse propensity reweighting $\bar{\mathcal{X}}(\mathcal{P})$, as defined in Line 2. For each $(i_1, i_2, \ldots, i_N) \in \Omega$, the corresponding term in $\bar{\mathcal{X}}(\mathcal{P})$ is an unbiased estimate for $\mathcal{B} \odot \mathcal{E}(i_1, i_2, \ldots, i_N)$:

$$\begin{aligned}
&\mathbb{E}[\frac{1}{\mathcal{P}_{i_1 i_2\cdots i_N}} \mathcal{B}_{\text{obs}} \odot \mathcal{E}(i_1, i_2, \ldots, i_N)] \\
&= \mathcal{P}_{i_1 i_2\cdots i_N} \cdot \frac{1}{\mathcal{P}_{i_1 i_2\cdots i_N}} \mathcal{B}_{\text{obs}} \odot \mathcal{E}(i_1, i_2, \ldots, i_N) \\
&= \mathcal{B} \odot \mathcal{E}(i_1, i_2, \ldots, i_N),
\end{aligned}$$

in which the second equality comes from noiseless observations. Thus $\bar{\mathcal{X}}(\mathcal{P})$ is an unbiased estimator for $\mathcal{B}$:

$$\begin{aligned}
\mathbb{E}\,\bar{\mathcal{X}}(\mathcal{P}) &= \mathbb{E}\Big[\sum_{(i_1, i_2, \ldots, i_N)\in\Omega} \frac{1}{\mathcal{P}_{i_1\cdots i_N}} \mathcal{B}_{\text{obs}} \odot \mathcal{E}(i_1, \ldots, i_N)\Big] \\
&= \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \mathbb{E}\Big[\frac{1}{\mathcal{P}_{i_1 i_2\cdots i_N}} \mathcal{B}_{\text{obs}} \odot \mathcal{E}(i_1, \ldots, i_N)\Big] \\
&= \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \mathcal{B} \odot \mathcal{E}(i_1, \ldots, i_N) = \mathcal{B}.
\end{aligned}$$

The input propensity tensor can be either true ($\mathcal{P}$) or estimated ($\widehat{\mathcal{P}}$). With the estimated propensity $\widehat{\mathcal{P}}$, we get

Algorithm 1 ConvexPE: Convex Propensity Estimation

```
Input: mask tensor $\Omega \in \mathbb{R}^{I_1\times\cdots\times I_N}$, link function $\sigma$, thresholds $\tau$, $\gamma$
Output: estimated propensity tensor $\widehat{\mathcal{P}}$
Compute $S_\square$, the square set of $\Omega$.
Compute best completion $\widehat{\mathcal{A}}_\square = \operatorname*{argmax}_{\Gamma\in\mathcal{S}_{\tau,\gamma}} \sum_{i=1}^{I_\square}\sum_{j=1}^{I_{\square^C}} [(\Omega_\square)_{ij}\log\sigma(\Gamma_{ij}) + (1-(\Omega_\square)_{ij})\log(1-\sigma(\Gamma_{ij}))]$,
    where $\mathcal{S}_{\tau,\gamma} = \{\Gamma\in\mathbb{R}^{I_\square\times I_{\square^C}} : \|\Gamma\|_\star \le \tau\sqrt{I_{[N]}}, \|\Gamma\|_\infty \le \gamma\}$.
Estimate propensities $\widehat{\mathcal{P}} = \sigma(\widehat{\mathcal{A}})$.
return $\widehat{\mathcal{P}}$
```

Algorithm 2 NonconvexPE: Nonconvex Propensity Estimation

```
Input: mask tensor $\Omega \in \mathbb{R}^{I_1\times\cdots\times I_N}$, link function $\sigma$, step size $t$, initialization $\{\bar{\mathcal{G}}^{\mathcal{A}}, \bar{U}_1^{\mathcal{A}}, \dots, \bar{U}_N^{\mathcal{A}}\}$ (or target rank
    $(r_1, \dots, r_N)$ with a certain initialization rule)
Output: estimated propensity tensor $\widehat{\mathcal{P}}$
Initialize core tensor and factor matrices $\mathcal{G}^{\mathcal{A}}, U_1^{\mathcal{A}}, \dots, U_N^{\mathcal{A}} \leftarrow \bar{\mathcal{G}}^{\mathcal{A}}, \bar{U}_1^{\mathcal{A}}, \dots, \bar{U}_N^{\mathcal{A}}$.
Define objective
    $f(\mathcal{G}^{\mathcal{A}}, \{U_n^{\mathcal{A}}\}_{n\in[N]}) := \sum_{i_1\cdots i_N} -\Omega_{i_1\cdots i_N}\log\sigma(\widehat{\mathcal{A}}_{i_1\cdots i_N}) - (1-\Omega_{i_1\cdots i_N})\log(1-\sigma(\widehat{\mathcal{A}}_{i_1\cdots i_N}))$,
    in which $\widehat{\mathcal{A}} = \mathcal{G}^{\mathcal{A}} \times_1 U_1^{\mathcal{A}} \times_2 \cdots \times_N U_N^{\mathcal{A}}$.
do
    Compute gradients with respect to core tensor and factor matrices $\left(\frac{\partial f}{\partial \mathcal{G}^{\mathcal{A}}}, \frac{\partial f}{\partial U_1^{\mathcal{A}}}, \dots, \frac{\partial f}{\partial U_N^{\mathcal{A}}}\right)$.
    Perform gradient descent update: $\mathcal{G}^{\mathcal{A}}, U_1^{\mathcal{A}}, \dots, U_N^{\mathcal{A}} \leftarrow \left(\mathcal{G}^{\mathcal{A}} - t\frac{\partial f}{\partial \mathcal{G}^{\mathcal{A}}}, U_1^{\mathcal{A}} - t\frac{\partial f}{\partial U_1^{\mathcal{A}}}, \dots, U_N^{\mathcal{A}} - t\frac{\partial f}{\partial U_N^{\mathcal{A}}}\right)$.
while not converged
Estimate propensities $\widehat{\mathcal{P}} = \sigma(\widehat{\mathcal{A}})$.
return $\widehat{\mathcal{P}}$
```

$\bar{\mathcal{X}}(\widehat{\mathcal{P}})$ instead of $\bar{\mathcal{X}}(\mathcal{P})$, $\widehat{\mathcal{X}}(\widehat{\mathcal{P}})$ instead of $\widehat{\mathcal{X}}(\mathcal{P})$; for brevity, we denote $\bar{\mathcal{X}}(\mathcal{P})$ and $\widehat{\mathcal{X}}(\mathcal{P})$ by $\bar{\mathcal{X}}$ and $\widehat{\mathcal{X}}$, respectively. We show the estimation error for $\mathcal{B}$ in Theorem 4.

5 Error analysis

To bound the relative estimation error $\|\widehat{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}\|_{\mathrm{F}}/\|\mathcal{B}\|_{\mathrm{F}}$, we first bound the error in the propensity estimates in ConvexPE, and then consider how this error propagates into the error of our final tensor estimate in TenIPS. Theorem 3 shows the optimality of the square unfolding for propensity estimation; Theorem 4 presents a special case of our bound on the tensor completion error with estimated propensities, with the full version as Appendix A, Theorem 5. We defer their proofs to Appendix B.

5.1 Error in propensity estimates

We first show a corollary of [15, Lemma 6 (2) and Lemma 7] that bounds the rank of an unfolding.

Lemma 1. *Suppose $\mathcal{X}$ has Tucker decomposition $\mathcal{X} = \mathcal{C} \times_1 U_1 \times_2 U_2 \times_3 \cdots \times_N U_N$, where $\mathcal{C} \in \mathbb{R}^{r_1^{\mathrm{true}}\times r_2^{\mathrm{true}}\times\cdots\times r_N^{\mathrm{true}}}$ and $U_n \in \mathbb{R}^{I_n\times r_n^{\mathrm{true}}}$ for $n \in [N]$. Given $S \subset [N]$, $\mathcal{X}_S = \underset{j\in S}{\otimes} U_j \cdot \mathcal{C}_S \cdot \left(\underset{j\in[N]\setminus S}{\otimes} U_j\right)^\top$, and thus* $\operatorname{rank}(\mathcal{X}_S) \le \min\left\{\prod_{n\in S} r_n^{\mathrm{true}}, \prod_{n\in[N]\setminus S} r_n^{\mathrm{true}}\right\}$.

Proof. [15, Lemma 6 (2)] states that

$$\mathcal{X}_{[n]} = (U_n \otimes U_{n-1} \otimes \cdots \otimes U_1)\, \mathcal{C}_{[n]}\, (U_N \otimes U_{N-1} \otimes \cdots \otimes U_{n+1})^\top \tag{1}$$

for $n \in [N]$. Thus Lemma 1 holds for $\mathcal{X}$ by applying Equation 1 to an entry-reordered tensor $\widetilde{\mathcal{X}} \in \mathbb{R}^{I_{n_1}\times I_{n_2}\times\cdots\times I_{n_N}}$, such that $S = \{n_j\}_{j\in[S]}$ and $\widetilde{\mathcal{X}}_{i_{n_1} i_{n_2}\cdots i_{n_N}} = \mathcal{X}_{i_1 i_2\cdots i_N}$. The upper bound for $\operatorname{rank}(\mathcal{X}_S)$ follows. □

As a corollary of [24, Lemma 1] and [25, Theorem 2], we have Lemma 2 for the Frobenius norm error of the propensity tensor estimate.

Lemma 2. *Assume that $\mathcal{P} = \sigma(\mathcal{A})$. Given a set $S \subset [N]$, together with the following assumptions:*

A1. *$\mathcal{A}_S$ has bounded nuclear norm: there exists a constant $\theta > 0$ such that $\|\mathcal{A}_S\|_\star \le \theta\sqrt{I_{[N]}}$.*

A2. *Entries of $\mathcal{A}$ have bounded absolute value: there exists a constant $\alpha > 0$ such that $\|\mathcal{A}\|_\infty \le \alpha$.*

Algorithm 3 TenIPS: Tensor completion by Inverse Propensity Sampling

```
Input: mask set $\Omega$, partially observed tensor $\mathcal{B}_{\text{obs}} \in \mathbb{R}^{I_1\times\cdots\times I_N}$, propensity tensor $\mathcal{P} \in \mathbb{R}^{I_1\times\cdots\times I_N}$, target rank $(r_1, r_2, \cdots, r_N)$
Output: estimated tensor $\widehat{\mathcal{X}}(\mathcal{P})$
$\bar{\mathcal{X}}(\mathcal{P}) \leftarrow \sum_{(i_1,i_2,\ldots,i_N)\in\Omega} \frac{1}{\mathcal{P}_{i_1 i_2\cdots i_N}} \mathcal{B}_{\text{obs}} \odot \mathcal{E}(i_1, i_2, \ldots, i_N)$
for $n = 1, 2, \ldots, N$ do                                              ▷ Recover factors
    $Q_n(\mathcal{P}) \leftarrow$ left $r_n$ singular vectors of $\bar{\mathcal{X}}(\mathcal{P})^{(n)}$
$\mathcal{W} \leftarrow \bar{\mathcal{X}}(\mathcal{P}) \times_1 Q_1(\mathcal{P})^\top \times_2 \cdots \times_N Q_N(\mathcal{P})^\top$     ▷ Recover core
$\widehat{\mathcal{X}}(\mathcal{P}) \leftarrow \mathcal{W}(\mathcal{P}) \times_1 Q_1(\mathcal{P}) \times_2 \cdots \times_N Q_N(\mathcal{P})$
return $\widehat{\mathcal{X}}(\mathcal{P})$
```

Table 1: Propensity estimation algorithms.

	base algorithm	hyperparameters
ConvexPE (Algorithm 1)	proximal-proximal-gradient	τ and γ
NonconvexPE (Algorithm 2)	gradient descent	step size t and target rank

Suppose we run ConvexPE *(Algorithm 1) with thresholds satisfying $\tau \geq \theta$ and $\gamma \geq \alpha$ to obtain an estimate $\widehat{\mathcal{P}}$ of $\mathcal{P}$. With $L_\gamma := \sup_{x\in[-\gamma,\gamma]} \frac{|\sigma'(x)|}{\sigma(x)(1-\sigma(x))}$, there exists a universal constant $C > 0$ such that if $I_S + I_{S^C} \geq C$, with probability at least $1 - \frac{C}{I_S + I_{S^C}}$, the estimation error*

$$\frac{1}{I_{[N]}} \|\widehat{\mathcal{P}} - \mathcal{P}\|_{\text{F}}^2 \leq 4eL_\gamma \tau \Big(\frac{1}{\sqrt{I_S}} + \frac{1}{\sqrt{I_{S^C}}}\Big). \qquad (2)$$

In the simplest case, when N is an even integer and $I_1 = \ldots = I_N = I$, the right-hand side (RHS) of the estimation error in Equation 2 is in $O(I^{N/4})$.

Theorem 3 then shows that the square unfolding achieves the smallest upper bound for propensity estimation among all possible unfoldings sets S.

Theorem 3. *Instate the same conditions as Lemma 2, and further assume that there exists a constant $c > 0$ such that $r_n^{\text{true}} \leq cI_n$ for every $n \in [N]$. Then $S = S_\square$ gives the smallest upper bound (RHS of the result in Lemma 2) on the propensity estimation error $\|\widehat{\mathcal{P}} - \mathcal{P}\|_{\text{F}}^2$ among all unfolding sets $S \subset [N]$.*

Proof. Denote $r_S^{\text{true}} := \prod_{n\in S} r_n^{\text{true}}$ and $r_{S^C}^{\text{true}} := \prod_{n\in[N]\setminus S} r_n^{\text{true}}$. We know from Lemma 1 that for every unfolding of $\mathcal{A}$, $\text{rank}(\mathcal{A}_S) \leq \min\Big\{r_S^{\text{true}}, r_{S^C}^{\text{true}}\Big\}$. Since $\|\mathcal{A}_S\|_\star \leq \sqrt{\text{rank}(\mathcal{A}_S)} \cdot \|\mathcal{A}\|_{\text{F}} \leq \alpha\sqrt{\text{rank}(\mathcal{A}_S)} \cdot I^{\frac{N}{2}}$, we need $\tau \geq \theta \geq \alpha\sqrt{\text{rank}(\mathcal{A}_S)}$ for the conditions of Lemma 2 to hold. For simplicity, suppose $\tau = \alpha\sqrt{\text{rank}(\mathcal{A}_S)}$, the smallest possible value for the exact recovery of $\mathcal{A}$. Without loss of generality, suppose $|S| \leq \frac{N}{2}$. We have

$$\begin{aligned}
\frac{1}{I_{[N]}} \|\widehat{\mathcal{P}} - \mathcal{P}\|_{\text{F}}^2 &\leq 4eL_\gamma \tau \Big(\frac{1}{\sqrt{I_S}} + \frac{1}{\sqrt{I_{S^C}}}\Big) \\
&= 4eL_\gamma \alpha \cdot \sqrt{\text{rank}(\mathcal{A}_S)} \Big(\frac{1}{\sqrt{I_S}} + \frac{1}{\sqrt{I_{S^C}}}\Big) \\
&\leq 4eL_\gamma \alpha \Big(\sqrt{\frac{r_S^{\text{true}}}{I_S}} + \sqrt{\frac{r_{S^C}^{\text{true}}}{I_{S^C}}}\Big) \\
&\leq 4eL_\gamma \alpha \Big(\sqrt{c^{|S|}} + \sqrt{c^{N-|S|}}\Big).
\end{aligned}$$

The final expression is the smallest when $S = S_\square$. □

5.2 Error in tensor completion: special case

We present a special case of our bound on the recovery error for a cubical tensor with equal multilinear rank as Theorem 4. This bound is dominated by the error from the matrix Bernstein inequality [27] on each of the N unfoldings, and asymptotically goes to 0 when the tensor size $I \to \infty$. Note that our full theorem applies to any tensor; we defer the formal statement to Appendix A, Theorem 5.

Theorem 4. *Consider an order-N cubical tensor $\mathcal{B}$ with size $I_1 = \cdots = I_N = I$ and multilinear rank $r_1^{\text{true}} = \cdots = r_N^{\text{true}} = r < I$, and two order-$N$ cubical tensors $\mathcal{P}$ and $\mathcal{A}$ with the same shape as $\mathcal{B}$. Each entry of $\mathcal{B}$ is observed with probability from the corresponding entry of $\mathcal{P}$. Assume $I \geq rN \log I$, and there exist constants $\psi, \alpha \in (0, \infty)$ such that $\|\mathcal{A}\|_\infty \leq \alpha$, $\|\mathcal{B}\|_\infty = \psi$. Further assume that for each $n \in [N]$, the condition number $\frac{\sigma_1(\mathcal{B}^{(n)})}{\sigma_r(\mathcal{B}^{(n)})} \leq \kappa$ is a constant independent of tensor sizes and dimensions. Then under the conditions of Lemma 2, with probability at least $1 - I^{-1}$, the fixed multilinear rank $(r, r, \ldots, r)$ approximation $\widehat{\mathcal{X}}(\widehat{\mathcal{P}})$ com-*

puted from ConvexPE *and* TenIPS *(Algorithms 1 and 3) with thresholds $\tau \ge \theta$ and $\gamma \ge \alpha$ satisfies*

$$\frac{\|\widehat{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}\|_{\mathrm{F}}}{\|\mathcal{B}\|_{\mathrm{F}}} \le CN\sqrt{\frac{r \log I}{I}}, \tag{3}$$

in which C depends on κ.

Note how this bound compares with the bounds for similar algorithms. HOSVD_w has a relative error of $O(rN^2 I^{-1/2} \log I)$ [19, Theorem 3.3] for noiseless recovery with known propensities, and SO-HOSVD achieves a better bound of $O(\sqrt{\frac{r \log I}{I^{N-1}}})$ [16, Theorem 3] but assumes that the tensor is MCAR. In contrast, our bound holds for the tensor MNAR setting and does not require known propensities. It is the first bound in this setting, to our knowledge.

We show a sketch of the proof for Theorem 4 to illustrate the main idea. This is the special case of the full proof in Appendix B.

Proof. (sketch)

The propensity estimation error in Lemma 2, Equation 2 propagates to the error between $\bar{\mathcal{X}}(\widehat{\mathcal{P}})$ and $\bar{\mathcal{X}}(\mathcal{P})$:

$$\begin{aligned}
&\|\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \bar{\mathcal{X}}\|_{\mathrm{F}}^2 \\
&= \sum_{(i_1, i_2, \ldots, i_N) \in \Omega} \mathcal{B}^2_{i_1 i_2 \cdots i_N} \Big(\frac{1}{\mathcal{P}_{i_1 i_2 \cdots i_N}} - \frac{1}{\widehat{\mathcal{P}}_{i_1 i_2 \cdots i_N}}\Big)^2 \\
&\le \psi^2 \sum_{(i_1, i_2, \ldots, i_N) \in \Omega} \Big(\frac{\mathcal{P}_{i_1 i_2 \cdots i_N} - \widehat{\mathcal{P}}_{i_1 i_2 \cdots i_N}}{\mathcal{P}_{i_1 i_2 \cdots i_N} \widehat{\mathcal{P}}_{i_1 i_2 \cdots i_N}}\Big)^2 \\
&\le \frac{\psi^2}{\sigma(-\gamma)^2 \sigma(-\alpha)^2} \sum_{(i_1, i_2, \ldots, i_N) \in \Omega} \Big(\mathcal{P}_{i_1 i_2 \cdots i_N} - \widehat{\mathcal{P}}_{i_1 i_2 \cdots i_N}\Big)^2 \\
&\le \frac{4eL_\gamma \tau \psi^2}{\sigma(-\gamma)^2 \sigma(-\alpha)^2} \Big(\frac{1}{\sqrt{I_S}} + \frac{1}{\sqrt{I_{S^C}}}\Big) I_{[N]}.
\end{aligned} \tag{4}$$

The second inequality comes from $\widehat{\mathcal{P}}_{i_1 i_2 \cdots i_N} \ge \sigma(-\gamma)$ and $\mathcal{P}_{i_1 i_2 \cdots i_N} \ge \sigma(-\alpha)$; the last inequality follows Lemma 2.

Then on each of the N unfoldings, the error of $\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}})$ from $\mathcal{B}^{(n)}$

$$\begin{aligned}
\|\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \mathcal{B}^{(n)}\| &\le \|\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \bar{\mathcal{X}}^{(n)}\| + \|\bar{\mathcal{X}}^{(n)} - \mathcal{B}^{(n)}\| \\
&\le \|\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \bar{\mathcal{X}}^{(n)}\|_{\mathrm{F}} + \|\bar{\mathcal{X}}^{(n)} - \mathcal{B}^{(n)}\|,
\end{aligned} \tag{5}$$

in which the first term can be bounded by Equation 4, and the second term can be bounded by applying the matrix Bernstein inequality [27] to the sum of $[\frac{\mathbf{1}(\mathcal{B}_{i_1 \cdots i_N} \text{ is observed})}{\mathcal{P}_{i_1 \cdots i_N}} \mathcal{B}_{\text{obs}} - \mathcal{B}] \odot \mathcal{E}(i_1, \ldots, i_N)$ over all entries.

The tensor $\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}$ is often full-rank; if we directly use $\sqrt{I} \cdot \|\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \mathcal{B}^{(n)}\|$ to bound $\|\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}\|_{\mathrm{F}}$, the final error bound we get would be $O(N\sqrt{\log I})$, which increases with the increase of tensor size I. Instead, we use the information of low multilinear rank to form the estimator $\widehat{\mathcal{X}}(\widehat{\mathcal{P}})$. In TenIPS (Algorithm 3),

$$\begin{aligned}
\widehat{\mathcal{X}}(\widehat{\mathcal{P}}) &= [\bar{\mathcal{X}}(\widehat{\mathcal{P}}) \times_1 Q_1^\top \times_2 \cdots \times_N Q_N^\top] \times_1 Q_1 \times_2 \cdots \times_N Q_N \\
&= \bar{\mathcal{X}}(\widehat{\mathcal{P}}) \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top,
\end{aligned}$$

in which each Q_n is the column space of $\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}})$. This projects each unfolding of $\bar{\mathcal{X}}(\widehat{\mathcal{P}})$ onto its truncated column space. By adding and subtracting $\mathcal{B} \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top$, the projection of $\mathcal{B}$ onto the column space of $\bar{\mathcal{X}}(\widehat{\mathcal{P}})$ in each mode, we get the estimation error

$$\begin{aligned}
&\|\widehat{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}\|_{\mathrm{F}}^2 \\
&= \|\bar{\mathcal{X}}(\widehat{\mathcal{P}}) \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top - \mathcal{B}\|_{\mathrm{F}}^2 \\
&= \|(\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}) \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top\|_{\mathrm{F}}^2 \\
&\quad + \|\mathcal{B} \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top - \mathcal{B}\|_{\mathrm{F}}^2 \\
&\quad + 2\langle (\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}) \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top, \\
&\qquad \mathcal{B} \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top - \mathcal{B} \rangle.
\end{aligned}$$

On the RHS, the third term is an inner product of two mutually orthogonal tensors and is thus 0. The first term is low multilinear rank, and thus can be bounded by the spectral norm of the unfoldings as

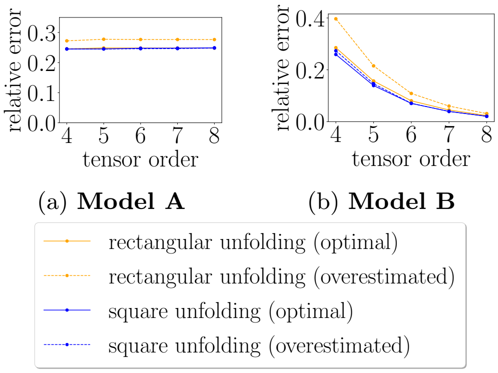
$$\begin{aligned}
&\|(\bar{\mathcal{X}}(\widehat{\mathcal{P}}) - \mathcal{B}) \times_1 Q_1 Q_1^\top \times_2 \cdots \times_N Q_N Q_N^\top\|_{\mathrm{F}}^2 \\
&\le \min_{n \in [N]} \Big\{ \|Q_n Q_n^\top (\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \mathcal{B}^{(n)})\|_{\mathrm{F}}^2 \Big\} \\
&\le \min_{n \in [N]} \Big\{ r \cdot \|\bar{\mathcal{X}}^{(n)}(\widehat{\mathcal{P}}) - \mathcal{B}^{(n)}\|^2 \Big\}.
\end{aligned}$$

The second term can be bounded by the sum of squares of residuals $\sum_{n \in [N]} \|\mathcal{B} \times_n (I - Q_n Q_n^\top)\|_{\mathrm{F}}^2$. Each of the summand here is the perturbation error of Q_n on the column space of $\mathcal{B}^{(n)}$, and thus can be bounded by the Davis-Kahan $\sin(\Theta)$ Theorem [28, 29, 30]. □

6 Experiments

All the code is in the GitHub repository at `https://github.com/udellgroup/TenIPS`. We ran all experiments on a Linux machine with Intel® Xeon® E7-4850 v4 2.10GHz CPU and 1056GB memory, and used the logistic link function $\sigma(x) = 1/(1 + e^{-x})$ throughout the experiments.

We use both synthetic and semi-synthetic data for evaluation. We first compare the propensity estimation performance under square and rectangular (along a specific mode) unfoldings, and then compare the tensor recovery error under different approaches. For both



(a) **Model A** (b) **Model B**

Figure 1: Propensity estimation performance on different orders of tensors with size 8. The small size makes it possible to evaluate high-order tensors within a reasonable time. Figure 1a follows Model A with a 40% observation ratio. Figure 1b follows Model B.

propensity estimation and tensor completion, the relative error is defined as $\|\widehat{\mathcal{T}} - \mathcal{T}\|_{\mathrm{F}}/\|\mathcal{T}\|_{\mathrm{F}}$, in which $\widehat{\mathcal{T}}$ is the predicted tensor and $\mathcal{T}$ is the true tensor.

There are four algorithms similar to TenIPS for tensor completion in our experiments: SqUnfold, which performs SVD to seek the low-rank approximation of the square unfolding of the propensity-reweighted $\mathcal{B}_{\mathrm{obs}}$; RectUnfold, which applies SVD to the unfolding of the propensity-reweighted $\mathcal{B}_{\mathrm{obs}}$ along a specific mode; HOSVD_w [19] and SO-HOSVD [16]. The popular nuclear-norm-regularized least squares LstSq that seeks $\hat{B} = \arg\min_X \sum_{(i,j)\in\Omega}(B_{ij} - X_{ij})^2 + \lambda\|X\|_\star$ on an unfolding of $\mathcal{B}_{\mathrm{obs}}$ takes much longer to finish in our experiments and is thus prohibitive in tensor completion practice, so we omit it from most of our results.

6.1 Synthetic data

We have the following observation models for synthetic tensors:

Model A. MCAR. The propensity tensor $\mathcal{P}$ has all equal entries.

Model B. MNAR with an approximately low multilinear rank parameter tensor $\mathcal{A}$. One special case is that $\mathcal{A}$ is proportional to $\mathcal{B}$: a larger entry is more likely to be observed.

In the first experiment, we evaluate the performance of propensity estimation on synthetic tensors with approximately low multilinear rank. In each of the above observation models, we generate synthetic cubical tensors with equal size on each mode, and predict the propensity tensor by estimating the parameter tensor on either the square or rectangular unfolding. Figure 1 compares the propensity estimation error on tensors with different orders. For each N, with $I = 8$ and $r = 2$, we generate an order-N parameter tensor $\mathcal{A}$ in the following way: We first generate $\mathcal{A}^\natural$ by Tucker decomposition $\mathcal{G}^A \times_1 U_1^A \times \cdots \times_N U_N^A \in \mathbb{R}^{I\times\cdots\times I}$, in which $\mathcal{G} \in \mathbb{R}^{r\times\cdots\times r}$ has i.i.d. $\mathcal{N}(0, 10^2)$ entries, and each $U_n^A \in \mathbb{R}^{I\times r}$ has random orthonormal columns. Then we generate a noise-corrupted $\mathcal{A}$ by $\mathcal{A}^\natural + (\gamma\|\mathcal{A}^\natural\|_{\mathrm{F}}/I^{N/2})\epsilon$, where the noise level $\gamma = 0.1$ and the noise tensor ϵ has i.i.d. $\mathcal{N}(0, 1)$ entries. The “optimal” hyperparameter setting uses $\tau = \theta$ and $\gamma = \alpha$ in 1-bit matrix completion, and the “overestimated” setting, $\tau = 2\theta$ and $\gamma = 2\alpha$. We can see that:

1 The square unfolding is always better than the rectangular unfolding in achieving a smaller propensity estimation error.

2 The propensity estimation error on the square unfolding increases less when the optimization hyperparameters τ and γ increase from their optima, θ and α. This suggests that the square unfolding makes the constrained optimization problem more robust to the selection of optimization hyperparameters.

3 It is most important to use the square unfolding for higher-order tensors: the ratios of relative errors between overestimated and optimal increase from 1.3 to 1.4, while those on the square unfoldings stay at 1.1. This aligns with Theorem 3 that the square unfolding outperforms in the worst case.

In the second experiment, we complete tensors with MCAR entries. With $N = 4$, $I = 100$ and $r = 5$, we generate an order-N data tensor $\mathcal{B} \in \mathbb{R}^{I\times\cdots\times I}$ in the following way: We first generate $\mathcal{B}^\natural$ by $\mathcal{G}^{\mathrm{true}} \times_1 U_1^{\mathrm{true}} \times_2 \cdots \times_N U_N^{\mathrm{true}}$, in which $\mathcal{G}^{\mathrm{true}} \in \mathbb{R}^{r\times\cdots\times r}$ has i.i.d. $\mathcal{N}(0, 100^2)$ entries, and each $U_n^{\mathrm{true}} \in \mathbb{R}^{I\times r}$ has random orthonormal columns. Then we generate a noise-corrupted $\mathcal{B}$ by $\mathcal{B}^\natural + (\gamma\|\mathcal{B}^\natural\|_{\mathrm{F}}/I^{N/2})\epsilon$, where the noise level $\gamma = 0.1$ and the noise tensor ϵ has i.i.d. $\mathcal{N}(0, 1)$ entries.

TenIPS is competitive for MCAR. We compare the relative error of TenIPS with SqUnfold, RectUnfold and HOSVD_w at different observation ratios in Figure 2. We can see that TenIPS and HOSVD_w achieve the lowest recovery error on average, and the results of these two methods are nearly identical.

In the third experiment, we complete tensors with MNAR entries. We use the same $\mathcal{B}$ as the second experiment, and further generate an order-4 parameter tensor $\mathcal{A} \in \mathbb{R}^{100\times\cdots\times 100}$ in the same way as $\mathcal{B}$. 99.97% of the propensities in $\mathcal{P} = \sigma(\mathcal{A})$ lie in the range of $[0.2, 0.8]$, as shown in Figure 3. In Table 2, we see:

1 TenIPS outperforms for MNAR. It has the smallest

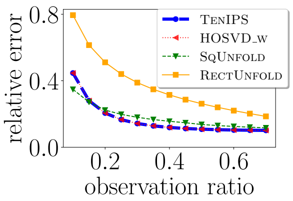


Figure 2: Error on MCAR tensors in the second experiment of Section 6.1.

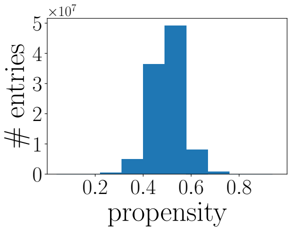


Figure 3: Propensity histogram in the third experiment of Section 6.1.

error among methods that can finish within a reasonable time.

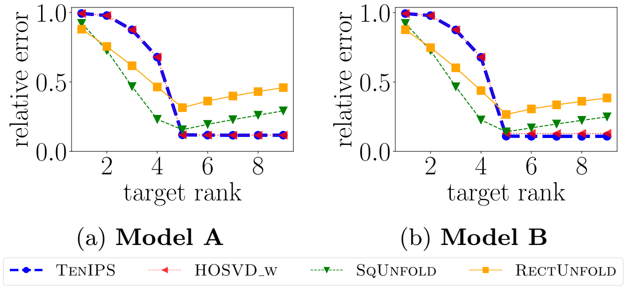
2 Tensor completion errors using estimated propensities are roughly equal to, and sometimes even smaller than those using true propensities despite a propensity estimation error.

3 On the sensitivity to hyperparameters: it is mentioned in the title of Table 2 that ConvexPE achieves a smaller accuracy within a similar time as NonconvexPE. However, this is because we set τ and γ correctly: $\tau = \theta$ and $\gamma = \alpha$. In real cases, θ and α are unknown, and are hard to infer from surrogate metrics within the optimization process. The misestimates of θ and α may lead to large propensity estimation errors: As an example, ConvexPE with $\tau = 100\theta$ and $\gamma = 100\alpha$ never achieves a relative error smaller than 0.7. Moreover, the relative error does not always decrease with more PPG iterations, despite the decrease of objective value. In every algorithm in Table 2, using these estimated propensities yields at least a relative error of 0.7 for the estimation of data tensor $\mathcal{B}$. On the other hand, the initialization and step size in NonconvexPE can be tuned more easily by monitoring the value of function f with the increase of number of iterations. More discussion can be found in Appendix D.

In the fourth experiment, we compare the above methods in both MCAR and MNAR settings when increasing target ranks. In Figure 4, we can see that both TenIPS and HOSVD_w are more stable at target ranks larger than the true rank, while RectUnfold and SqUnfold achieve smaller errors at smaller ranks. This shows that TenIPS and HOSVD_w are robust to large target ranks, which is the case when $r_n \geq r_n^{\text{true}}$, for all $n \in [N]$.

Table 2: Completion performance on the order-4 MNAR synthetic cubical tensor with size 100. The "time" here is the time taken for the tensor completion step, with true or estimated propensities. $\widehat{\mathcal{P}}_1$ is from running the provable ConvexPE (Algorithm 1) at target rank 25 for 84 seconds, and has relative error 0.08 from the true tensor $\mathcal{P}$; $\widehat{\mathcal{P}}_2$ is from running the gradient-descent-based NonconvexPE (Algorithm 2) with i.i.d. Uniform$[-1, 1]$ initialization and at step size 5×10^{-6} for 81 seconds, and has relative error 0.13. The bold number in each column indicates the best.

algorithm	time (s)	relative error from $\mathcal{B}$		
		with $\mathcal{P}$	with $\widehat{\mathcal{P}}_1$	with $\widehat{\mathcal{P}}_2$
TenIPS	26	**0.110**	**0.110**	**0.109**
HOSVD_w	35	0.129	0.116	0.110
SqUnfold	29	0.141	0.138	0.139
RectUnfold	**8**	0.259	0.256	0.256
LstSq	>600	-	-	-
SO-HOSVD	>600	-	-	-



(a) **Model A**

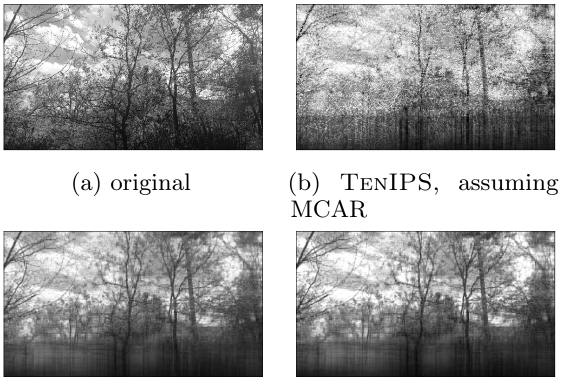
(b) **Model B**

Figure 4: Relative errors at different target ranks on the MNAR data from the third experiment. The true multilinear rank of the data tensor is 5 for each mode.

6.2 Semi-synthetic data

We use the video from [31] and generate synthetic propensities. The video was taken by a camera mounted at a fixed position with a person walking by. We convert it to grayscale and discard the frames with severe vibration, which yields an order-3 data tensor $\mathcal{B} \in \mathbb{R}^{2200\times1080\times1920}$ that takes 102.0GB memory. To get an MNAR tensor $\mathcal{B}_{\text{obs}}$, we generate the parameter tensor $\mathcal{A}$ by entrywise transformation $\mathcal{A} = (\mathcal{B} - 128)/64$, which gives propensities in $[0.12, 0.88]$ in $\mathcal{P} = \sigma(\mathcal{A})$. Finally we subsample $\mathcal{B}$ using propensities $\mathcal{P}$ to get $\mathcal{B}_{\text{obs}}$.

We first compare the tensor completion performance of TenIPS with different sources of propensities. In Figure 5, we visualize the 500-th frame in three TenIPS experiments by fixed-rank approximation with target multilinear rank $(50, 50, 50)$: the original frame without missing pixels 5a, the frame recovered under MCAR assumption (tensor recovery error 0.42) 5b, the frame recovered by propensities under the MNAR assumption with the true propensity tensor $\mathcal{P}$ (tensor recovery error 0.28) 5c, and the frame recovered by propensities under the MNAR assumption with the estimated propensity



(a) original

(b) TenIPS, assuming MCAR

(c) TenIPS, assuming MNAR with true $\mathcal{P}$

(d) TenIPS, assuming MNAR with estimated $\widehat{\mathcal{P}}$

Figure 5: Video recovery visualization on Frame 500 of the [31] video data. The missingness patterns in 5b, 5c and 5d only refer to our assumption in tensor recovery; the partially observed data tensors we start from are the same and are MNAR.

Table 3: Completion performance on the video task. The memory usages are those in Python 3. The bold number in each row indicates the best.

Setting	relative error (memory in GB)		
	TenIPS	HOSVD_w	RectUnfold
I	0.28 (0.008)	0.44 (0.008)	**0.14** (2.3)
II	**0.20** (0.05)	0.31 (0.05)	0.25 (0.05)

tensor $\widehat{\mathcal{P}}$ from ConvexPE (propensity estimation error 0.15, tensor recovery error 0.28) 5d. We can see that:

1 With MNAR pixels, the image recovered from the naive MCAR assumption in Figure 5b is more noisy than that from MNAR in Figure 5c and 5d, and misses more details.

2 There is no significant difference between the recovered video frames in 5c and 5d, in terms of both the frame image itself and the tensor recovery error.

We then compare TenIPS with HOSVD_w and RectUnfold on this video task; we omit SqUnfold and SO-HOSVD because SqUnfold and RectUnfold are equivalent on an order-3 tensor and SO-HOSVD cannot finish within a reasonable time. In Table 3 Setting I, TenIPS and HOSVD_w have target rank $(50, 50, 50)$, and RectUnfold has target rank 50. We can see that RectUnfold has a smaller error in this setting but uses more than 250× memory, because it does not seek low dimensional representations along the two dimensions of the video frame.

Another advantage of the tensor methods TenIPS and HOSVD_w compared to RectUnfold is that the target rank for different modes is not required to be the same. For example, if we limit memory usage to 0.05GB (Table 3, Setting II), TenIPS and HOSVD_w can afford a target rank $(5, 500, 500)$ and achieve smaller errors, while RectUnfold can only afford a target rank of 1.

Also, with similar memory consumption in both settings, TenIPS achieves smaller errors than HOSVD_w.

7 Conclusion

This paper develops a provable two-step approach for MNAR tensor completion with unknown propensities. The square unfolding allows us to recover propensities with a smaller upper bound, and we then use HOSVD complete MNAR tensor with the estimated propensities. This method enjoys theoretical guarantee and fast running time in practice.

This paper is the first provable method for completing a general MNAR tensor. There are many avenues for improvement and extensions. For example, one could explore whether nonconvex matrix completion methods can be generalized to MNAR tensors, explore other observation models, and design provable algorithms that estimate the propensities even faster.

Acknowledgements

MU, CY, and LD gratefully acknowledge support from NSF Awards IIS-1943131 and CCF-1740822, the ONR Young Investigator Program, DARPA Award FA8750-17-2-0101, the Simons Institute, Canadian Institutes of Health Research, the Alfred P. Sloan Foundation, and Capital One. The authors thank Jicong Fan for helpful discussions, and thank several anonymous reviewers for useful comments.

References

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

[2] Sahand Negahban and Martin J Wainwright. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research*, 13(May):1665–1697, 2012.

[3] Tony Cai and Wen-Xin Zhou. A max-norm constrained minimization approach to 1-bit matrix

completion. *The Journal of Machine Learning Research*, 14(1):3619–3647, 2013.

[4] J Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35(3):283–319, 1970.

[5] Richard A Harshman et al. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. 1970.

[6] Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[7] Ivan V Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[8] Akshay Krishnamurthy and Aarti Singh. Low-rank matrix and tensor completion via adaptive sampling. In *Advances in Neural Information Processing Systems*, pages 836–844, 2013.

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

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

[11] Morteza Ashraphijuo and Xiaodong Wang. Fundamental conditions for low-cp-rank tensor completion. *The Journal of Machine Learning Research*, 18(1):2116–2145, 2017.

[12] Navid Ghadermarzy, Yaniv Plan, and Ozgur Yilmaz. Learning tensors from partial binary measurements. *IEEE Transactions on Signal Processing*, 67(1):29–40, 2018.

[13] Allen Liu and Ankur Moitra. Tensor completion made practical. *arXiv preprint arXiv:2006.03134*, 2020.

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

[15] Cun Mu, Bo Huang, John Wright, and Donald Goldfarb. Square deal: Lower bounds and improved relaxations for tensor recovery. In *International Conference on Machine Learning*, pages 73–81, 2014.

[16] Dong Xia, Ming Yuan, and Cun-Hui Zhang. Statistically optimal and computationally efficient low rank tensor completion from noisy entries. *arXiv preprint arXiv:1711.04934*, 2017.

[17] Tatsuya Yokota, Burak Erem, Seyhmus Guler, Simon K Warfield, and Hidekata Hontani. Missing slice recovery for tensors using a low-rank model in embedded space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8251–8259, 2018.

[18] Anru Zhang et al. Cross: Efficient low-rank tensor completion. *The Annals of Statistics*, 47(2):936–964, 2019.

[19] Longxiu Huang and Deanna Needell. Hosvd-based algorithm for weighted tensor completion. *arXiv preprint arXiv:2003.08537*, 2020.

[20] Wenqi Wang, Vaneet Aggarwal, and Shuchin Aeron. Tensor completion by alternating minimization under the tensor train (TT) model. *arXiv preprint arXiv:1609.05587*, 2016.

[21] Longhao Yuan, Qibin Zhao, Lihua Gui, and Jianting Cao. High-dimension tensor completion via gradient-based optimization under tensor-train format. *arXiv preprint arXiv:1804.01983*, 2018.

[22] Ryota Tomioka, Kohei Hayashi, and Hisashi Kashima. Estimation of low-rank tensors via convex optimization. *arXiv preprint arXiv:1010.0789*, 2010.

[23] Anastasia Aidini, Grigorios Tsagkatakis, and Panagiotis Tsakalides. 1-bit tensor completion. *Electronic Imaging*, 2018(13):261–1, 2018.

[24] Mark A Davenport, Yaniv Plan, Ewout Van Den Berg, and Mary Wootters. 1-bit matrix completion. *Information and Inference: A Journal of the IMA*, 3(3):189–223, 2014.

[25] Wei Ma and George H Chen. Missing not at random in matrix completion: The effectiveness of estimating missingness probabilities under a low nuclear norm assumption. In *Advances in Neural Information Processing Systems*, pages 14871–14880, 2019.

[26] Ernest K Ryu and Wotao Yin. Proximal-proximal-gradient method. *arXiv preprint arXiv:1708.06908*, 2017.

[27] Joel A Tropp et al. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015.

[28] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[29] Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12(1):99–111, 1972.

[30] Yi Yu, Tengyao Wang, and Richard J Samworth. A useful variant of the davis–kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

[31] Osman Asif Malik and Stephen Becker. Low-rank tucker decomposition of large tensors using tensorsketch. In *Advances in Neural Information Processing Systems*, pages 10096–10106, 2018.