Tensor Sandwich: Tensor Completion for Low CP-Rank Tensors via Adaptive Random Sampling

***Abstract*—We propose an adaptive and provably accurate tensor completion approach based on combining matrix completion techniques (see, e.g., [1]–[3]) for a small number of slices with a modified noise robust version of Jennrich's algorithm (see, e.g., [4, Section 3]). In the simplest case, this leads to a sampling strategy that more densely samples two outer slices (the bread), and then more sparsely samples additional inner slices (the bbq-braised tofu) for the final completion. Under mild assumptions on the factor matrices, the proposed algorithm completes an $n \times n \times n$ tensor with CP-rank r with high probability while using at most $\mathcal{O}(nr \log^2 r)$ adaptively chosen samples. Empirical experiments further verify that the proposed approach works well in practice, including as a low-rank approximation method in the presence of additive noise.**

I. Introduction

Consider a CP rank-r tensor $\mathcal{T} \in \mathbb{R}^{n\times n\times n}$ where $r \leq n$ with CP-decomposition

$$\mathcal{T} = \sum_{i=1}^{r} \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i. \tag{I.1}$$

Here $\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i \in \mathbb{R}^n$ for all $i \in [r] := \{1, \ldots, r\}$, and $\circ$ denotes the outer product so that $\mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i \in \mathbb{R}^{n\times n\times n}$ has entries $(\mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i)_{h,j,k} = (\mathbf{a}_i)_h (\mathbf{b}_i)_j (\mathbf{c}_i)_k$ for all $i \in [r]$. The tensor completion problem is to reconstruct $\mathcal{T}$ after observing a subset of its entries $\mathcal{T}_{i,j,k}$ for $(i,j,k) \in \Omega \subset [n]^3$. In this paper Ω will consist of random entries adaptively sampled in several rounds (i.e., so that the entries observed in round $\ell + 1$ are selected via a distribution that depends on the entries observed in rounds 1 to ℓ). We hasten to add, however, that the general proof approach taken herein can itself be adapted to also yield tensor completion results based on non-adaptive sampling at the expense of further restricting the class of tensors considered below for which the methods will be guaranteed to succeed.

In order to define the class of tensors for which the proposed approach will be guaranteed to succeed we need several definitions. Define the *factor matrix* $\boldsymbol{A} = \begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_r\end{bmatrix} \in \mathbb{R}^{n\times r}$ to have columns given by the factor vectors $\mathbf{a_j}$ in (I.1), as well as for $\boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{n\times r}$ with columns arranged in the same corresponding orders. Also, define the *coherence* of an r-dimensional subspace $U \subset \mathbb{R}^n$ to be

$$\mu(U) := \frac{n}{r} \max_{i\in[n]} \|\mathcal{P}_U \boldsymbol{e}_i\|_2^2,$$

where $\mathcal{P}_U$ is the orthogonal projection onto U, and where $\boldsymbol{e}_i \in \mathbb{R}^n$ for $i \in [n]$ denotes the ith standard basis vector. Finally, the *Kruskal rank of a matrix* is the maximum integer r such that any r columns of the matrix are linearly independent. We are now able to state our main result.

Theorem 1. *Suppose $\mathcal{T} = \sum_{i=1}^{r} \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i \in \mathbb{R}^{n\times n\times n}$ is a CP rank-r tensor* (I.1) *with $r \leq n$ such that:*

(a) The factor matrices $\mathbf{A}, \mathbf{B}$ both have full column rank,
(b) the column space of $\mathbf{A}$ has coherence bounded above by μ_0, and
(c) for some $s \in [n]$, every $s \times r$ submatrix of $\mathbf{C}$ has Kruskal rank ≥ 2.

Then, for any $\delta > 0$, there exists an adaptive random sampling strategy and an associated reconstruction algorithm (see Algorithm 2) that, after observing at most $C_1 s\mu_0 nr \log^2(r^2/\delta)$ entries of $\mathcal{T}$, completes $\mathcal{T}$ with probability at least $1 - s\delta$. Here $C_1 > 0$ is absolute constant that is independent of all other quantities.

The proof of our sandwich sampling algorithm involves three stages. First, w.l.o.g., we pick s mode-3 slices and then use a matrix completion algorithm that with high probability recovers these slices and observes at most $C_1' \mu_0 nr \log^2(r^2/\delta)$ entries in each slice. Second, if this $n \times n \times s$ subtensor is correctly completed, we can then use a deterministic variant of Jennrich's Algorithm on the completed subtensor to learn the factor matrices $\mathbf{A}$ and $\mathbf{B}$. Third, once we know the factor matrices $\mathbf{A}$ and $\mathbf{B}$, we can deterministically find r sample locations in each of the n mode-3 slices whose values allow a censored least squares problem to solve for the third factor matrix $\mathbf{C}$. This three-stage procedure uses at most $C_1' s\mu_0 nr \log^2(r^2/\delta)$ samples to complete the s initial mode-3 slices in order to learn $\mathbf{A}$ and $\mathbf{B}$, and then nr additional samples to learn $\mathbf{C}$ thereafter, for a total of at most $C_1' s\mu_0 nr \log^2(r^2/\delta) + nr \leq C_1 s\mu_0 nr \log^2(r^2/\delta)$ samples. See Figure 1 for a schematic illustration of the overall sampling strategy where fibers are sampled through the middle of the sandwich for simplicity.

We note that assumptions a and c are the necessary assumptions for the Jennrich's step to work with any $n \times n \times s$ subtensor. Furthermore, if the columns of the factor matrices are drawn from any continuous distribution, assumption c for $s = 2$ holds with probability 1. Finally, something akin to assumption b is always required in completion problems.

A. Related Work

Many prior works on low-rank tensor completion use non-adaptive and uniform sampling [5]–[12]. While some of those works [6], [10], [12] can handle CP-ranks up to roughly $n^{3/2}$ instead of n, all of them require at least $\mathcal{O}(n^{3/2})$ samples,

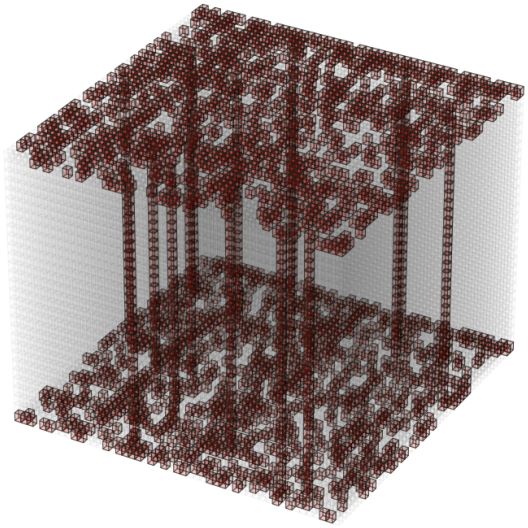

Fig. 1. A schematic depiction of the sampling strategy where $s = 2$ slices have been sampled relatively densely in order to compute $\mathbf{A}$ and $\mathbf{B}$, and where additional fibers where then sampled elsewhere to help compute C

even when the rank is $r = \mathcal{O}(1)$. Furthermore, [6] shows that completing an $n \times n \times n$ rank-r tensor from $n^{3/2-\epsilon}$ uniformly random samples is NP-hard by comparison to the problem of refuting a 3-SAT formula with n variables and $n^{3/2-\epsilon}$ clauses.

In [13], the authors propose a method for completing CP-rank $r \leq n$ tensors using adaptive sampling which, for order-3 tensors, requires $\mathcal{O}(\mu_0^2 n r^{5/2} \log^2 r)$ samples. Our algorithm requires a number of samples which has a more favorable dependence on the coherence μ_0 and rank r. Furthermore, our result only requires coherence assumptions about $\mathbf{A}$, instead of about $\mathbf{A}$ *and* $\mathbf{B}$. These improvements come at the expense of requiring the mild additional assumptions a and c in our Theorem 1 related to Jennrich's algorithm, however.

The first step in our tensor completion algorithm involves using an adaptive sampling algorithm to complete s mode-3 slices of the tensor. Our tensor completion algorithm and results are based on the adaptive matrix completion algorithm and results in [2] which, with high probability, uses $\mathcal{O}(\mu_0 n r \log^2 r)$ samples to complete a rank r matrix. However, our algorithm can be adapted to use other adaptive matrix completion results such as, e.g., [3] with relative ease.

In the censored least squares phase of our algorithm, we can sample entire fibers of the tensor as done in Figure 1. We note that doing so is similar in spirit to the fiber sampling approach of Sørensen and De Lathauwer [14]. However, their work focuses on determining algebraic constraints on the factor matrices of a low rank tensor which, when satisfied, allow the tensor to be completed from the sampled fibers. As such, our results cannot be directly compared with [14].

II. Proof of Theorem 1

In this section we follow our 3 stage proof outline.

A. Completing s mode-3 slices of T

We start by picking any subset of indices $S \subset [n]$ with $|S| = s$ elements. For each $k \in S$, the mode-3 slice $\mathcal{T}_{:,:,k} \in \mathbb{R}^{n \times n}$ satisfies

$$\mathcal{T}_{:,:,k} = \sum_{i=1}^{r} \langle \mathbf{c}_i, \mathbf{e}_k \rangle \mathbf{a}_i \mathbf{b}_i^T,$$

and so, col-span$(\mathcal{T}_{:,:,k}) \subseteq$ col-span$(\mathbf{A})$. By assumption, col-span$(\mathbf{A})$ has coherence bounded by μ_0. Thus, the assumptions required by Theorem 1 in [2] hold. Therefore, for each slice $T_{:,:,k}$ $k \in S$, with probability at least $1 - \delta$, the adaptive sampling procedure in Algorithm 1 uses at most $C_1' \mu_0 n r \log^2(r^2/\delta)$ samples for some absolute constant $C_1' > 0$, and completes $\mathcal{T}_{:,:,k}$.

By taking a simple union bound over each of the s slices $k \in S$, we have that with probability at least $1 - s\delta$, this strategy will successfully complete all of the s slices $\mathcal{T}_{:,:,k}$ for $k \in S$ and use fewer than $C_1' s \mu_0 n r \log^2(r^2/\delta)$ samples. Let $\Omega_1 = \{(i,j,k) | (i,j) \text{ sampled according to [2] for slice } k \in S\}$, the set of locations of T sampled to complete these s slices.

B. Learning mode-1 and 2 factor matrices via Modified Jennrich's Algorithm

Let $\mathbf{u}, \mathbf{v}$ be random vectors uniformly drawn from the unit sphere $\mathbb{S}^{s-1}$. Denote the sub-vector of $\mathbf{c}$ with entrees indexed in S by $\tilde{\mathbf{c}}_i = (\mathbf{c}_i)_S$, and construct two auxiliary matrices $\mathbf{T}_u, \mathbf{T}_v$ using the completed slices by adding up the linear combinations of the completed slices, weighted by the random vectors $\mathbf{u}, \mathbf{v}$. In terms of the components, we have:

$$\mathbf{T}_u = \sum_{i=1}^{r} \langle \tilde{\mathbf{c}}_i, \mathbf{u} \rangle \mathbf{a}_i \mathbf{b}_i^T$$

$$\mathbf{T}_v = \sum_{i=1}^{r} \langle \tilde{\mathbf{c}}_i, \mathbf{v} \rangle \mathbf{a}_i \mathbf{b}_i^T$$

Denote the $r \times r$ matrix $\mathbf{D}_u$ which has along its diagonal entries the values $\langle \tilde{\mathbf{c}}_i, \mathbf{u} \rangle$, and similarly $\mathbf{D}_v$. Notice the following identity with the product $\mathbf{T}_u (\mathbf{T}_v)^\dagger$

$$\begin{aligned} \mathbf{T}_u (\mathbf{T}_v)^\dagger &= \mathbf{A} \mathbf{D}_u \mathbf{B}^T \left(\mathbf{A} \mathbf{D}_v \mathbf{B}^T \right)^\dagger \\ &= \mathbf{A} \mathbf{D}_u \mathbf{D}_v^{-1} \mathbf{A}^\dagger \end{aligned}$$

Where the matrix $\mathbf{D}_u \mathbf{D}_v^{-1}$ is a diagonal matrix with (i,i)-entry equal to $\frac{\langle \tilde{\mathbf{c}}_i, \mathbf{u} \rangle}{\langle \tilde{\mathbf{c}}_i, \mathbf{v} \rangle}$. Clearly the matrix $\mathbf{T}_u (\mathbf{T}_v)^\dagger$ is diagonalizable, and so by computing the eigen-decomposition of $\mathbf{T}_u (\mathbf{T}_v)^\dagger$ we recover the columns of $\mathbf{A}$ and the eigenvlaues along the diagonal of $\mathbf{D}_u \mathbf{D}_v^{-1}$. We can order the eigenvectors in descending order by magnitude of their corresponding eigenvalue. Note it is here that we employ assumption c: Since $\mathbf{C}_S$ (i.e., the rows of $\mathbf{C}$ indexed by S) has k-rank at least 2, the ratios $\frac{\langle \tilde{\mathbf{c}}_i, \mathbf{u} \rangle}{\langle \tilde{\mathbf{c}}_i, \mathbf{v} \rangle}$ for each i will almost surely be distinct from one another, and thus the ordering of the eigenvalues is unique.

Let $\mathbf{P}$ be the permuation matrix that interchanges the columns of $\mathbf{A}$ so that instead of being in $\mathbf{D}_u \mathbf{D}_v^{-1}$ order, they

are in $\mathbf{D}_u$ order (ordered greatest to least in terms of the magnitude of $\langle \tilde{\mathbf{c}}_i, \mathbf{u} \rangle$). Now notice that

$$\begin{aligned} \mathbf{A}^\dagger \mathbf{T}_u &= \mathbf{A}^\dagger \mathbf{A} \mathbf{P} \mathbf{D}_u \mathbf{B}^T \\ &= \mathbf{P} \mathbf{D}_u \mathbf{B}^T \\ &= (\mathbf{B} \mathbf{D}_u \mathbf{P})^T. \end{aligned} \tag{II.1}$$

This means that the rescaled columns of $\mathbf{B}$ that are in $\mathbf{D}_u$ order are now in $\mathbf{D}_u \mathbf{D}_v^{-1}$ order after we apply the inverse $\mathbf{A}^\dagger$ to $\mathbf{T}_u$. That is, we have found the matching components of $\mathbf{A}$ and $\mathbf{B}$ up to a re-scaling of their outer product! This was achieved by learning the columns of $\mathbf{A}$ from $\mathbf{T}_u(\mathbf{T}_v)^\dagger$ and then, crucially, the matching columns of $\mathbf{B}$ from $\mathbf{A}^\dagger \mathbf{T}_u$. The scaling due to $\mathbf{D}_u$ will be resolved in the final step of the algorithm, where we solve for the missing third factor (i.e., $\mathbf{B}, \mathbf{C}$ and $\mathbf{B}\mathbf{D}, \mathbf{C}(\mathbf{D}^{-1})$ are both valid pairs of factors for any diagonal matrix $\mathbf{D}$).

C. Learning the mode-3 factor matrix

After obtaining $\mathbf{A}$ and a rescaled $\mathbf{B}$ in order to find the remaining components, $\mathbf{C}$, we will need Ω_2, a second set of revealed locations of the entries of $\mathcal{T}$. We will also need the solutions to n instances of the following censored least square problem related to those revealed values

$$(\mathbf{A} \odot \mathbf{B})_{K_k} \mathbf{c} = \mathbf{t}_{K_k} \tag{II.2}$$

where $k \in [n]$,$\mathbf{c} = (\mathbf{C}_{k,:})^T$, $\mathbf{t} = \text{vec}(\mathcal{T}_{:,:,k})^T$ and $K_k = \{i + n(j-1) | (i,j,k) \in \Omega_2\}$. In (II.2), the term $\mathbf{t}_{K_k}$ denotes the vector of length $|K_k|$ which includes only the entries of $\mathbf{t}$ which have indicies appearing in the set K_k. Similarly $(\mathbf{A} \odot \mathbf{B})_{K_k}$ is the matrix where we restrict rows of $(\mathbf{A} \odot \mathbf{B})$ to only those which have indices appearing in the set.

Note $(\mathbf{A} \odot \mathbf{B})$ is full rank because $\mathbf{A}$ and $\mathbf{B}$ are assumed to be full rank (see, e.g., [15]). This implies the uncensored system is consistent with a unique solution. The difficulty in the censored case is that unobserved values in a particular column of the right hand side of (II.2) could force the discarding of rows of the matrix $(\mathbf{A} \odot \mathbf{B})$ that cause the system to become under-determined.

That is, we must ensure that $(\mathbf{A} \odot \mathbf{B})_{K_k}$ has full column rank for each k. If we do not vary our sampling procedure from frontal slice to frontal slice, then this corresponds to sampling tubes of the original tensor of the form $\mathcal{T}_{i,j,:}$ and $K_k = K$ for all $k \in [n]$. In order to arrange the sampling to accomplish this, consider a $\mathbf{QR}$ with column-pivoting factorization of $(\mathbf{A} \odot \mathbf{B})^T$, see algorithm 5.4.1 in [16]). This produces factors and a permutation matrix $\mathbf{\Pi} \in \mathbb{R}^{n^2 \times n^2}$ such that $(\mathbf{A} \odot \mathbf{B})^T \mathbf{\Pi} = \mathbf{QR}$, where

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{R}_2 \end{bmatrix}$$

and $\mathbf{R}_1 \in \mathbb{R}^{r \times r}$ is upper-triangular and non-singular. But this means that the first r columns of $\mathbf{\Pi}$ selects a set of columns of $(\mathbf{A} \odot \mathbf{B})^T$ which are linearly independent, so for each $\mathbf{\Pi}_{:,p}$, $p \in [r]$ we have that there exists some $q \in [n^2]$ such that $\mathbf{e}_q = \mathbf{\Pi}_{:,p}$, the corresponding column of the identity in $\mathbb{R}^{n^2 \times n^2}$. Let $i = q \mod n, j = \lfloor \frac{q}{n} \rfloor$. Define then Ω_2 all the tuples of the form $(i, j, :)$. That is, we can read off from $\mathbf{\Pi}$ which r fibers of length n to sample in order to ensure (II.2) is consistent for each $k \in [n]$. We have specified at most nr new sample locations, and thus $\Omega = \Omega_1 \cup \Omega_2$, the set of all samples employed to complete the tensor $\mathcal{T}$ is at most $|\Omega_1| + |\Omega_2| \leq C_1' s \mu_0 nr \log^2(r^2/\delta) + nr \leq C_1 s \mu_0 nr \log^2(r^2/\delta)$.

Remark 1. Note, $\mathbf{\Pi}$ is not unique, and indeed we could use different selections corresponding to other valid permutations from frontal slice to frontal slice to vary the sampling pattern when finding $\mathbf{C}$. Additionally, including more rows of $(\mathbf{A} \odot \mathbf{B})$ beyond the first r as specified by $\mathbf{\Pi}$ may be numerically advantageous when computing $\mathbf{C}$.

Algorithm 1 Consistency Preserving Fiber Sampler

input :
$\gamma \geq 1$, over-sample parameter
r rank parameter
$\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times r}$, estimates for factor matrices
output: Ω_2, set of sample locations
Compute $\mathbf{QR}$ with column-pivoting, $\mathbf{QR} = (\mathbf{A} \odot \mathbf{B})^T \mathbf{\Pi}$
for $p \in [\gamma r]$ **do**
  $q \leftarrow \text{nonzero}(\mathbf{\Pi}_{:,p})$
  $i \leftarrow q \mod n$
  $j \leftarrow \lfloor \frac{q}{n} \rfloor$
  Include (i, j, k) in Ω_2 for all $k \in [n]$
end

Algorithm 2 Tensor Sandwich

input : $S \subseteq [n]$ slices to complete
output: $\hat{T}$, completed tensor
`# Slice Complete Phase`
for $k \in S$ **do**
  Use, e.g., [3], or algorithm 1 in [2] to complete $\mathcal{T}_{:,:,k}$. Algorithm 1 in [2] uses at most $C_1' \mu_0 nr \log^2(r^2/\delta)$ adaptively chosen samples.
end
`# Jennrich Complete Phase`
Generate random vectors $\mathbf{u}, \mathbf{v} \in \mathbb{S}^{s-1}$;
$\mathbf{T}_u \leftarrow \sum_{k \in S} u_k \mathbf{T}_{::k}$
$\mathbf{T}_v \leftarrow \sum_{k \in S} v_k \mathbf{T}_{::k}$
Compute eigen-decomposition $\mathbf{A} \mathbf{\Lambda} \mathbf{A}^{-1} = \mathbf{T}_u (\mathbf{T}_u)^\dagger$
$\mathbf{B} \leftarrow \mathbf{A}^{-1} \mathbf{T}_u$
`# Censored Least Squares Phase`
$\Omega_2 \leftarrow$ fiber sampler($\mathbf{A}, \mathbf{B}$,r,γ)
$\mathbf{C}^T \leftarrow$ censored least squares $((\mathbf{A} \odot \mathbf{B}), \mathcal{T}_{i,j,k}$ s.t. $(i,j,k) \in \Omega_2)$
$\hat{\mathcal{T}} \leftarrow [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$

III. Experiments

In this section, we show that tensors sampled according to our overall strategy and completed using Algorithm (2) support our theoretical findings. In particular we demonstrate that once sample complexity bounds are satisfied, we can achieve very precise levels of relative error by sampling what is overall

a small percentage of the total tensor's entries. For example, in the rank 20 case in Figure 2, the mean relative error is 0.0068 having sampled at most $\frac{43961}{8000000} \approx 0.55\%$ of the total entries. Our experiments also verify the linear dependence on rank in the sample complexity bound. Furthermore, in our second set of experiments we show that the method performs useful completion even in the presence of noise. The code and results from the numerical experiments is available in the supplementary material.

In all our experiments, the three modes of the tensor have length $n = 200$. The data is generated by drawing factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathbb{R}^{n \times r}$ with i.i.d standard Guassian entries and then normalizing the columns. We then weight the components using quadratically decaying weights, i.e.

$$\mathcal{T} = \sum_{i=1}^{r} \left(\frac{1}{i^2} \right) \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i. \tag{III.1}$$

Errors are averaged over ten independent trials.

For each of our trials, $s = 2$. After uniformly selecting two frontal slices to complete, within these slices we sample per Algorithm 1 in [3] using a sample budget of m samples per slice. Slices are then completed using the SDP formulation of the nuclear norm minimization solved via Douglas-Rachford splitting, see [17]. Convergence is declared once the primal residual, dual residual and duality gap are below 10^{-6} for each of the s selected slices.

Following matrix completion on these frontal slices and estimation of $\mathbf{A}$ and $\mathbf{B}$, We select γr rows of $(\mathbf{A} \odot \mathbf{B})$ using Algorithm 1 and the corresponding fibers of $\mathcal{T}$ to solve the censored least square problem. In all the experiments in this section, $\gamma = 4$. This results in at most $\gamma n r$ new entries revealed in the original tensor. Total number of entries sampled then is always bounded by $sm + \gamma n r$ (some overlap of samples is possible in the two slices already completed). Note that for all experiments $0.001 n r \log^2(n) < m < 2 n r \log^2(n)$.

In figure 2, we see a phase transition of our method. For a given rank, prior to a threshold, the error is dominated by the inability to accurately complete the s slices. Once sufficient samples are obtained within these slices, completion reliably succeeds, and the small relative errors in the less than 1% range in this region are dominated by other considerations, such as halting criteria for the slice completion step. We see in Figure 2 empirically the linear relationship of the sample complexity on the rank of the tensor - it takes approximately twice as many samples to reach this phase transition in terms of relative error when we compare the rank 5 case with the rank 10 case.

In figure 3, we add mean-zero i.i.d. Gaussian noise to each entry in our tensor and perform the same completion strategy on the noisy tensor as described earlier in this section. For each trial, the noise tensor $\mathcal{N}$ is scaled to the appropriate signal-to-noise ratio, i.e. $\text{SNR} = 10 \log_{10} \frac{\|\mathcal{T}\|}{\|\mathcal{N}\|}$. Weights $\frac{1}{i^2}$ in (III.1) are all set to 1 for these experiments. The sample complexity proportions are fixed with respect to slice completion versus fibers sampled to estimate $\mathbf{C}$ but do scale according to rank in order to facilitate comparison. In all cases the total number of entries is between only 0.43% and 0.75% of the total number of entries.

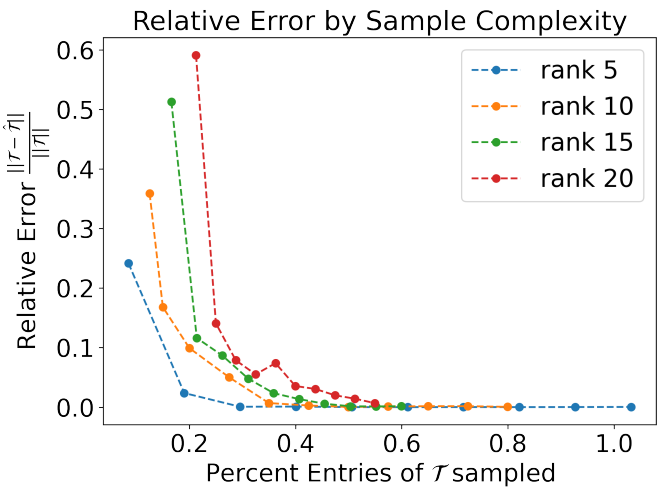

Fig. 2. Mean relative error of completed tensors of varying rank as sample complexity increases. Each value is the average of ten trials.

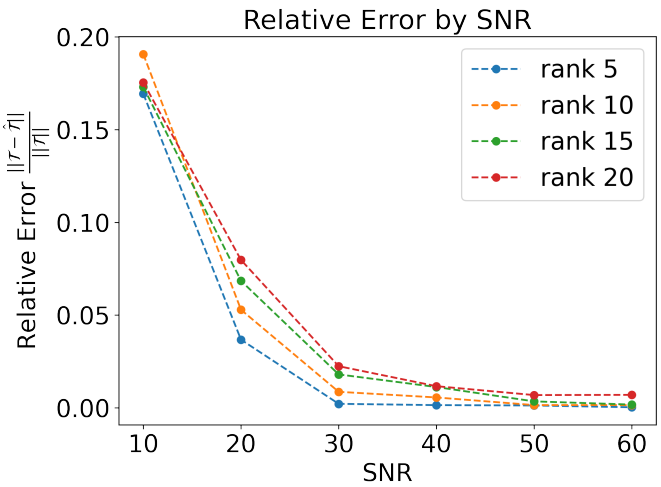

Fig. 3. Median relative error of completed tensors of varying rank for different levels of white Gaussian noise. Sample complexity is less than 0.75% for all trials, scaled linearly by rank.

References

[1] E. Candes and B. Recht, "Exact matrix completion via convex optimization," *Communications of the ACM*, vol. 55, no. 6, pp. 111–119, 2012.

[2] A. Krishnamurthy and A. Singh, "On the power of adaptivity in matrix completion and approximation," 2014.

[3] Y. Chen, S. Bhojanapalli, S. Sanghavi, and R. Ward, "Completing any low-rank matrix, provably," *Journal of Machine Learning Research*, vol. 16, no. 94, pp. 2999–3034, 2015. [Online]. Available: http://jmlr.org/papers/v16/chen15b.html

[4] A. Moitra, *Algorithmic aspects of machine learning*. Cambridge University Press, 2018.

[5] P. Jain and S. Oh, "Provable tensor factorization with missing data," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[6] B. Barak and A. Moitra, "Noisy tensor completion via the sum-of-squares hierarchy," in *Conference on Learning Theory*. PMLR, 2016, pp. 417–445.

[7] M. Yuan and C.-H. Zhang, "On tensor completion via nuclear norm minimization," *Foundations of Computational Mathematics*, vol. 16, no. 4, pp. 1031–1068, 2016.

[8] ——, "Incoherent tensor norms and their applications in higher order tensor completion," *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6753–6766, 2017.

[9] A. Potechin and D. Steurer, "Exact tensor completion with sum-of-squares," in *Conference on Learning Theory*. PMLR, 2017, pp. 1619–1673.

[10] A. Montanari and N. Sun, "Spectral algorithms for tensor completion," *Communications on Pure and Applied Mathematics*, vol. 71, no. 11, pp. 2381–2425, 2018.

[11] A. Liu and A. Moitra, "Tensor completion made practical," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 905–18 916, 2020.

[12] B. Kivva and A. Potechin, "Exact nuclear norm, completion and decomposition for random overcomplete tensors via degree-4 sos," *arXiv preprint arXiv:2011.09416*, 2020.

[13] A. Krishnamurthy and A. Singh, "Low-rank matrix and tensor completion via adaptive sampling," *Advances in neural information processing systems*, vol. 26, 2013.

[14] M. Sørensen and L. De Lathauwer, "Fiber sampling approach to canonical polyadic decomposition and application to tensor completion," *SIAM Journal on Matrix Analysis and Applications*, vol. 40, no. 3, pp. 888–917, 2019.

[15] T. B. JM., "The k-rank of a khatri–rao product," *Unpublished Note, Heijmans Institute of Psychological Research, University of Groningen, The Netherlands.*, 2000.

[16] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.

[17] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "Conic optimization via operator splitting and homogeneous self-dual embedding," *Journal of Optimization Theory and Applications*, vol. 169, no. 3, pp. 1042–1068, June 2016. [Online]. Available: http://stanford.edu/~boyd/papers/scs.html