Learning Tensors From Partial Binary Measurements

Navid Ghadermarzy, Yaniv Plan, and Ozgur Yilmaz, Member, IEEE

Abstract—We generalize the 1-bit matrix completion problem to higher order tensors. Consider a rank-$r$ order-$d$ tensor $T$ in $\mathbb{R}^{N_1} \times \cdots \times \mathbb{R}^{N_d}$ with bounded entries. We show that when $r = O(1)$, such a tensor can be estimated efficiently from only $m = O(N(N_d-1))$ binary measurements. This shows that the sample complexity of recovering a low-rank tensor from 1-bit measurements of a subset of its entries is roughly the same as recovering it from unquantized measurements—a result that had been known only in the matrix case, i.e., when $d = 2$. By using a certain atomic M-norm as a convex proxy for rank, we allow for approximately low-rank tensors and give error bounds based on the M-norm of the tensor. We prove that the theoretical bounds are optimal both in the M-norm bound and the size $N$. Moreover, we show the advantage of directly using the low-rank tensor structure, rather than matricization, both theoretically and numerically. Specifically, we show how the 1-bit measurement model can be used for context-aware recommender systems.

Index Terms—Tensor completion, matrix completion, max norm, low-rank tensor, 1-bit tensor completion, context aware recommender systems.

I. INTRODUCTION

The problem of matrix completion, i.e., recovering a matrix from partial noisy measurements of a subset of its entries, arises in a wide variety of practical applications including collaborative filtering [1], sensor localization [2], [3], system identification [4] and seismic data interpolation [5]. The low-rank structure of a matrix makes it possible to complete it by sampling a small number of its entries (much smaller than its ambient dimension). Matrix completion is useful for applications where acquiring the full data is either expensive or impossible due to physical limitations.

Assuming that we have access to $m$ randomly selected entries of $M$, the set of indices of which is denoted by $\Omega$, the matrix completion problem entails finding a matrix $\hat{M}$ with smallest rank that agrees with the $m$ obtained samples. That is, defining $M_{\Omega}$ to be the projection of $M$ onto the set of matrices supported on $\Omega$,

$$\hat{M} := \arg \min_{M} \text{rank}(M) \text{ s.t. } M_{\Omega} = M_{\Omega}.$$

This problem has been studied extensively in the literature [6]–[9]. In general, rank-minimization is NP-hard. Therefore, one of the most common approaches is minimizing convex surrogates of the rank, such as nuclear-norm [8]–[10] and max-norm [11], [12]. Both these convex regularizers can be used to recover the underlying matrix with as few as $O(rN \log(N))$ measurements which is close to $O(Nr)$, the number of free variables in a rank-$r$ matrix in $\mathbb{R}^{N \times N}$.

A practical example where this problem arises naturally is the Netflix problem: Users and movies are arranged in the rows and columns of a matrix $M$, respectively. Each entry $M(i,j)$ represents the rating that user $i$ gives to movie $j$ [13]. In practice it is not possible to have all the ratings from all the users. This is where matrix completion becomes important, assuming that the true matrix is approximately low-rank. Furthermore, in most recommender systems like Netflix, the ratings are highly quantized, sometimes even to a single bit. For example, this happens when the users just answer if they liked a movie or not. This is where the problem of 1-bit matrix completion comes into play. Here the problem is recovering a low-rank matrix from binary (1-bit) measurements of a subset of its entries.

In many applications data can be represented as multidimensional arrays (tensors) [14]–[17]. Seismic images are usually 3 or 5 dimensional, 2 or 4 dimensions for sources and receivers locations and 1 dimension for time or frequency [18], [19]. In hyperspectral imaging, datacubes are indexed by two spatial variables and a frequency/wavelength variable [20]. Also in some applications it is useful to introduce new information as an extra dimension. For example adding the age of users as a dimension can improve the performance of recommender systems [21]. Similar to matrices, a common assumption is that the tensor is low rank, i.e., it can be written as the sum of a few rank-one tensors.

In this paper, we generalize the problem of 1-bit matrix completion to the case of tensors and analyze error bounds for recovering a low-rank tensor from partial 1-bit observations. Matricizing a low-rank tensor does not increase its rank but loses the (low-dimensional) structure of the dimensions that get merged to form the rows (or columns) of the matricized tensor [22]. Therefore, we work directly with the (approximately) low-rank tensor structure. We consider 1-bit tensor completion under M-norm (and max-qnorm) constraint, which is a robust proxy of tensor-rank [23]. In [23], generalizing the matrix max-norm to the case of tensors led us to define two functions which are closely related, i.e., max-qnorm and atomic M-norm. To be precise, direct generalization of matrix max-norm defines a quasi-norm (max-qnorm) which is non-convex (and thus not a norm). However, we can prove near-optimal sample complexity for tensor completion using max-qnorm constrained optimiza-
tion, which also gives promising results in practice. Moreover, in [23], studying the dual of the dual of this quasi-norm resulted in defining a convex norm, atomic M-norm, which can be used to prove similar near-optimal sample complexity. In this paper, we study constrained ML estimations using both these functions. The main contributions of the paper are summarized in the next subsection.

A. Contributions

To the best of our knowledge, this is the first paper that analyzes the sample complexity of 1-bit tensor completion without matricization. It is worth mentioning that the closest line of research to 1-bit tensor completion is the problem of context-aware recommender systems [21] and the work of [24], which studies collaborative filtering based on tensor factorization. However, in [24] the measurement model and the choice of regularization is different, and the sample complexity has not been analyzed.

Consider a rank-$r$, order-$d$ tensor $T^d \in \mathbb{R}^{N_1 \times \cdots \times N_d}$ where $N_i = O(N)$ for $1 \leq i \leq d$. The main contributions of the current paper is as follows.

1) We formulate and analyze the problem of 1-bit tensor completion. We show the advantage of working with a tensor directly without matricizing when $r \ll N$, both theoretically and numerically.

2) We analyze 1-bit tensor completion using M-norm constraints on the underlying tensor (this is a convex constraint). We prove that, with high probability, the mean squared error (MSE) of recovering a rank-$r$ tensor $T^d$ from $m$ 1-bit measurements by solving an M-norm constrained log-likelihood is $O(\sqrt{rdm} - \sqrt{N_d})$. Moreover, we analyze a related non-convex function, max-qnorm, and prove that MSE of optimizing a log-likelihood function constrained by max-qnorm is $O(\sqrt{rdm} - \sqrt{N_d})$.

3) The M-norm gives a convex proxy for the rank of a tensor (See Remark 1 in Section II-C). We derive an information-theoretic lower bound that proves the MSE of any arbitrary algorithm is $\Omega(R \sqrt{N/m})$ for a tensor with M-norm less than $R$. This proves that our upper bound is optimal in $N$ and the M-norm bound $R$ (but not necessarily in $r$).

4) We propose a numerical method to approximate the solution of max-qnorm constrained 1-bit tensor completion and show its advantage over 1-bit matrix completion using synthetic and real-world data. Specifically, we illustrate that one gets significant improvement by applying the 1-bit tensor completion to context-aware recommender systems.

B. Organization

In Section II we review some basic definitions related to tensors. Section III summarizes some related results in the literature. Section IV contains the main results regarding 1-bit tensor completion, regularized by max-qnorm and M-norm. In Section V we prove a lower bound on the best error bound achievable with any arbitrary algorithm. We explain an algorithm for approximating the solution of max-qnorm constrained tensor completion in VI and present numerical results on synthetic data. We present the results on some real data in Section VII. Finally in Section VIII we provide all the proofs of the theorems.

II. NOTATIONS AND BASICS ON Tensors

In this section, we introduce some basic definitions related to tensors. In particular, we introduce M-norm and max-qnorm [23] and briefly discuss some of their main properties.

A. Notations

We adopt the notations used in Kolda and Bader’s review on tensor decompositions [25]. In what follows all universal constants are denoted by $c$ or $C$. Lowercase Greek letters are used to denote scalars, e.g., $\alpha$, $\gamma$, and $\sigma$. Matrices are denoted by $M$ or $A$ and tensors are denoted by $T$ or $X$. Other uppercase letters can either denote a matrix or a tensor depending on the context. A $d$-dimensional tensor is an element of $\times_{i=1}^d \mathbb{R}^{N_i}$. The order of a tensor is the number of its dimensions (modes) and is usually denoted by $d$, and its length (size) is denoted by $(N_1, N_2, \ldots, N_d)$. When all the sizes are the same along all the dimensions, we denote $\mathbb{R}^{d}$ as $\mathbb{R}^{N}$. For a tensor $X \in \mathbb{R}_{\geq 0}^{d}$, elements of the tensor are alternately specified as either $X_{i_1,i_2,\ldots,i_d}$ or $X(i_1,i_2,\ldots,i_d)$, where $1 \leq i_j \leq N_j$ for $1 \leq j \leq d$. We also use $X_{\omega}$ with $\omega = (i_1,i_2,\ldots, i_d)$, to refer to $X(i_1,i_2,\ldots,i_d)$. Inner products are denoted by $\langle \cdot, \cdot \rangle$. The symbol $\circ$ represents both matrix and vector outer products where $T = U_1 \circ U_2 \circ \cdots \circ U_d$ means $T(i_1,i_2,\ldots,i_d) = \sum_{k} U_{1}(i_1,k)U_{2}(i_2,k)\cdots U_{d}(i_d,k)$, where $k$ ranges over the columns of the factors. In the special case when $u_j$’s are vectors, $T = u_1 \circ u_2 \circ \cdots \circ u_d$ means $T(i_1,i_2,\ldots,i_d) = u_{1}(i_1)u_{2}(i_2)\cdots u_{d}(i_d)$. Finally $[N] := \{1, \ldots, N\}$.

B. Rank of a Tensor

A unit tensor $U \in \times_{i=1}^d \mathbb{R}^{N_i}$ is a tensor that can be written as an outer product of $d$ vectors

$$U = u^{(1)} \circ u^{(2)} \circ \cdots \circ u^{(d)}$$

(1)

where $u^{(j)} \in \mathbb{R}^{N_j}$ is a unit-norm vector. The vectors $u^{(j)}$ are called the components of $U$. A rank-1 tensor is a non-zero scalar multiple of a unit tensor. The rank of a tensor $T$, denoted by rank($T$), is defined as the smallest number of rank-1 tensors that sum to $T$, i.e.,

$$\text{rank}(T) = \min_r \left\{ r : T = \sum_{i=1}^{r} \lambda_i U_i \right\}$$

$$= \sum_{i=1}^{r} \lambda_i u^{(1)}_i \circ u^{(2)}_i \circ \cdots \circ u^{(d)}_i$$

where $U_i \in \mathbb{R}^{N_j}$ is a unit tensor. This low-rank decomposition is also known as CANDECOMP/PARAFAC (CP) decomposition. For a tensor $T = \sum_{j=1}^{r} v^{(1)}_j \circ v^{(2)}_j \circ \cdots \circ v^{(d)}_j$ we define $V_j := [v^{(1)}_j, v^{(2)}_j, \ldots, v^{(d)}_j]$ to be the $j$-th factor matrix of $T$. The Frobenius
norm of a tensor is defined as
\[
\|T\|_F^2 := \langle T, T \rangle = \sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \cdots \sum_{i_d=1}^{N_d} T_{i_1, i_2, \ldots, i_d}^2.
\] (2)

C. M-Norm and Max-Qnorm

The max-qnorm of a tensor is defined as [23]
\[
\|T\|_{\max} := \min_{T = U^{(1)} \otimes \cdots \otimes U^{(d)}} \left\{ \sqrt{d} \|U(j)\|_2 \right\},
\] (3)
where \(\|U\|_2 := \sup_{\|x\|_2 = 1} \|Ux\|_\infty\) is the maximum row norm of \(U\). In [23], we thoroughly discussed this generalization and showed that it is a quasi-norm when \(d \geq 2\) (thus the name max-qnorm). Moreover, analyzing the dual of the dual of the max-qnorm led us to define a closely related atomic M-norm which is sufficient for estimating the true \(M\) of a bounded low-rank tensor from 1-bit measurements, they are robust proxies for the rank of a tensor and max-qnorm are independent of both M-norm and max-qnorm of a bounded low-rank tensor.

Remark 1: The atomic M-norm is indeed a norm and is closely related to the max-qnorm [23, Lemma 5]. The M-norm and max-qnorm are robust proxies for the rank of a tensor and both M-norm and max-qnorm of a bounded low-rank tensor is upper-bounded by a quantity that just depends on its rank and its infinity norm and is independent of \(N\). In particular, assume \(T \in \mathbb{R}^{N_1 \times \cdots \times N_d}\) is a rank-\(r\) tensor with \(\|T\|_\infty = \alpha\). Then [23, Th. 7]
\begin{itemize}
    \item \(\alpha \leq \|T\|_M \leq (r\sqrt{d})^{d-1} \alpha\).
    \item \(\alpha \leq \|T\|_{\max} \leq (r\sqrt{d})^{d-1} \alpha\).
\end{itemize}

III. PRIOR WORK

A. 1-Bit Matrix Completion

In the 1-bit matrix completion problem, the measurements are in the form of \(Y(\omega) = \text{sign}(M^T(\omega) + Z(\omega))\) for \(\omega \in \Omega\), where \(Z\) is a noise matrix. Notice that without the noise, there will be no difference in the 1-bit measurements of \(M^T\) and \(\alpha M^T\) for any positive \(\alpha\). However, if we assume that the additive noise comes from a log-concave distribution, the problem can be solved by minimizing a regularized negative log-likelihood function given the measurements. In other words, the noise matrix has a dithering effect which is essential for recovery of the matrix. Under these assumptions, a nuclear-norm constrained maximum likelihood (ML) optimization was used in [26] to recover \(M^T\) and it was proved that it is minimax rate-optimal under the uniform sampling model. In particular, in order to recover a rank-\(r\) matrix \(M^T\) with \(\|M^T\|_\infty \leq \alpha\) from 1-bit measurements, they considered the weaker assumption that the matrix belongs to the set \(\{A \mid \|A\|_* \leq \alpha \sqrt{rN^2}, \|A\|_\infty \leq \alpha\}\) and proved
\[
\frac{1}{N^2} \|M^T - M_{\text{recovered}}\|_F^2 \leq C_\alpha \sqrt{\frac{rN}{m}},
\] provided that \(m > CN\log(N)\). It was shown in [26] that these estimates are near-optimal.

A line of research followed [26] that concentrated on the recovery of low rank matrices from 1-bit measurements [27]–[29]. Here we emphasize [27] which considered using max-norm-constrained ML estimation instead of nuclear-norm-constrained ML. The max norm, an alternative convex proxy of the rank of a matrix, bounds the spectral norm of the rows of the low rank factors of \(M^T\), say \(U\) and \(V\). It is defined as \(\|M^T\|_{\max} := \min \|U\|_2 \|V\|_2\) s.t. \(M^T = UV^T\) [30], where \(\|U\|_2\) is the maximum row norm of the matrix \(U\). The max norm has been extensively investigated in the machine learning community after the work of [31] and shown to be empirically superior to nuclear-norm minization for collaborative filtering problems.

A max-norm constrained ML estimation was analyzed in [27] and it was shown to be near optimal. To be precise, it was proved in [27] that under some weak assumptions on the sampling distribution, with probability \(1 - \frac{1}{N}\),
\[
\frac{1}{N^2} \|M^T - M_{\text{recovered}}\|_F^2 \leq C_\alpha \left( \alpha \sqrt{\frac{rN}{m}} + U_\alpha \sqrt{\log(N)} \right).
\]
Notice that \(m = O(rN)\) is sufficient for estimating the true matrix efficiently.

B. 1-Bit Tensor Completion

The generalization of 1-bit matrix completion to 1-bit tensor completion is new. However, tensor factorization for context-aware recommender systems has been investigated in [24], [32], [33] with two distinct differences. First they lack the 1-bit machinery introduced in [26] which we show its importance in Section VII and second is the lack of theoretical analysis. It is worth mentioning that the area of tensor completion using noisy (not 1-bit) measurements is a very active field of research [34]–[41].

Another paper [42] considered the problem of 1-bit tensor completion and was published around the same time this manuscript was submitted. The main focus of [42] is on finding efficient ways of applying matrix completion to the problem of tensor completion. In this paper, on the other hand, we prove that the sample complexity can be significantly reduced by analyzing the 1-bit tensor completion problem directly (without matricization).

IV. MAIN RESULTS

Next, we explain our main results. After stating the observation model and our main goal in detail in Section IV-A, we provide recovery guarantees for 1-bit tensor completion using max-qnorm and M-norm constrained ML estimation in Section IV-B. The proofs are presented in Section VIII. Necessary tools such as Rademacher complexity and discrepancy, are provided in Appendices A and B.

A. Problem Formulation and Observation Model

Given a \(d\)-dimensional tensor \(T^T \in \bigotimes_{i=1}^d \mathbb{R}^{N_i}\) (for simplicity we assume \(N_i = N\) and a random subset of its indices \(\Omega \subset [N] \times [N] \times \cdots \times [N]\), suppose that we only observe 1-bit
measurements of $T^2$ on $\Omega$ according to the rule
\[
Y_\omega = \begin{cases} 
+1 & \text{with probability } f(T^2_\omega), \\
-1 & \text{with probability } 1 - f(T^2_\omega)
\end{cases} \quad \text{for } \omega \in \Omega. 
\]
(5)

Here $f : \mathbb{R} \to [0, 1]$ is a monotone differentiable function which can be interpreted as the cumulative distribution of some noise function $Z_\omega$ where the observation then becomes [26]
\[
Y_\omega = \begin{cases} 
+1 & \text{with probability } T_\omega + Z_\omega \geq 0, \\
-1 & \text{with probability } T_\omega + Z_\omega < 0
\end{cases} \quad \text{for } \omega \in \Omega. 
\]

A standard assumption is that observed indices are chosen uniformly at random, i.e., we randomly pick $|\Omega|$ indices out of all the $N^d$ indices and then take 1-bit measurements on this set. Here, $\Omega$ is chosen at random such that $\mathbb{P}(\omega \in \Omega) = \frac{m}{N^d}$. Alternatively, the measurements may be picked uniformly at random with replacement so each index is sampled independently of the other measurements. We take a generalization of the latter model, i.e., we assume a general sampling distribution $\Omega = \{\pi_\omega\}$ for $\omega \in [N] \times [N] \times \cdots \times [N]$ such that $\sum_\omega \pi_\omega = 1$ and then each index is sampled using the distribution $\Omega$. Although in this method it is possible to choose a single index multiple times, it has been proven in various contexts that sampling with replacement can be as powerful theoretically as sampling without replacement [12], [27]. Other than theoretical considerations, assuming non-uniform sampling is a better model for various applications of 1-bit matrix and tensor completion, including the Netflix problem. The challenges and benefits of using non-uniform sampling are thoroughly discussed in [27], [43].

We assume that $f$ is such that the following quantities are well defined:
\[
L_\alpha := \sup_{|x| \leq \alpha} \frac{|f'(x)|}{f(x)(1 - f(x))}, \\
\beta_\alpha := \sup_{|x| \leq \alpha} \frac{f(x)(1 - f(x))}{(f'(x))^2},
\]
where $\alpha$ is the upper bound of the absolute values of entries of $T$. Here, $L_\alpha$ controls the steepness of $f$, $\beta_\alpha$ controls its flatness. Furthermore, we assume $f$ and $f'$ are non-zero in $[-\alpha, \alpha]$ and the quantity:
\[
U_\alpha := \sup_{|x| \leq \alpha} \log \left( \frac{1}{f(x)(1 - f(x))} \right)
\]
is well-defined. A few well-known examples are [26]:
- Logistic regression/Logistic noise defined as $f(x) = \frac{e^x}{1 + e^x}$, $L_\alpha = 1$, $\beta_\alpha = \frac{(1 + e^\alpha)^2}{e^\alpha}$, and $U_\alpha = 2 \log(e^{\frac{\alpha}{\beta}} + e^{-\frac{\alpha}{\beta}})$.
- Probit regression/Gaussian noise defined as $f(x) = \Phi\left(\frac{x}{\sigma}\right)$, where $\Phi$ is the cumulative distribution function of a standard Gaussian random variable $L_\alpha \leq \frac{2}{\sqrt{\pi}}(\frac{\alpha}{\sigma} + 1)$, $\beta_\alpha \leq \pi \sigma^2 \exp\left(\frac{\alpha^2}{2\sigma^2}\right)$, and $U_\alpha \leq \left(\frac{2}{\sigma} + 1\right)^2$.

The problem of 1-bit tensor completion entails recovering $T^2$ from the measurements $Y_\omega$ given the function $f$. To do this, similar to [26], we minimize the negative log-likelihood function
\[
\mathcal{L}_{\Omega,Y}(X) = \sum_{\omega \in \Omega} \left( \mathbb{I}_{Y_\omega = +1} \log \left( \frac{1}{f(X_\omega)} \right) + \mathbb{I}_{Y_\omega = -1} \log \left( \frac{1}{1 - f(X_\omega)} \right) \right),
\]
given our observation and subject to $X$ having low max-q-norm or M-norm and small infinity-norm. Defining
\[
K^T_{\Omega,X}(\alpha, R) := \{T \in \mathbb{R}^{N_1 \times \cdots \times N_d} : ||T||_\infty \leq \alpha, ||T||_{\max} \leq R\}, \\
K^T_M(\alpha, R) := \{T \in \mathbb{R}^{N_1 \times \cdots \times N_d} : ||T||_\infty \leq \alpha, ||T||_{M} \leq R\},
\]
we analyze the following optimization problems:
\[
\hat{T}_{\Omega} = \arg \min_X \mathcal{L}_{\Omega,Y}(X) \quad \text{s.t. } X \in K^T_{\Omega,X}(\alpha, R_{max}). \\
\hat{T}_M = \arg \min_X \mathcal{L}_{\Omega,Y}(X) \quad \text{s.t. } X \in K^T_M(\alpha, R_{M}).
\]

Remark 2: Flat low-rank tensors are contained in $K^T_{\Omega,X}(\alpha, R_{\Omega, max})$ and $K^T_M(\alpha, R_M)$. In particular, for a rank-$r$ tensor $T^r$ with $||T^r||_\infty \leq \alpha$, it is sufficient to choose $R_{\max} = r^{\frac{\alpha}{dN}} \alpha$ in (8) and $R_M = r^{\frac{\alpha}{dN}} \alpha$ in (9) for $T^2$ to be feasible [23]. M-norm, max-q-norm and infinity norm are all continuous functions and therefore, both $K^T_{\Omega,X}(\alpha, R_{\Omega, max})$ and $K^T_M(\alpha, R_M)$ contain approximately low-rank tensors. We assume the data is generated from one of these sets of approximate low-rank tensors. Considering nearly low rank structure is a common approach is the literature [26], [27] which is the best model for many real applications where the underlying tensor might be close to a low-rank tensor rather than an exactly low rank tensor.

B. Max-Qnorm and M-Norm Constrained 1-Bit Tensor Completion

In this section we analyze the problem of 1-bit tensor recovery using max-q-norm constrained ML estimation as well as M-norm constrained ML estimation. These two optimization problems are closely related, and their analysis and results are similar. We explain the corresponding results together.

Theorem 3: Suppose that we have $m$ 1-bit measurements of a subset of the entries of a tensor $T^2 \in \mathbb{R}^{N^d}$ following the probability distribution explained above where $f$ is chosen such that $L_\alpha$, $\beta_\alpha$, and $U_\alpha$ are well defined. Moreover, the indices of the tensor are sampled according to a probability distribution $\Omega$. Then assuming $||T^2||_{\max} \leq R_{\max}$, there exist absolute constants $C_{\max}$ and $C_M$ such that for a sample size $2 \leq m \leq N^d$ and for any $\delta > 0$ the maximizer $\hat{T}_{\Omega, \max}$ of (8) satisfies:
\[
||T^2 - \hat{T}_{\Omega, \max}||_1^2 := \sum_{\omega} \pi_\omega (\hat{T}_{\Omega, \max}(\omega) - T^2(\omega))^2 \\
\leq C_{\max} c_2^d \beta_\alpha \left( L_\alpha R_{max} \sqrt{\frac{dN}{m} + U_\alpha} \frac{\log(\frac{m}{\delta})}{m} \right)
\]
(10)
with probability at least \(1 - \delta\). Similarly, assuming \(\|T^\delta\|_M \leq R_M\) for any \(\delta > 0\), the maximizer \(\hat{T}_M\) of (9) satisfies:

\[
\|T^\delta - \hat{T}_M\|_1^2 = \sum_\omega \pi_\omega (\hat{T}_M(\omega) - T^\delta(\omega))^2
\]

\[
\leq C_M \beta_\alpha \left\{ L_\alpha R_M \left[ \frac{dN}{m} + U_\alpha \sqrt{\frac{\log(\frac{1}{\delta})}{m}} \right] \right\}
\]

(11)

with probability at least \(1 - \delta\).

Corollary 4: A rank-\(r\) tensor \(T^\delta\) with \(\|T^\delta\|_\infty \leq \alpha\) satisfies \(\|T^\delta\|_{\text{max}} \leq \sqrt{r^{d-1} \alpha}\) and therefore, the maximizer \(\hat{T}_M\) of (8) with \(R_{\text{max}} = \sqrt{r^{d-1} \alpha}\) satisfies

\[
\|T^\delta - \hat{T}_M\|_1^2 \leq C_M \beta_\alpha \left\{ L_\alpha \sqrt{r^{d-1} \alpha} \left[ \frac{dN}{m} + U_\alpha \sqrt{\frac{\log(\frac{1}{\delta})}{m}} \right] \right\},
\]

with probability greater than \(1 - \delta\). Moreover, \(\|T^\delta\|_M \leq (r \sqrt{r^{d-1} \alpha})\) and the maximizer \(\hat{T}_M\) of (9) with \(R_M = (r \sqrt{r^{d-1} \alpha})\) satisfies

\[
\|T^\delta - \hat{T}_M\|_1^2 \leq C_M \beta_\alpha \left\{ L_\alpha (r \sqrt{r^{d-1} \alpha}) \left[ \frac{dN}{m} + U_\alpha \sqrt{\frac{\log(\frac{1}{\delta})}{m}} \right] \right\},
\]

with probability greater than \(1 - \delta\).

Remark 5: There are two differences in the right hand sides of (10) and (11). First is the difference in \(R_{\text{max}}\) and \(R_M\) which is due to the different upper bounds on max-\(q\)-norm and M-norm of a bounded rank-\(r\) tensor. The second difference is in the constants \(C_{\text{max}}\) and \(C_M\) where \(C_M < \frac{C_{\text{max}}}{\sqrt{r^{d-1} \alpha}}\) and \(C_M < \frac{C_{\text{max}}}{\sqrt{r^{d-1} \alpha}}\). The key difference here is that the set of low M-norm tensors gives a convex proxy for the set of low-rank tensors – it contains the set of low-rank tensors, but since it is a convex set, it is robust. That is, it also contains all tensors that are close to low-rank tensors. To be precise, fixing the scale, the set of approximately low rank tensors (low M-norm tensors) is larger than the set of exactly low-rank tensors and thus has a larger tight error bound. We see the same behavior when \(d = 2\) as well [26], [27], [47]. Moreover, we show that the square root is not removable in Theorem 13. An interesting study of the square root issue for empirical risk minimization with a smooth loss function is summarized in [48, Table 1]. The distinction between parametric (finite dimensional classes) vs Scale-sensitive measures of complexity, e.g., the difference between considering exact-rank constraints and M-norm constraint is the main reason behind the square root in the error bound.

V. INFORMATION THEORETIC LOWER BOUND

We use a classical information-theoretic technique that shows that with limited amount of information we can distinguish only a limited number of tensors from each other. To that end, as in [26], [27], we first construct a set of tensors \(\chi\) such that for any two distinct members \(X^i\) and \(X^j\) of \(\chi\), \(\|X^i - X^j\|_1\) is large. Therefore, we should be able to distinguish the tensors in this set if the recovery error of an arbitrary algorithm is small enough. However, Fano’s inequality will imply that the probability of choosing the right tensor among this set is going to be small and therefore force a lower bound on the performance of any arbitrary recovery algorithm.

The lower bound is achieved using [23, Lemma 18], which constructs a packing set for the set of low M-norm tensors given by

\[
K^T_M(\alpha, R_M) := \{ T \in \mathbb{R}^{N_1 \times \cdots \times N_d} : \|T\|_1 \leq \alpha, \|T\|_M \leq R_M \}.
\]

where

\[
m = O(r^{3d-3} N)\) using (9). When \(r \ll N\), these are significant improvements over matricizing, which would require \(\Omega(rN^{\frac{d}{2}})\) measurements.

Remark 9: 1-bit tensor completion can recover the magnitude of the tensor because of the dithering effect of the noise we add before sampling. If the SNR is too low, we might end up estimating the noise by a low rank tensor and if the SNR is too high, we risk losing magnitude information. In Section VI, we conduct some experiments and discuss the optimal noise level using synthetic data.

Remark 10: 1-bit matrix completion using max-norm is analyzed in [27]. A direct generalization resulted in the error bound (10) which recovers their result when \(d = 2\). However, the direct generalization of max-norm to tensors is a quasi-norm and the resulting constraint is non-convex. The closely related upper bound using M-norm (11) is governed by a convex optimization and gives similar results when \(d = 2\). In this case the only difference is the Grothendieck’s constant \(1.67 < c_1 c_2^3 < 1.79\) [45].

Remark 11: Some results in the matrix completion literature (non 1-bit) provide error bounds of the form \(\frac{\sqrt{d}}{\sqrt{m}}\) instead of \(\frac{\sqrt{d}}{\sqrt{m}}\) [10], [46]. The key difference here is that the set of low M-norm tensors gives a convex proxy for the set of low-rank tensors – it contains the set of low-rank tensors, but since it is a convex set, it is robust. That is, it also contains all tensors that are close to low-rank tensors. To be precise, fixing the scale, the set of approximately low rank tensors (low M-norm tensors) is larger than the set of exactly low-rank tensors and thus has a larger tight error bound. We see the same behavior when \(d = 2\) as well [26], [27], [47]. Moreover, we show that the square root is not removable in Theorem 13. An interesting study of the square root issue for empirical risk minimization with a smooth loss function is summarized in [48, Table 1]. The distinction between parametric (finite dimensional classes) vs Scale-sensitive measures of complexity, e.g., the difference between considering exact-rank constraints and M-norm constraint is the main reason behind the square root in the error bound.
We include [23, Lemma 18] and an adoption of the proof in [26] in Section VIII-B. In the following $1.68 < K_2 := c_1c_2 < 1.79$ is the Grothendieck’s constant.

**Theorem 12:** Fix $\alpha, R_M$, and $N$ and set $r := \lceil \left( \frac{R_M}{K_2} \right)^2 \rceil$ such that $\alpha \geq 1, \alpha^2rN \geq C_0, r \geq 4$ and $r \leq O(\frac{N}{\alpha^2})$. Let $\beta_n$ be defined as in (6). Suppose $\hat{f}(x)$ is decreasing for $x > 0$ and for any tensor $T$ in the set $K_{\alpha}^T(\alpha, R_M)$, assume we obtained $m$ measurements (denoted by $Y_{i0}$) from a random subset $\Omega$ of the entries following the model (5). Consider any arbitrary algorithm that takes these measurements and returns an approximation $\hat{T}_M(Y_{i0})$. Then there exist a tensor $T \in K_{\alpha}^T(\alpha, R_M)$ such that

$$\mathbb{P}\left[ \frac{\|T-\hat{T}_M(Y_{i0})\|_F^2}{N^d} \geq \min \left( C_1, C_2 \sqrt{\frac{\beta_n}{\alpha^2} \frac{R_M}{K_2} \sqrt{\frac{N}{m}}} \right) \right] \geq \frac{3}{4}$$

**Remark 13:** A similar lower bound can be obtained for the max-qnorm constrained algorithm as well. Consider the same conditions as in Theorem 12 except for setting $r := \lceil \left( \frac{R_{max}}{\alpha} \right)^2 \rceil$ and assuming $R_{max} \geq \frac{\sqrt{2}}{2} \alpha^d$. For any tensor $T$ in the set $K_{\alpha}^T(\alpha, R_{max})$, assume we obtained $m$ measurements (denoted by $Y_{i0}$) from a random subset $\Omega$ of the entries following the model (5). Consider any arbitrary algorithm that takes these measurements and returns an approximation $\hat{T}_{max}(Y_{i0})$. Then there exist a tensor $T \in K_{\alpha}^T(\alpha, R_{max})$ such that

$$\mathbb{P}\left[ \frac{\|T-\hat{T}_{max}(Y_{i0})\|_F^2}{N^d} \geq \min \left( C_1, C_2 \sqrt{\frac{\beta_n}{\alpha^2} R_{max} \sqrt{\frac{N}{m}}} \right) \right] \geq \frac{3}{4}$$

**Remark 14:** Theorems 3 and 12 prove that the sample complexity achieved by (9) is optimal in both $N$ and $R_M$. Evidence in the matrix case suggests that the recovery lower bound for the set of low rank tensors (instead of low M-norm) should be bounded by $\frac{\alpha}{m}$ instead of $\sqrt{\frac{\alpha}{m}}$, provided that $m$ is large enough. We postpone a lower bound for exact low-rank tensors to future work.

**VI. NUMERICAL RESULTS**

In this section, we present some numerical results of using (8) for 1-bit tensor completion. The optimization problem we want to solve is in the form of

$$\min_X \mathcal{L}_{\Omega, Y}(X) \quad \text{s.t.} \quad X \in K_{max}^T(\alpha, R_{max}),$$

where $\mathcal{L}_{\Omega, Y}(X)$ is the negative log-likelihood function defined in (7). Similar to the matrix-completion problem $\mathcal{L}_{\Omega, Y}$ is a smooth function from $\mathbb{R}^{N^d} \rightarrow \mathbb{R}$. First, we mention a few practical considerations.

**Max-qnorm Constrained 1-bit Tensor Completion**

We are not aware of any algorithm that is able to (approximately) solve the M-norm constrained problem (9). Therefore, instead, we focus on algorithms to (approximately) solve the max-qnorm constrained 1-bit tensor completion problem. To this end, we employ a similar approach to the one in [23] and use the low rank factors to estimate the max-qnorm of a tensor. In particular, defining $f(V_1, \ldots, V_d) := \mathcal{L}_{\Omega, Y}(V_1 \circ \cdots \circ V_d)$ the problem becomes

$$\min_{V_1, \ldots, V_d} f(V_1, \ldots, V_d)$$

$$\text{s.t.} \quad \|V_i\|_2,\infty \leq \frac{R_{max}^2}{\alpha^2}$$

As explained in details in [23], in the definition of max-qnorm (3), there is no limit on the size of the low rank factors. Due to computational restrictions we limit the factor size by twice the dimension size, i.e., $V_i \in \mathbb{R}^{N \times 2N}$. Although we end up approximating the solution of (13), our numerical results shows that, for synthetic tensors with rank smaller than $N$, $2N$ is a large enough factor size since results do not improve significantly when larger factors are used, at least in our experiments where $N < 100$. Another practical benefit of (13) is that we can use alternating minimization on the small factors instead of the whole tensor, which is crucial in applications where storing the whole tensor is too expensive.

The next practical problem is in choosing $R_{max}$. Although theory suggests an upperbound of $\frac{\sqrt{2\alpha}}{\alpha}$ [23], in practice this bound can be much higher than the actual max-qnorm of a rank-$r$ tensor. Moreover, in many practical applications, we do not have an accurate upper bound on the rank of the tensor (or the rank of a good approximation of a tensor). Therefore, we use cross validation to estimate the optimal value of $R_{max}$.

In summary, to approximate the solution of (13), in our experiments we use 10% of the available data for cross validating the choice of $R_{max}$ and optimize over the factors $V_i$ to $V_d$ alternatingly while at each iteration, defining $f_i(V_i) := f(V_1, \ldots, V_{i-1}, V_{i+1}, \ldots, V_d)$, we solve the simpler sub-problem

$$\min_{V_i} f_i(V_i) \quad \text{s.t.} \quad \|V_i\|_{2,\infty} \leq \frac{R_{max}^2}{\alpha^2},$$

while fixing all the other factors $V_j, j \neq i$. We solve each sub-problem (14) by employing a projected gradient algorithm. The algorithm updates all the factors

$$[V_i] \leftarrow \mathbb{P}_C([V_i - \gamma \nabla f_i]).$$

where $\mathbb{P}_C$ simply projects the factor onto the set of matrices with $\ell_2,\infty$ norm less than $R_{max}$. This projection looks at each row of the matrix and if the norm of a row is bigger than $R_{max}$, it scales that row back down to $R_{max}$ and leaves other rows unchanged.

For all the results presented in this paper we initialize the factors $V_1$ through $V_d$ to have random Gaussian entries and solve (14) by using the limited-memory projected quasi-newton algorithm implemented in [49]. This uses limited memory BFGS and uses a Barzilai-Borwein scaling of the gradient with a nonmonotone Armijo line search along the feasible direction to find the next iterate in the SPG step.

**Remark 15:** The constraint on the boundedness of the tensor $\|V_1 \circ \cdots \circ V_d\|_{\infty} \leq \alpha$ can also be incorporated into (14), which introduces new challenges to the algorithm. First, the exact projection onto the set of $\{V_i \|V_i\|_{2,\infty} \leq \alpha\}$ is not as straightforward as the projection onto $\{V_i \|V_i\|_{2,\infty} \leq \frac{R_{max}^2}{\alpha^2}\}$.
An approximate projection by rescaling the factor via

\[ V_i = \frac{\alpha}{\| V_1 \circ \cdots \circ V_d \|_\infty} V_i, \text{ if } \| V_1 \circ \cdots \circ V_d \|_\infty > \alpha \]

was introduced in [27]. On the other hand, the exact projection can be formulated as a quadratic linear program, which is very computationally expensive. The second complication is that adding this constraint makes the constraint set \( C \) the intersection of two convex sets, which again makes exact projection expensive.

In our synthetic experiments, we noticed that adding this constraint does not change the results significantly, especially for low-rank tensors. Furthermore, in our applications, we concentrate on the performance of the algorithm in recovering the sign of the tensor which reduces the importance of projecting onto the set of bounded tensors and therefore, in this section we just report the results obtained by (approximately) solving (14).

Remark 16: The optimization problem (14) can be solved in parallel over the rows of \( V_i \) as both the objective function and the regularizer \( (\| \cdot \|_2, \infty) \) are decomposable in the rows.

A. Synthetic Experiments

In this section we present extensive numerical results on synthetic experiments to compare the performance of 1-bit tensor completion using max-qnorm with matricization using nuclear norm [26]. An important component of the measurements which affects the results is the choice of the function \( f \). We investigate the optimal choice of \( \sigma \) when we use the Gaussian noise, i.e., \( f(x) = \Phi(x) \) in the log-likelihood function (7). Fig. 1 shows the results for \( d = 3, 4 \), where \( N = 30, 15 \) respectively. Here, we average the recovery error over 20 realizations when \( r = 5 \) and \( \sigma \) varies from 0.001 to 10. Moreover, in all cases, \( m = |\Omega| = \frac{N^d}{2} \).

In Fig. 1, we show the results for matricized nuclear-norm constrained 1-bit matrix completion (MC-Nuclear) and max-qnorm as explained above. The underlying low-rank tensor is generated from low rank \( N \times r \) factors with entries drawn i.i.d. from a uniform distribution on \([-1, 1]\). The tensor is then scaled to have \( \| T \|_\infty = 1 \). The figure shows the relative squared error

\[ RSE := \frac{\| T_{\text{recovered}} - T \|_F^2}{\| T \|_F^2} \]

against different values of \( m \). The MC-Nuclear results show the case where we matricize the tensor first and then solve a 1-bit matrix completion problem using the algorithm in [26], but with added cross validation for the optimal nuclear-norm bound. Similar to the matrix case [26], the results deteriorate in the high-noise regime as we end up measuring the noise, which in the 1-bit regime looks like a coin toss. Choosing small noise on the other limits the dithering effect which is essential for successful recovery. Notice that the same experiment can be done with various values of \( r \) and \( m \). However, finding the optimal value of \( \sigma \) for different cases is time consuming and unnecessary as for most cases \( \sigma = 0.1 \) seems to be close to the optimal value when \( \| T \|_\infty = 1 \).

Once we fix the choice of \( \sigma \) we move on to comparing results for different ranks and tensor sizes. We present results for various rank \( r \in \{3, 5, 10\} \) and various sample sizes \( \frac{m}{N^d} \in \{0.1, 0.2, \ldots, 1\} \). Fig. 2 shows the results for 3-dimensional tensors \( T \in \mathbb{R}^{30 \times 30 \times 30} \). A balanced matricization is not possible in this case and therefore, tensor completion has an added advantage when the tensor has an odd order. We can see this advantage clearly in Fig. 2. When the number of observations is small, e.g., when \( \frac{m}{N^d} = 0.3 \) the average error of 1-bit tensor completion is four to six times better than matricizing. Note that in both matricization and max-qnorm constrained recovery we use 10% of the observations to validate the choice of the regularization parameter. Fig. 3 shows the results for \( T \in \mathbb{R}^{15 \times 15 \times 15} \). Again, we see a significant improvement over matricization.

VII. Applications

In this section we test the performance of (13) on real-world data, in particular, for predicting exam scores and user ratings. We show the improvements gained by solving 1-bit tensor completion instead of matricizing. We transform the ratings (or
scores) to 1-bit data by replacing them with whether they are above or below some approximate mean-rating. Next we investigate the ability of 1-bit tensor completion for predicting if an observed rating is above or below average. In all the applications we know the maximum value of the ratings and use this value to rescale the data to have unit infinity norm. This helps us in choosing the appropriate function \( f \) based on our synthetic experiments in Section VI-A, i.e., Gaussian noise with \( \sigma = 0.1 \). In Remark 17 we explained the consequences of this choice in more details.

To be more precise, we explain the general application setup we use in this section briefly. Assume \( T^s \) is the true tensor and we have access to a subset of its entries for training (\( T^s_{\Omega_{\text{train}}} \)) and another subset of the entries for testing (\( T^s_{\Omega_{\text{test}}} \)). \( \Omega_{\text{train}} \) and \( \Omega_{\text{test}} \) are two disjoint subsets of the entries. Below is a brief recipe of the experiments done in this section.

1) Scale the observed entries to have unit infinity norm.
2) Using an approximate mean-value \( \eta \), take \( \text{sign}(T^s_{\Omega_{\text{train}}} - \eta) \) to be the 1-bit measurements. Notice that with synthetic low rank tensors we add a dithering noise using some function \( f \) before taking the sign (see (5)). However, our experiments showed that assuming the noise to be intrinsic in the data gives slightly better results than adding. The same approach was taken in [26].
3) Use (13) to get an initial estimate \( \hat{T}_{\text{init}} \).
4) Add the approximate mean-value \( \eta \) and scale the resulting tensor back to get the final \( \hat{T} \), which is an approximation of \( T^s \).
5) Evaluate \( \hat{T} \) by comparing \( \hat{T} \) and \( T^s \) on the test indices.

**Remark 17:** In the applications, we empirically observe that we get better predictions if we do not dither the original tensor, i.e., if we assume the noise to be implicit in the data. Moreover, as explained in Remark 15, in the applications section we also ignore the infinity-norm constraint. Therefore, in theory changing the value of \( \sigma \) should not change the results of predicting the sign of the tensor. For example, if we consider recovering the original tensor with two noise functions \( f_1(x) = \Phi(\frac{x}{\sigma_1}) \) and \( f_2(x) = \Phi(\frac{x}{\sigma_2}) \), we would have

\[
\Phi\left(\frac{T(\omega)}{\sigma_1}\right) = \Phi\left(\frac{\frac{\sigma_1}{\sigma_2}T(\omega)}{\sigma_2}\right),
\]

which shows that the recovered tensors achieved by \( f_1 \) and \( f_2 \) should be scalar multiples of each other and the sign predictions should be the same. In Fig. 4 we illustrate this by showing the results of applying noise functions with different values of \( \sigma \) for recovering a rank-5 tensor in \( \mathbb{R}^{30 \times 30 \times 30} \) whose 1-bit measurement have been obtained with \( \sigma = 0.15 \). The plot in the left shows the average RSE and as expected the best recovery of the original tensor is obtained when we use the true noise function that we originally used to get the observations. However, the right plot shows that the percentage of correct sign predictions is very robust to the choice of \( \sigma \) and all the results are very close to each other which supports the above discussion.

In this section, we apply our algorithm to the data on pupil attainments in schools in Scotland which contains the information of 3,435 children who attended 148 primary schools and 19 secondary schools in Scotland.\(^1\) We generate a 5-dimensional tensor in \( \mathbb{R}^{148 \times 19 \times 2 \times 4 \times 10} \) which includes the information of primary school, secondary school, gender, social class and attainment score of the students and estimate the verbal reasoning score of the students based on varying number of 1-bit information (from 230 samples to 2100 samples which is less than 1% of the total entries of the tensor). The scores are ranged from \(-40 \) to \(40 \) and we take the 1-bit information to be the sign of the score. We use 10% of the observations for cross validating the choice of parameters and another 10% of the total scores as a test set. Fig. 5(a) shows the percentage of correct sign predictions and Fig. 5(b) shows the mean absolute error

\[
\text{MAE} := \sum_{\omega \in \Omega_{\text{test}}} |T^s(\omega) - \hat{T}(\omega)| / |\Omega_{\text{test}}|
\]

on the test set. Notice that the scores are in the range of \(-40 \) to \(40 \) and the Mean absolute error is in the range of \(8 \) to \(10 \). The matrix completion results refer to matricizing the tensor to a 592 \( \times \) 380 matrix by putting the first and fourth dimension in the rows and the rest in the columns. This matricization results in the most balanced rearrangement which is the recommended way for better results [22]. In both figures we can see that using tensor completion outperforms matrix completion significantly.

\(^1\)Available at http://www.bristol.ac.uk/cmm/learning/mmssoftware/data-rev.html
TABLE I
RESULTS OF A COMPARISON BETWEEN 1-BIT AND MULTI-BIT MATRIX AND TENSOR COMPLETION ALGORITHMS ON IN-CAR-MUSIC DATA [51] FOR PREDICTING WHETHER THE UNOBSERVED RATINGS WERE ABOVE OR BELOW AVERAGE. THE MULTI-BIT METHODS USES THE ORIGINAL RATINGS FROM 0 TO 5 AND THE CONTEXT-AWARE METHODS INCLUDE THE CONTEXT INFORMATION SUCH AS TIME OF THE DAY, WEATHER, LOCATION OF DRIVING AND MOOD.

| Original rating | 1-bit matrix completion | multi-bit matrix completion | 1-bit TC (context-aware) | multi-bit TC (context-aware) | Overall |
|-----------------|------------------------|-----------------------------|--------------------------|-----------------------------|---------|
| 1               | 75%                    | 65%                         | 54%                      | 51%                         | 70%     |
| 2               | 75%                    | 66%                         | 54%                      | 51%                         | 70%     |
| 3               | 75%                    | 65%                         | 54%                      | 51%                         | 70%     |
| 4               | 75%                    | 66%                         | 54%                      | 51%                         | 70%     |
| 5               | 75%                    | 65%                         | 54%                      | 51%                         | 70%     |

TABLE II
RESULTS OF A COMPARISON BETWEEN 1-BIT AND MULTI-BIT MATRIX TENSOR COMPLETION ALGORITHMS ON TIJUANA RESTAURANT DATA [52] FOR PREDICTING WHETHER THE UNOBSERVED RATINGS WERE ABOVE OR BELOW AVERAGE. TF REFERS TO USING TENSOR FACTORIZATION WITHOUT THE MAX-QNORM REGULARIZATION. FOR THE 1-BIT RESULTS WE USE \( f(x) = \Phi(\hat{x}) \).

| Original rating | 1-bit matrix completion | multi-bit matrix completion | 1-bit TC (max-qnorm, \( \sigma = 0.1 \)) | multi-bit TC (max-qnorm, \( \sigma = 0.5 \)) | Overall |
|-----------------|------------------------|-----------------------------|--------------------------|-----------------------------|---------|
| 1               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 2               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 3               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 4               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 5               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |

TABLE III
RESULTS OF A COMPARISON BETWEEN 1-BIT AND MULTI-BIT MATRIX TENSOR COMPLETION ALGORITHMS ON TIJUANA RESTAURANT DATA [52] SHOWING THE MEAN ABSOLUTE ERROR.

| Original rating | 1-bit matrix completion | multi-bit matrix completion | 1-bit TC (max-qnorm, \( \sigma = 0.1 \)) | multi-bit TC (max-qnorm, \( \sigma = 0.5 \)) | Overall |
|-----------------|------------------------|-----------------------------|--------------------------|-----------------------------|---------|
| 1               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 2               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 3               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 4               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |
| 5               | 75%                    | 65%                         | 70%                      | 75%                         | 74%     |

B. In-Car Music Recommender System

In this section we apply our algorithm to an in-car music recommender system [51] which contains 4000 ratings from 42 users and 140 music tracks. The ratings are acquired via a mobile application and each rating is accompanied with one of the 26 context information which ranges from the landscape to the weather condition or the driver’s mood. The resulting tensor is of size 42 \( \times \) 140 \( \times \) 26 and we use 3200 ratings for the minimization (less than 2% of the entries of the tensor), 400 ratings for validating the choice of max-qnorn bound and the other 400 ratings as test data. Table I shows the results using tensor completion while considering the context information and matrix completion. Both matrix and tensor completion results are obtained by the max-qnorn constrained tensor completion algorithm explained in [23]. The table shows the correct 1-bit predictions made by each algorithm considering their original ratings in the first six columns and the total average in the last row. Bringing context into the reconstruction results in an impressive improvement of at least 17% and moreover, using 1-bit information does not change the results too much compared to using the full information. Note that the results are averaged over 10 different random training and test sets.

Remark 8: In [26], a similar experiment was done with movie ratings which showed a significant improvement for using 1-bit data instead of original (multi-bit) data. However, here we just see a small improvement. We believe this is due to the very careful way the in-car music data was gathered. To be precise 1-bit matrix (and tensor) completion seems to be working better when there is an implicit noise in the ratings given by the users which can’t be accounted for when we fit the data exactly.

Remark 9: It is generally true that the prediction should be more accurate when the original rating is further away from the average. This trend can still be seen in Table I as well except for one case which is due to the very few instances of 0-rating in the test data (generally 3 to 5 ratings).

C. A Restaurant Context-Aware Recommender System

Next, we apply our algorithm to a restaurant recommender system [52]. The data was gathered by asking 50 users who went to 40 restaurants in the city of Tijuana and rated their experience through a Web based platform. The contextual information added is the day of the week (midweek and weekend) and the place (school, home and work). The resulting tensor is in \( \mathbb{R}^{40 \times 50 \times 6} \) and the ratings are ranged from 1 to 5 with an average of 3.40. We take the 1-bit information to be whether or not a rating is higher than the average or not. In Table II we report the percentage of correct above-or-below-average predictions for different ratings in the test set which we will refer to as 1-bit predictions while we have access to 1-bit information of 1000 entries (around 9% of the indices). To be precise, we show what percentage of the test ratings have been correctly predicted to be above or below average. In Table III we show the mean absolute error of the recovered tensor on the test set, i.e., taking \( \Omega \) to be the test set, we show \( MAE := \frac{\sum_{\omega \in \Omega} | T(\omega) - \hat{T}(\omega) |}{| \Omega |} \).

For this restaurant-ratings data set, a naive matricization, lead- to using the full information. Note that the results are averaged over 10 different random training and test sets.

1) The first question is the importance of using max-qnorn regularizer. In the first two rows of Tables II and III we show the results without using max-qnorn and just using tensor factorization with \( r = 10 \), i.e., by doing alternating minimization over \( N_i \times 10 \) factors (\( r = 10 \) is the best rank found by numerous empirical experiments). Both results of 1-bit prediction and the mean absolute error is generally worse than using max-qnorn. The difference is significant in the 1-bit predictions. It is worth mentioning though that the tensor factorization is a non-convex problem and a more complex algorithm might get better results but it is not the focus of this paper.

2) The second question we investigate is the effect of choosing \( f(x) \) in (7) on the results. The 4th and 5th row of Tables II and III shows the results for two different values of \( \sigma \). In short, smaller noise (not too small though) result...
in better 1-bit predictions and larger noise (not too large though) results in better prediction of the actual score. 
This behavior can be explained by examining the likelihood function III more closely. When \( \sigma \) is small the dithering function \( f \) is spiky around \( X(\omega) = 0 \) and therefore it is more sensitive to whether or not \( X(\omega) \) is positive or negative rather not how large it is. Therefore, using larger values of \( \sigma \) does a better job in recovering the original rating but has less sensitivity to the sign of \( X(\omega) \). This can be seen in Tables II and III where we recover the sign of the ratings better when we use \( \sigma = 0.1 \) but do worse in terms of mean absolute error. This is more evident in Table III where using smaller \( \sigma \) does a better job in recovering the ratings that are close to the average and worse in the ratings that are further from the average.

Remark 17 investigates these differences in more details.

As expected the best results for recovering the original ratings is obtained by exact tensor completion. However, notice that 1-bit TC outperform exact tensor completion for ratings that are around the mean significantly and struggles for correct scale of the ratings that are far away from the mean.

Remark 20: Although in theory the sign predictions should be the same with \( \sigma = 0.1 \) and \( \sigma = 0.5 \), we see a slight difference in the sign predictions in Table II. We believe this difference is a result of the algorithm not converging to the global optimum.

VIII. PROOFS

A. Proof of Theorem 3

Our proof closely follows the one in [27, Sec. 7.1]. Therefore, we just briefly explain the steps. In what follows we prove the max-qnorm constrained ML estimation error. The proof for M-norm constraint one \((11)\) follows the exact same steps where the only difference is a constant difference in the Rademacher complexity of the unit balls of these two norms. Define the loss function \( g(x; y) : \mathbb{R} \times \{ \pm 1 \} \to \mathbb{R} \) as:

\[
g(x; y) = \mathbb{I}_{[y=1]} \log \left( \frac{1}{f(x)} \right) + \mathbb{I}_{[y=-1]} \log \left( \frac{1}{1-f(x)} \right)
\]

Regarding the tensor completion problem as a prediction problem where we consider the tensor \( T \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_d} \) as a function from \([N_1] \times [N_2] \times \cdots \times [N_d] \rightarrow \mathbb{R} \) where the value of the function at \((i_1, i_2, \ldots, i_d)\) is the corresponding entry, the proof is based on a general excess risk bound developed in [53]. For a subset \( S = \{ \omega_1, \omega_2, \ldots, \omega_m \} \) of the set \([N_1] \times \cdots \times [N_d]\) and a tensor \( T, D_S(T; Y) \) is the average empirical loss according to \( g \). Precisely, \( D_S(T; Y) = \frac{1}{m} \sum_{i=1}^m g(T(\omega_i); Y(\omega_i)) \). When the sample set \( S \) is drawn i.i.d. according to \( \Pi \) (with replacement), we define:

\[
D_{\Pi}(T; Y) := \mathbb{E}_{S \sim \Pi}[g(T(\omega_i); Y(\omega_i))]
\]

\[
= \frac{1}{m} \sum_{i=1}^m \mathbb{I}_{\{\omega_i \in S\}} g(T(\omega_i), Y(\omega_i)).
\]

Notice that since \( \hat{T}_{\text{max}} \) is the optimal solution of optimization problem \((8)\) and \( T^* \) is feasible for this problem we have:

\[
D_S(\hat{T}_{\text{max}}; Y) \leq D_S(T^*; Y) = \frac{1}{m} \sum_{i=1}^m g(T^*(\omega_i); Y(\omega_i))
\]

Therefore, we have

\[
\mathbb{E}_Y [D_T(\hat{T}_{\text{max}}; Y) - D_T(T^*; Y)]
\]

\[
= \mathbb{E}_Y [D_T(\hat{T}_{\text{max}}; Y) - D_S(T^*; Y)] + \mathbb{E}_Y [D_S(T^*; Y) - D_T(T^*; Y)]
\]

\[
\leq \mathbb{E}_Y [D_T(\hat{T}_{\text{max}}; Y) - D_S(T^*; Y)]
\]

\[
\leq \sup_{T \in K_{\text{max}}(\alpha, R_{\text{max}})} \left\{ \mathbb{E}_Y [D_T(T; Y)] - \mathbb{E}_Y [D_S(T; Y)] \right\}
\]

\[
\leq \mathbb{E}_Y [D_S(T^*; Y) - D_T(T^*; Y)].
\]

Notice that the left hand side of \((17)\) is equivalent to weighted Kullback-Leibler divergence between \( f(T^*) \) and \( f(\hat{T}_{\text{max}}) \). Now we focus on bounding the right hand side of \((17)\). Using Hoeffding’s inequality on the random variable \( Z_\omega := g(T^*(\omega); Y(\omega)) - \mathbb{E}[g(T^*(\omega); Y(\omega))] \) we conclude that with probability \( 1 - \delta \) over choosing the sampling subset \( S \):

\[
D_T(T^*; Y) - D_T(T^*; Y) \leq U_\alpha \sqrt{\log(\frac{1}{\delta})}. 
\]

Moreover, a combination of [53, Th. 8, (4) of Th. 12], the Rademacher complexity of the max-qnorm ball [23, Lemma 6] and noting that \( g(\cdot, y) \) is an \( L_\alpha \)-Lipschitz function yields:

\[
\sup_{T \in K_{\text{max}}(\alpha, R_{\text{max}})} \left\{ \mathbb{E}_Y [D_T(T; Y)] - \mathbb{E}_Y [D_S(T; Y)] \right\}
\]

\[
\leq 12L_\alpha R_{\text{max}} c_1 c_2^2 \sqrt{\frac{dN}{m}} + U_\alpha \sqrt{\frac{8 \log(\frac{1}{\delta})}{m}}.
\]

Gathering \((17), (18), \) and \((19)\) we get:

\[
\mathbb{K}_\Pi(f(T^*), f(\hat{T}_{\text{max}}))
\]

\[
\leq 12L_\alpha R_{\text{max}} c_1 c_2^2 \sqrt{\frac{dN}{m}} + U_\alpha \sqrt{\frac{8 \log(\frac{1}{\delta})}{m}} + U_\alpha \sqrt{\frac{\log(\frac{1}{\delta})}{2m}}.
\]

This, together with \((22)\) and [26, Lemma 2] proves \((10)\). The upperbound \((11)\) can be proved by following the exact same arguments with the only difference being in the right hand side of \((19)\) where the Rademacher complexity of unit M-norm ball does not have the constant \( c_1 c_2^2 \) and the upper bound \( R_{\text{M}} \) is different than \( R_{\text{max}} \).

B. Proof of Theorem 12

Lemma 21 ([23, Lemma 18]): Let \( r = [\left( \frac{R_{\text{max}}}{\alpha \delta} \right)^2] \) and let \( K_{\text{M}}^*(\alpha, R) \) be defined as in Section V and let \( \gamma \leq 1 \) be such that \( \frac{\tau_2}{\tau_1} \) is an integer and suppose \( \frac{\tau_1}{\tau_2} \leq N \). Then the there exist
a set $\chi^T_{(\alpha, \gamma)} \subseteq K^T_M(\alpha, R)$ with

$$|\chi^T_{(\alpha, \gamma)}| \geq \exp \frac{rN}{16\gamma^2}$$

such that

1. For $T \in \chi^T_{(\alpha, \gamma)}$, $|T(\omega)| = \alpha \gamma$ for $\omega \in \{\mathcal{N} \times [N] \times \cdots \times [N]\}$.
2. For any $T^{(i)}$, $T^{(j)} \in \chi^T_{(\alpha, \gamma)}$, $T^{(i)} \neq T^{(j)}$

$$\|T^{(i)} - T^{(j)}\|_p^p \geq \frac{\alpha^2 r N^d}{2}.$$

Next choosing $\gamma$ in a way that $\frac{4\sqrt{2}e}{\alpha}$ is an integer and

$$\frac{4\sqrt{2}e}{\alpha} \leq \gamma \leq \frac{8e}{\alpha},$$

we transform $\chi^T_{(\alpha, \gamma)}$ so that the entries of each element of the packing set come from the set $\{\alpha, \alpha' := (1 - \gamma)\alpha\}$ by defining

$$\chi := \left\{ X + \alpha \left(1 - \frac{\gamma}{2}\right) 1 : X \in \chi^T_{(\alpha', \gamma)} \right\}.$$

Notice that for any $X \in \chi$, $\|X\|_\infty \leq \alpha$ and $\|X\|_M \leq R_M$ and therefore, we can use $\chi$ as a packing set for $K^T_M$. Once the packing sets are generated, using the exact same proof as [26, Sec. A.3.2], we can prove an upper bound. Choose $\epsilon$ to be

$$\epsilon^2 = \min \left( \frac{1}{1024}, C_2 \alpha \sqrt{\frac{\beta_2}{m}} \right) \sqrt{\frac{rN}{m}}. \quad (21)$$

And assume there is an algorithm that using 1-bit measurements $Y_i$ of $X$, returns $\hat{X}$ such that

$$\frac{1}{N^d} \|X - \hat{X}\|_F^2 \leq \epsilon^2,$$

with probability at least $\frac{1}{4}$. Notice that this means that with probability at least $\frac{1}{4}$, we can distinguish the elements of the packing set as for any $X^{(i)}$, $X^{(j)} \in \chi$, $\frac{1}{N^d} \|X^{(i)} - X^{(j)}\|_F^2 \geq 4\epsilon^2$. Hence, defining $X^*$ to be the closest element of $\chi$ to $\hat{X}$, if we show that $\mathbb{P}(X \neq X^*) > \frac{3}{4}$ the proof is done. Otherwise, using Fano’s inequality [26] and following the steps in [26, Sec. A.3.2] we have

$$\frac{1}{4} \leq 1 - \mathbb{P}(X \neq X^*) \leq 16\epsilon^2 \left( \frac{\frac{64m\epsilon^2}{\beta_2} + \frac{1}{rN}}{rN} \right).$$

$$\leq 1024\epsilon^2 \left( \frac{\frac{64m\epsilon^2}{\beta_2} + 1}{\alpha^2 r N^d} \right).$$

where $\alpha' := (1 - \gamma)\alpha$. Comparing $64m\epsilon^2$ and $\beta_2$, either

$$\frac{1}{4} \leq \frac{2048\epsilon^2}{\frac{\alpha^2 r N^d}{\beta_2}}.$$

which is a contradiction when $C_0 > 8$ or

$$\epsilon^2 \geq \frac{\alpha \sqrt{\beta_2}}{512 \sqrt{2}} \frac{rN}{m},$$

which is a contradiction when $C_2 \leq \frac{1}{512 \sqrt{2}}$.

### APPENDIX A

#### RADEMACHER COMPLEXITY

**Definition 22:** [27] Let $\mathbb{P}$ be a probability distribution on a set $\chi$ and assume the set $S := \{X_1, \ldots, X_m\}$ is $m$ independent samples drawn from $\chi$ according to $\mathbb{P}$. For a class of functions $F$ defined from $\chi$ to $\mathbb{R}$, its empirical Rademacher complexity is defined as:

$$\hat{R}_S(F) = \frac{2}{|S|} \mathbb{E}_{\varepsilon} \left[ \sup_{f \in F} \sum_{i=1}^m \varepsilon_i f(X_i) \right],$$

where $\varepsilon_i$ is a Radamacher random variable. The Rademacher complexity with respect to distribution $\mathbb{P}$ over a sample $S$ of $|S|$ point drawn independently is defined as the expectation of the empirical Rademacher complexity defined as:

$$R_S(F) = \mathbb{E}_{\mathbb{S} \sim \mathbb{P}}[\hat{R}_S(F)]$$

The following two lemmas were proved in [23].

**Lemma 23:**

$$\sup_{S: |S| = m} \hat{R}_S(\mathbb{B}_M(1)) < 6 \sqrt{\frac{dN}{m}}$$

**Lemma 24:**

$$\sup_{S: |S| = m} \hat{R}_S(\mathbb{B}_M^T(1)) < 6C_4 \epsilon^2 \sqrt{\frac{dN}{m}}$$

### APPENDIX B

#### DISCREPANCY

The Hellinger distance for two scalars $0 \leq p, q \leq 1$ is defined by:

$$d^2_{H}(p, q) = (\sqrt{p} - \sqrt{q})^2 + ((\sqrt{1 - p} - \sqrt{1 - q})^2),$$

which gives a standard notation on the distance between two probability distributions. We generalize this definition to define Hellinger between two tensors $T, U \in [0, 1]^{N_1 \times \cdots \times N_d}$ as:

$$d_{H}^2(T, U) = \frac{1}{N_1 \cdots N_d} \sum_{i_1, \ldots, i_d} d^2_{H}(S(i_1, \ldots, i_d), U(i_1, \ldots, i_d)).$$

The Kullback-Leibler divergence between two tensors $T, U$ is:

$$\mathbb{K}(T || U) := \frac{1}{N_1 \cdots N_d} \sum_{i_1, \ldots, i_d} \mathbb{K}(S(i_1, \ldots, i_d) || U(i_1, \ldots, i_d)).$$

where $\mathbb{K}(p||q) := p \log \left( \frac{p}{q} \right) + (1 - p) \log \left( \frac{1 - p}{1 - q} \right)$, for $0 \leq p, q \leq 1$. It is easy to prove that for any two scalars $0 \leq p, q \leq 1$, $d^2_{H}(p, q) \leq \mathbb{K}(p||q)$ and hence:

$$d_{H}^2(T, U) \leq \mathbb{K}(T || U). \quad (22)$$

### REFERENCES

[1] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, “Using collaborative filtering to weave an information tapestry,” *Commun. ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[2] A. Singer and M. Cucuringu, “Uniqueness of low-rank matrix completion by rigidity theory,” *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 4, pp. 1621–1641, 2010.

[3] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, “Semidefinite programming based algorithms for sensor network localization,” *ACM Trans. Sens. Netw.*, vol. 2, no. 2, pp. 188–220, 2006.
