JMLR: Workshop and Conference Proceedings vol 49:1–29, 2016

Noisy Tensor Completion via the Sum-of-Squares Hierarchy

Boaz Barak B@BOAZBARAK.ORG
Harvard John A. Paulson School of Engineering and Applied Sciences

Ankur Moitra MOITRA@MIT.EDU
Massachusetts Institute of Technology. Department of Mathematics and the Computer Science and Artificial Intelligence Lab.

Abstract

In the noisy tensor completion problem we observe m entries (whose location is chosen uniformly at random) from an unknown $n_1 \times n_2 \times n_3$ tensor T. We assume that T is entry-wise close to being rank r. Our goal is to fill in its missing entries using as few observations as possible. Let $n = \max(n_1, n_2, n_3)$. We show that if $m = n^{3/2}r$ then there is a polynomial time algorithm based on the sixth level of the sum-of-squares hierarchy for completing it. Our estimate agrees with almost all of T's entries almost exactly and works even when our observations are corrupted by noise. This is also the first algorithm for tensor completion that works in the overcomplete case when $r > n$, and in fact it works all the way up to $r = n^{3/2-\epsilon}$.

Our proofs are short and simple and are based on establishing a new connection between noisy tensor completion (through the language of Rademacher complexity) and the task of refuting random constant satisfaction problems. This connection seems to have gone unnoticed even in the context of matrix completion. Furthermore, we use this connection to show matching lower bounds. Our main technical result is in characterizing the Rademacher complexity of the sequence of norms that arise in the sum-of-squares relaxations to the tensor nuclear norm. These results point to an interesting new direction: Can we explore computational vs. sample complexity tradeoffs through the sum-of-squares hierarchy?

Keywords: tensor completion; sum-of-squares; computational vs. statistical tradeoffs

1. Introduction

Matrix completion is a central and well-studied problem in machine learning and has a diverse range of applications. One of the original motivations for it comes from the *Netflix Problem* where the goal is to predict user-movie ratings based on all the ratings we have observed so far, from across many different users. We can organize this data into a large, partially observed matrix where each row represents a user and each column represents a movie. The goal is to fill in the missing entries. The usual assumptions are that the ratings depend on only a few hidden characteristics of each user and movie and that the underlying matrix is approximately *low rank*. Another standard assumption is that it is incoherent, which we elaborate on later. How many entries of M do we need to observe in order to fill in its missing entries? And are there efficient algorithms for this task?

There have been thousands of papers on this topic and by now we have a relatively complete set of answers. A representative result (building on earlier works by Fazel (2002), Recht et al. (2010), Srebro and Shraibman (2005), Candès and Recht (2009), Candès and Tao (2010)) due to Keshavan et al. (2010) can be phrased as follows: Suppose M is an unknown $n_1 \times n_2$ matrix that has rank r but each of its entries has been corrupted by independent Gaussian noise with standard deviation δ.

Then if we observe roughly

$$m = (n_1 + n_2)r \log(n_1 + n_2)$$

of its entries, the locations of which are chosen uniformly at random, there is an algorithm that outputs a matrix X that with high probability satisfies

$$\text{err}(X) = \frac{1}{n_1 n_2} \sum_{i,j} \left| X_{i,j} - M_{i,j} \right| \leq O(\delta) \ .$$

There are extensions to non-uniform sampling models (Lee and Shraibman, 2013; Chen et al., 2014), as well as various efficiency improvements (Jain et al., 2013; Hardt, 2014). What is particularly remarkable about these guarantees is that the number of observations needed is within a logarithmic factor of the number of parameters — $(n_1 + n_2)r$ — that define the model.

In fact, there are benefits to working with even higher-order structure but so far there has been little progress on natural extensions to the tensor setting. To motivate this problem, consider the *Groupon Problem* (which we introduce here to illustrate this point) where the goal is to predict user-activity ratings. The challenge is that which activities we should recommend (and how much a user liked a given activity) depends on *time* as well — weekday/weekend, day/night, summer/fall/winter/spring, etc. or even some combination of these. As above, we can cast this problem as a large, partially observed tensor where the first index represents a user, the second index represents an activity and the third index represents the time period. It is again natural to model it as being close to low rank, under the assumption that a much smaller number of (latent) factors about the interests of the user, the type of activity and the time period should contribute to the rating. How many entries of the tensor do we need to observe in order to fill in its missing entries? This problem is emblematic of a larger issue: Can we always solve linear inverse problems when the number of observations is comparable to the number of parameters in the mode, or is computational intractability an obstacle?

In fact, one of the advantages of working with tensors is that their decompositions are unique in important ways that matrix decompositions are not. There has been a groundswell of recent work that uses tensor decompositions for exactly this reason for parameter learning in phylogenetic trees (Mossel and Roch, 2005), HMMs (Mossel and Roch, 2005), mixture models (Hsu and Kakade, 2013), topic models (Anandkumar et al., 2015) and to solve community detection (Anandkumar et al., 2013). In these applications, one assumes access to the entire tensor (up to some sampling noise). But given that the underlying tensors are low-rank, can we observe fewer of their entries and still utilize tensor methods?

A wide range of approaches to solving tensor completion have been proposed (Liu et al., 2013; Gandy et al., 2011; Signoretto et al., 2010; Tang et al., 2013; Mu et al., 2014; Kressner et al., 2014; Jain and Oh, 2014; Bhojanapalli and Sanghavi, 2015; Yuan and Zhang, 2014). However, in terms of provable guarantees none[1] of them improve upon the following näive algorithm. If the unknown tensor T is $n_1 \times n_2 \times n_3$ we can treat it as a collection of n_1 matrices each of size $n_2 \times n_3$. It is easy to see that if T has rank at most r then each of these slices also has rank at most r (and they

1. Most of the existing approaches rely on computing the tensor nuclear norm, which is hard to compute (Gurvits, 2003; Harrow and Montanaro, 2013). The only other algorithms we are aware of are due to Jain and Oh (2014) and Bhojanapalli and Sanghavi (2015) and require that the factors be orthogonal. This is a rather strong assumption. First, orthogonality requires the rank to be at most n. Second, even when $r \leq n$, most tensors need to be "whitened" to be put in this form and then a random sample from the "whitened" tensor would correspond to a (dense) linear combination of the entries of the original tensor, which would be quite a different sampling model.

inherit incoherence properties as well). By treating a third-order tensor as nothing more than an *unrelated* collection of n_1 low-rank matrices, we can complete each slice separately using roughly $m = n_1(n_2+n_3)r\log(n_2+n_3)$ observations in total. When the rank is constant, this is a *quadratic* number of observations even though the number of parameters in the model is *linear*.

Here we show how to solve the (noisy) tensor completion problem with many fewer observations. Let $n_1 \leq n_2 \leq n_3$. We give an algorithm based on the sixth level of the sum-of-squares hierarchy that can accurately fill in the missing entries of an unknown, incoherent $n_1 \times n_2 \times n_3$ tensor T that is entry-wise close to being rank r with roughly

$$m = (n_1)^{1/2}(n_2+n_3)r\log^4(n_1+n_2+n_3)$$

observations. Moreover, our algorithm works even when the observations are corrupted by noise. When $n = n_1 = n_2 = n_3$, this amounts to about $n^{1/2}r$ observations per slice which is much smaller than what we would need to apply matrix completion on each slice separately. Our algorithm needs to leverage the structure between the various slices.

1.1. Our Results

We give an algorithm for noisy tensor completion that works for third-order tensors. Let T be a third-order $n_1 \times n_2 \times n_3$ tensor that is entry-wise close to being low rank. In particular let

$$T = \sum_{\ell=1}^{r} \sigma_\ell \, a_\ell \otimes b_\ell \otimes c_\ell + \Delta \tag{1}$$

where σ_ℓ is a scalar and a_ℓ, b_ℓ and c_ℓ are vectors of length n_1, n_2 and n_3 respectively. Here Δ is a tensor that represents noise. Its entries can be thought of as representing model misspecification because T is not exactly low rank or noise in our observations or both. We will only make assumptions about the average and maximum absolute value of entries in Δ. The vectors a_ℓ, b_ℓ and c_ℓ are called *factors*, and we will assume that their norms are roughly $\sqrt{n_i}$ for reasons that will become clear later. Moreover we will assume that the magnitude of each of their entries is bounded by C in which case we call the vectors C-incoherent[2]. (Note that a random vector of dimension n and norm $\sqrt{n}$ will be $O(\sqrt{\log n})$-incoherent with high probability.) The advantage of these conventions are that a typical entry in T does not become vanishingly small as we increase the dimensions of the tensor. This will make it easier to state and interpret the error bounds of our algorithm.

Let Ω represent the locations of the entries that we observe, which (as is standard) are chosen uniformly at random and without replacement. Set $|\Omega| = m$. Our goal is to output a hypothesis X that has small entry-wise error, defined as:

$$\text{err}(X) = \frac{1}{n_1 n_2 n_3} \sum_{i,j,k} \left| X_{i,j,k} - T_{i,j,k} \right|$$

This measures the error on both the observed and unobserved entries of T. Our goal is to give algorithms that achieve *vanishing* error, as the size of the problem increases. Moreover we will want algorithms that need as few observations as possible. Here and throughout let $n_1 \leq n_2 \leq n_3$ and $n = \max\{n_1, n_2, n_3\}$. Our main result is:

2. Incoherence is often defined based on the span of the factors, but we will allow the number of factors to be larger than any of the dimensions of the tensor so we will need an alternative way to ensure that the non-zero entries of the factors are spread out

Theorem 1 (Main theorem) *Suppose we are given m observations whose locations are chosen uniformly at random (and without replacement) from a tensor T of the form (1) where each of the factors a_ℓ, b_ℓ and c_ℓ are C-incoherent. Let $\delta = \frac{1}{n_1 n_2 n_3} \sum_{i,j,k} |\Delta_{i,j,k}|$. And let $r^* = \sum_{\ell=1}^{r} |\sigma_\ell|$. Then there is a polynomial time algorithm that outputs a hypothesis X that with probability $1 - \epsilon$ satisfies*

$$err(X) \leq 4C^3 r^* \sqrt{\frac{(n_1)^{1/2}(n_2 + n_3)\log^4 n + \log 2/\epsilon}{m}} + 2\delta$$

provided that $\max_{i,j,k} |\Delta_{i,j,k}| \leq \sqrt{\frac{m}{\log 2/\epsilon}}\delta$.

Since the error bound above is quite involved, let us dissect the terms in it. In fact, having an additive δ in the error bound is unavoidable. We have not assumed anything about Δ in (1) except a bound on the average and maximum magnitude of its entries. If Δ were a random tensor whose entries are $+\delta$ and $-\delta$ then no matter how many entries of T we observe, we cannot hope to obtain error less than δ on the unobserved entries[3]. The crucial point is that the remaining term in the error bound becomes $o(1)$ when $m = \widetilde{\Omega}((r^*)^2 n^{3/2})$ which for polylogarithmic r^* improves over the näive algorithm for tensor completion by a *polynomial* factor in terms of the number of observations. Moreover our algorithm works without any constraints that factors a_ℓ, b_ℓ and c_ℓ be orthogonal or even have low inner-product.

In non-degenerate cases we can even remove another factor of r^* from the number of observations we need. Suppose that T is a tensor as in (1), but let σ_ℓ be Gaussian random variables with mean zero and variance one. The factors a_ℓ, b_ℓ and c_ℓ are still fixed, but because of the randomness in the coefficients σ_ℓ, the entries of T are now random variables.

Corollary 2 *Suppose we are given m observations whose locations are chosen uniformly at random (and without replacement) from a tensor T of the form (1), where each coefficient σ_ℓ is a Gaussian random variable with mean zero and variance one, and each of the factors a_ℓ, b_ℓ and c_ℓ are C-incoherent. Further, suppose that for a $1 - o(1)$ fraction of the entries of T, we have $\mathrm{var}(T_{i,j,k}) \geq r/\operatorname{polylog}(n) = V$ and that Δ is a tensor where each entry is a Gaussian with mean zero and variance $o(V)$. Then there is a polynomial time algorithm that outputs a hypothesis X that satisfies*

$$X_{i,j,k} = \Big(1 \pm o(1)\Big) T_{i,j,k}$$

for a $1 - o(1)$ fraction of the entries. The algorithm succeeds with probability at least $1 - o(1)$ over the randomness of the locations of the observations, and the realizations of the random variables σ_ℓ and the entries of Δ. Moreover the algorithm uses $m = C^6 n^{3/2} r \operatorname{polylog}(n)$ observations.

In the setting above, it is enough that the coefficients σ_ℓ are random and that the non-zero entries in the factors are spread out to ensure that the typical entry in T has variance about r. Consequently, the typical entry in T is about $\sqrt{r}$. This fact combined with the error bounds in Theorem 1 immediately yield the above corollary. Remarkably, the guarantee is interesting even when $r = n^{3/2-\epsilon}$ (the so-called overcomplete case). In this setting, if we observe a subpolynomial fraction of the entries of T

3. The factor of 2 is not important, and comes from needing a bound on the empirical error of how well the low rank part of T itself agrees with our observations so far. We could replace it with any other constant factor that is larger than 1.

we are able to recover almost all of the remaining entries almost entirely, even though there are no known algorithms for decomposing an overcomplete, third-order tensor even if we are given *all* of its entries, at least without imposing much stronger conditions that the factors be nearly orthogonal (Ge and Ma, 2015).

We believe that this work is a natural first step in designing practically efficient algorithms for tensor completion. Our algorithms manage to leverage the structure across the slices through the tensor, instead of treating each slice as an independent matrix completion problem. Now that we know this is *possible*, a natural follow-up question is to get more efficient algorithms. Our algorithms are based on the sixth level of the sum-of-squares hierarchy and run in polynomial time, but are quite far from being practically efficient as stated. Recent work of Hopkins et al. (2015) shows how to speed up sum-of-squares and obtain *nearly linear time* algorithms for a number of problems where the only previously known algorithms ran in a prohibitively large degree polynomial running time. Another approach would be to obtain similar guarantees for alternating minimization. Currently, the only known approaches (Jain and Oh, 2014) require that the factors are orthonormal and only work in the undercomplete case. Finally, it would be interesting to get algorithms that recover a low rank tensor exactly when there is no noise.

1.2. Our approach

All of our algorithms are based on solving the following optimization problem:

$$\min \|X\|_{\mathcal{K}} \text{ s.t. } \exists X \text{ with } \frac{1}{m} \sum_{(i,j,k)\in\Omega} |X_{i,j,k} - T_{i,j,k}| \leq 2\delta \tag{2}$$

and outputting the minimizer X, where $\|\cdot\|_{\mathcal{K}}$ is some norm that can be computed in polynomial time. It will be clear from the way we define the norm that the low rank part of T will itself be a good candidate solution. But this is not necessarily the solution that the convex program finds. How do we know that whatever it finds not only has low entry-wise error on the observed entries of T, but also on the unobserved entries too?

This is a well-studied topic in statistical learning theory, and as is standard we can use the notion of Rademacher complexity as a tool to bound the error. This has even been used in the context of matrix completion, and our work is inspired by Srebro and Shraibman (2005). The Rademacher complexity is a property of the norm we choose, and our main innovation is to use the sum-of-squares hierarchy to suggest a suitable norm. Our results are based on establishing a connection between noisy tensor completion and refuting random constraint satisfaction problems. Moreover, our analysis follows by embedding algorithms for refutation within the sum-of-squares hierarchy as a method to bound the Rademacher complexity.

A natural question to ask is: Are there other norms that have even better Rademacher complexity than the ones we use here, and that are still computable in polynomial time? It turns out that *any* such norm would immediately lead to much better algorithms for refuting random constraint satisfaction problems than we currently know. We have not yet introduced Rademacher complexity yet, so we state our lower bounds informally:

Theorem 3 (informal) *For any $\epsilon > 0$, if there is a polynomial time algorithm that achieves error*

$$err(X) \leq r^* \sqrt{\frac{n^{3/2-\epsilon}}{m}}$$

through the framework of Rademacher complexity then there is an efficient algorithm for refuting a random 3-SAT formula on n variables with $m = n^{3/2-\epsilon}$ clauses. Moreover the natural sum-of-squares relaxation requires at least $n^{2\epsilon}$-levels in order to achieve the above error (again through the framework of Rademacher complexity).

These results follow directly from the works of Grigoriev (2001), Schoenebeck (2008) and Feige (2002). There are similar connections between our upper bounds and the work of Coja-Oghlan et al. (2007) who give an algorithm for strongly refuting random 3-SAT. In Section 2 we explain some preliminary connections between these fields, at which point we will be in a better position to explain how we can borrow tools from one area to address open questions in another. We state this theorem more precisely in Corollary 16 and Corollary 33, which provide both conditional and unconditional lower bounds that match our upper bounds.

1.3. Computational vs. Sample Complexity Tradeoffs

It is interesting to compare the story of matrix completion and tensor completion. In matrix completion, we have the best of both worlds: There are efficient algorithms which work when the number of observations is close to the information theoretic minimum. In tensor completion, we gave algorithms that improve upon the number of observations needed by a polynomial factor but still require a polynomial factor more observations than can be achieved if we ignore computational considerations. We believe that for many other linear inverse problems (e.g. sparse phase retrieval), there may well be gaps between what can be achieved information theoretically and what can be achieved with computationally efficient estimators. Moreover, proving lower bounds against the sum-of-squares hierarchy offers a new type of evidence that problems are hard, that does not rely on reductions from other average-case hard problems which seem (in general) to be brittle and difficult to execute while preserving the *naturalness* of the input distribution. In fact, even when there are such reductions (Berthet and Rigollet, 2013), the sum-of-squares hierarchy offers a methodology to make sharper predictions for questions like: Is there a quasi-polynomial time algorithm for sparse PCA, or does it require exponential time?

Organization

In Section 2 we introduce Rademacher complexity, the tensor nuclear norm and strong refutation. We connect these concepts by showing that any norm that can be computed in polynomial time and has good Rademacher complexity yields an algorithm for strongly refuting random 3-SAT. In Section 3 we show how a particular algorithm for strong refutation can be embedded into the sum-of-squares hierarchy and directly leads to a norm that can be computed in polynomial time and has good Rademacher complexity. In Section 4 we establish certain spectral bounds that we need, and prove our main upper bounds. In Section 5 we prove lower bounds on the Rademacher complexity of the sequence of norms arising from the sum-of-squares hierarchy by a direct reduction to lower bounds for refuting random 3-XOR. In Appendix A we give a reduction from noisy tensor completion on asymmetric tensors to symmetric tensors. This is what allows us to extend our analysis to arbitrary order d tensors, but the proofs are essentially identical to those in the $d = 3$ case but more notationally involved so we omit them.

2. Noisy Tensor Completion and Refutation

Here we make the connection between noisy tensor completion and strong refutation explicit. Our first step is to formulate a problem that is a special case of both, and studying it will help us clarify how notions from one problem translate to the other.

2.1. The Distinguishing Problem

Here we introduce a problem that we call the *distinguishing problem*. We are given random observations from a tensor and promised that the underlying tensor fits into one of the two following categories. We want an algorithm that can tell which case the samples came from, and succeeds using as few observations as possible. The two cases are:

1. Each observation is chosen uniformly at random (and without replacement) from a tensor T where independently for each entry we set

$$T_{i,j,k} = \begin{cases} a_i a_j a_k & \text{with probability } 7/8 \\ 1 & \text{with probability } 1/16 \\ -1 & \text{else} \end{cases}$$

   where a is a vector whose entries are ± 1.

2. Alternatively, each observation is chosen uniformly at random (and without replacement) from a tensor T each of whose entries is independently set to either $+1$ or -1 and with equal probability.

In the first case, the entries of the underlying tensor T are *predictable*. It is possible to guess a $15/16$ fraction of them correctly, once we have observed enough of its entries to be able to deduce a. And in the second case, the entries of T are completely unpredictable because no matter how many entries we have observed, the remaining entries are still random. Thus we cannot predict any of the unobserved entries better than random guessing.

Now we will explain how the distinguishing problem can be equivalently reformulated in the language of refutation. We give a formal definition for strong refutation later (Definition 13), but for the time being we can think of it as the task of (given an instance of a constraint satisfaction problem) certifying that there is no assignment that satisfies many of the clauses. We will be interested in 3-XOR formulas, where there are n variables $v_1, v_2, ..., v_n$ that are constrained to take on values $+1$ or -1. Each clause takes the form

$$v_i \cdot v_j \cdot v_k = T_{i,j,k}$$

where the right hand side is either $+1$ or -1. The clause represents a parity constraint but over the domain $\{+1, -1\}$ instead of over the usual domain $\mathbb{F}_2$. We have chosen the notation suggestively so that it hints at the mapping between the two views of the problem. Each observation $T_{i,j,k}$ maps to a clause $v_i \cdot v_j \cdot v_k = T_{i,j,k}$ and vice-versa. Thus an equivalent way to formulate the distinguishing problem is that we are given a 3-XOR formula which was generated in one of the following two ways:

1. Each clause in the formula is generated by choosing an ordered triple of variables (v_i, v_j, v_k) uniformly at random (and without replacement) and we set

$$v_i \cdot v_j \cdot v_k = \begin{cases} a_i a_j a_k & \text{with probability } 7/8 \\ 1 & \text{with probability } 1/16 \\ -1 & \text{else} \end{cases}$$

   where a is a vector whose entries are ± 1. Now a represents a planted solution and by design our sampling procedure guarantees that many of the clauses that are generated are consistent with it.

2. Alternatively, each clause in the formula is generated by choosing an ordered triple of variables (v_i, v_j, v_k) uniformly at random (and without replacement) and we set $v_i \cdot v_j \cdot v_k = z_{i,j,k}$ where $z_{i,j,k}$ is a random variable that takes on values $+1$ and -1.

In the first case, the 3-XOR formula has an assignment that satisfies a $15/16$ fraction of the clauses in expectation by setting $v_i = a_i$. In the second case, any fixed assignment satisfies at most half of the clauses in expectation. Moreover if we are given $\Omega(n \log n)$ clauses, it is easy to see by applying the Chernoff bound and taking a union bound over all possible assignments that with high probability there is no assignment that satisfies more than a $1/2 + o(1)$ fraction of the clauses.

This will be the starting point for the connections we establish between noisy tensor completion and refutation. Even in the matrix case these connections seem to have gone unnoticed, and the same spectral bounds that are used to analyze the Rademacher complexity of the nuclear norm (Srebro and Shraibman, 2005) are also used to refute random 2-SAT formulas (Goerdt and Krivelevich, 2001), but this is no accident.

2.2. Rademacher Complexity

Ultimately our goal is to show that the hypothesis X that our convex program finds is entry-wise close to the unknown tensor T. By virtue of the fact that X is a feasible solution to (2) we know that it is entry-wise close to T on the observed entries. This is often called the empirical error:

Definition 4 *For a hypothesis X, the empirical error is*

$$\textit{emp-err}(X) = \frac{1}{m} \sum_{(i,j,k) \in \Omega} |X_{i,j,k} - T_{i,j,k}|$$

Recall that $\text{err}(X)$ is the average entry-wise error between X and T, over all (observed and unobserved) entries. Also recall that among the candidate X's that have low empirical error, the convex program finds the one that minimizes $\|X\|_{\mathcal{K}}$ for some polynomial time computable norm. The way we will choose the norm $\|\cdot\|_{\mathcal{K}}$ and our bound on the maximum magnitude of an entry of Δ will guarantee that the low rank part of T will with high probability be a feasible solution. This ensures that $\|X\|_{\mathcal{K}}$ for the X we find is not too large either. One way to bound $\text{err}(X)$ is to show that no hypothesis in the unit norm ball can have too large a gap between its error and its empirical error (and then dilate the unit norm ball so that it contains X). With this in mind, we define:

Definition 5 *For a norm $\|\cdot\|_{\mathcal{K}}$ and a set Ω of observations, the generalization error is*

$$\sup_{\|X\|_{\mathcal{K}}\leq 1} \Big| \textit{err}(X) - \textit{emp-err}(X) \Big|$$

It turns out that one can bound the generalization error via the Rademacher complexity.

Definition 6 *Let $\Omega = \{(i_1,j_1,k_1),(i_2,j_2,k_2),...,(i_m,j_m,k_m)\}$ be a set of m locations chosen uniformly at random (and without replacement) from $[n_1]\times[n_2]\times[n_3]$. And let $\sigma_1,\sigma_2,...,\sigma_\ell$ be random ± 1 variables. The Rademacher complexity of (the unit ball of) the norm $\|\cdot\|_{\mathcal{K}}$ is defined as*

$$R^m(\|\cdot\|_{\mathcal{K}}) = \mathop{\mathbf{E}}_{\Omega,\sigma}\Big[\sup_{\|X\|_{\mathcal{K}}\leq 1}\Big|\sum_{\ell=1}^m \sigma_\ell X_{i_\ell,j_\ell,k_\ell}\Big|\Big]$$

Standard symmetrization arguments from empirical process theory show that the Rademacher complexity can be used to bound the generalization error. In particular, the following theorem follows from Theorem 6, Part 12 and Theorem 8 in (Bartlett and Mendelson, 2003):

Theorem 7 *Let $\epsilon\in(0,1)$ and suppose each X with $\|X\|_{\mathcal{K}}\leq 1$ has bounded loss — i.e. $|X_{i,j,k}-T_{i,j,k}|\leq a$ and that locations (i,j,k) are chosen uniformly at random and without replacement. Then with probability at least $1-\epsilon$, for every X with $\|X\|_{\mathcal{K}}\leq 1$, we have*

$$\textit{err}(X) \leq \textit{emp-err}(X) + 2R^m(\|\cdot\|_{\mathcal{K}}) + 2\frac{\max_{i,j,k}|T_{i,j,k}|}{\sqrt{m}} + 2a\sqrt{\frac{\ln(1/\epsilon)}{m}}$$

We remark that generalization bounds are often stated in the setting where samples are drawn i.i.d., but here the locations of our observations are sampled without replacement. Nevertheless for the settings of m we are interested in, the fraction of our observations that are repeats is $o(1)$ — in fact it is subpolynomial — and we can move back and forth between both sampling models at negligible loss in our bounds.

In much of what follows it will be convenient to think of

$$\Omega = \{(i_1,j_1,k_1),(i_2,j_2,k_2),...,(i_m,j_m,k_m)\}$$

and $\{\sigma_\ell\}_\ell$ as being represented by a sparse tensor Z, defined below.

Definition 8 *Let Z be an $n_1\times n_2\times n_3$ tensor such that*

$$Z_{i,j,k} = \begin{cases} 0, \textit{ if } (i,j,k)\notin\Omega \\ \sum_\ell \textit{ s.t. } {\scriptstyle (i,j,k)=(i_\ell,j_\ell,k_\ell)}\ \sigma_\ell \end{cases}$$

This definition greatly simplifies our notation. In particular we have

$$\sum_{\ell=1}^m \sigma_\ell X_{i_\ell,j_\ell,k_\ell} = \sum_{i,j,k} Z_{i,j,k}X_{i,j,k} = \langle Z, X\rangle$$

where we have introduced the notation $\langle\ \cdot\ ,\ \cdot\ \rangle$ to denote the natural inner-product between tensors. Our main technical goal in this paper will be to analyze the Rademacher complexity of a sequence of successively tighter norms that we get from the sum-of-squares hierarchy, and to derive implications for noisy tensor completion and for refutation from these bounds.

2.3. The Tensor Nuclear Norm

Here we introduce the tensor nuclear norm and analyze its Rademacher complexity. Many works have suggested using it to solve tensor completion problems (Liu et al., 2013; Signoretto et al., 2010; Yuan and Zhang, 2014). This suggestion is quite natural given that it is based on a similar guiding principle as that which led to ℓ_1-minimization in compressed sensing and the nuclear norm in matrix completion (Fazel, 2002). More generally, one can define the atomic norm for a wide range of linear inverse problems (Chandrasekaran et al., 2012), and the ℓ_1-norm, the nuclear norm and the tensor nuclear norm are all special cases of this paradigm. Before we proceed, let us first formally define the notion of incoherence that we gave in the introduction.

Definition 9 *A length n_i vector a is* C-incoherent *if $\|a\| = \sqrt{n_i}$ and $\|a\|_\infty \leq C$.*

Recall that we chose to work with vectors whose typical entry is a constant so that the entries in T do not become vanishingly small as the dimensions of the tensor increase. We can now define the tensor nuclear norm[4]:

Definition 10 (tensor nuclear norm) *Let $\mathcal{A} \subseteq \mathbb{R}^{n_1 \times n_2 \times n_3}$ be defined as*

$$\mathcal{A} = \left\{ X \text{ s.t. } \exists \text{ distribution } \mu \text{ on triples of } C\text{-incoherent vectors with } X_{i,j,k} = \mathop{\mathbf{E}}_{(a,b,c)\leftarrow\mu}[a_i b_j c_k] \right\}$$

The tensor nuclear norm *of X which is denoted by $\|X\|_\mathcal{A}$ is the infimum over α such that $X/\alpha \in \mathcal{A}$.*

In particular $\|T - \Delta\|_\mathcal{A} \leq r^*$. Finally we give an elementary bound on the Rademacher complexity of the tensor nuclear norm. Recall that $n = \max(n_1, n_2, n_3)$.

Lemma 11 $R^m(\|\cdot\|_\mathcal{A}) = O(C^3\sqrt{\frac{n}{m}})$

Proof Recall the definition of Z given in Definition 8. With this we can write

$$\mathop{\mathbf{E}}_{\Omega,\sigma}\left[\sup_{\|X\|_\mathcal{A}\leq 1}\left|\sum_{\ell=1}^m \sigma_\ell X_{i_\ell,j_\ell,k_\ell}\right|\right] = \mathop{\mathbf{E}}_{\Omega,\sigma}\left[\sup_{C\text{-incoherent } a,b,c}|\langle Z, a\otimes b\otimes c\rangle|\right]$$

We can now adapt the discretization approach in Friedman et al. (1989), although our task is considerably simpler because we are constrained to C-incoherent a's. In particular, let

$$S = \left\{ a \middle| a \text{ is } C\text{-incoherent and } a \in \left(\epsilon\mathbb{Z}\right)^n \right\}$$

By standard bounds on the size of an ϵ-net (Matoušek, 2002), we get that $|S| \leq O(C/\epsilon)^n$. Suppose that $|\langle Z, a\otimes b\otimes c\rangle| \leq M$ for all $a, b, c \in S$. Then for an arbitrary, but C-incoherent a we can expand it as $a = \sum_i \epsilon^i a^i$ where each $a^i \in S$ and similarly for b and c. And now

$$|\langle Z, a\otimes b\otimes c\rangle| \leq \sum_i\sum_j\sum_k \epsilon^i\epsilon^j\epsilon^k|\langle Z, a^i\otimes b^i\otimes c^i\rangle| \leq (1-\epsilon)^{-3}M$$

4. The usual definition of the tensor nuclear norm has no constraints that the vectors a, b and c be C-incoherent. However, adding this additional requirement only serves to further restrict the unit norm ball, while ensuring that the low rank part of T (when scaled down) is still in it, since the factors of T are anyways assumed to be C-incoherent. This makes it easier to prove recovery guarantees because we do not need to worry about sparse vectors behaving very differently than incoherent ones, and since we are not going to compute this norm anyways this modification will make our analysis easier.

Moreover since each entry in $a \otimes b \otimes c$ has magnitude at most C^3 we can apply a Chernoff bound to conclude that for any particular $a, b, c \in S$ we have

$$|\langle Z, a \otimes b \otimes c \rangle| \leq O\Big(C^3 \sqrt{m \log 1/\gamma}\Big)$$

with probability at least $1 - \gamma$. Finally, if we set $\gamma = (\frac{\epsilon}{C})^{-n}$ and we set $\epsilon = 1/2$ we get that

$$R^m(\mathcal{A}) \leq \frac{(1-\epsilon)^{-3}}{m} \max_{a,b,c \in S} |\langle Z, a \otimes b \otimes c \rangle| = O\Big(C^3 \sqrt{\frac{n}{m}}\Big)$$

and this completes the proof. ■

The important point is that the Rademacher complexity of the tensor nuclear norm is $o(1)$ whenever $m = \omega(n)$. In the next subsection we will connect this to refutation in a way that allows us to strengthen known hardness results for computing the tensor nuclear norm (Gurvits, 2003; Harrow and Montanaro, 2013) and show that it is even hard to compute in an average-case sense based on some standard conjectures about the difficulty of refuting random 3-SAT.

2.4. From Rademacher Complexity to Refutation

Here we show the first implication of the connection we have established. Any norm that can be computed in polynomial time and has good Rademacher complexity immediately yields an algorithm for strongly refuting random 3-SAT and 3-XOR formulas. Next let us finally define strong refutation.

Definition 12 *For a formula ϕ, let $\mathrm{opt}(\phi)$ be the largest fraction of clauses that can be satisfied by any assignment.*

In what follows, we will use the term *random 3-XOR formula* to refer to a formula where each clause is generated by choosing an ordered triple of variables (v_i, v_j, v_k) uniformly at random (and without replacement) and setting $v_i \cdot v_j \cdot v_k = z$ where z is a random variable that takes on values $+1$ and -1.

Definition 13 *An algorithm for strongly refuting random 3-XOR takes as input a 3-XOR formula ϕ and outputs a quantity $alg(\phi)$ that satisfies*

1. *For any 3-XOR formula ϕ, $\mathrm{opt}(\phi) \leq alg(\phi)$*
2. *If ϕ is a random 3-XOR formula with m clauses, then with high probability $alg(\phi) = 1/2 + o(1)$*

This definition only makes sense when m is large enough so that $\mathrm{opt}(\phi) = 1/2 + o(1)$ holds with high probability, which happens when $m = \omega(n)$. The goal is to design algorithms that use as few clauses as possible, and are able to certify that a random formula is indeed far from satisfiable (without underestimating the fraction of clauses that can be satisfied) and to do so as close as possible to the information theoretic threshold.

Now let us use a polynomial time computable norm $\|\cdot\|_{\mathcal{K}}$ that has good Rademacher complexity to give an algorithm for strongly refuting random 3-XOR. As in Section 2.1, given a formula ϕ we

map its m clauses to a collection of m observations according to the usual rule: If there are n variables, we construct an $n \times n \times n$ tensor Z where for each clause of the form $v_i \cdot v_j \cdot v_k = z_{i,j,k}$ we put the entry $z_{i,j,k}$ at location (i, j, k). All the rest of the entries in Z are set to zero. We solve the following optimization problem:

$$\max \eta \text{ s.t. } \exists X \text{ with } \|X\|_{\mathcal{K}} \leq 1 \text{ and } \frac{1}{m}\langle Z, X\rangle \geq 2\eta \tag{3}$$

Let η^* be the optimum value. We set $\text{alg}(\phi) = 1/2 + \eta^*$. What remains is to prove that the output of this algorithm solves the strong refutation problem for 3-XOR.

Theorem 14 *Suppose that $\|\cdot\|_{\mathcal{K}}$ is computable in polynomial time and satisfies $\|X\|_{\mathcal{K}} \leq 1$ whenever $X = a \otimes a \otimes a$ and a is a vector with ± 1 entries. Further suppose that for any X with $\|X\|_{\mathcal{K}} \leq 1$ its entries are bounded by C^3 in absolute value. Then (3) can be solved in polynomial time and if $R^m(\|\cdot\|_{\mathcal{K}}) = o(1)$ then setting $alg(\phi) = 1/2 + \eta^*$ solves strong refutation for 3-XOR with $O(C^6 m \log n)$ clauses.*

Proof The key observation is the following inequality which relates (3) to $\text{opt}(\phi)$.

$$2\,\text{opt}(\phi) - 1 \leq \frac{1}{m} \sup_{\|X\|_{\mathcal{K}} \leq 1} \langle Z, X\rangle$$

To establish this inequality, let $v_1, v_2, ..., v_n$ be the assignment that maximizes the fraction of clauses satisfied. If we set $a_i = v_i$ and $X = a \otimes a \otimes a$ we have that $\|X\|_{\mathcal{K}} \leq 1$ by assumption. Thus X is a feasible solution. Now with this choice of X for the right hand side, every term in the sum that corresponds to a satisfied clause contributes $+1$ and every term that corresponds to an unsatisfied clause contributes -1. We get $2\,\text{opt}(\phi) - 1$ for this choice of X, and this completes the proof of the inequality above.

The crucial point is that the expectation of the right hand side over Ω and σ is exactly the Rademacher complexity. However we want a bound that holds with high probability instead of just in expectation. It follows from McDiarmid's inequality and the fact that the entries of Z and of X are bounded by 1 and by C^3 in absolute value respectively that if we take $O(C^6 m \log n)$ observations the right hand side will be $o(1)$ with high probability. In this case, rearranging the inequality we have

$$\text{opt}(\phi) \leq 1/2 + \frac{1}{m} \sup_{\|X\|_{\mathcal{K}} \leq 1} \langle Z, X\rangle$$

The right hand side is exactly $\text{alg}(\phi)$ and is $1/2 + o(1)$ with high probability, which implies that both conditions in the definition for strong refutation hold and this completes the proof. ■

We can now combine Theorem 14 with the bound on the Rademacher complexity of the tensor nuclear norm given in Lemma 11 to conclude that if we could compute the tensor nuclear norm we would also obtain an algorithm for strongly refuting random 3-XOR with only $m = \Omega(n \log n)$ clauses. It is not obvious but it turns out that any algorithm for strongly refuting random 3-XOR implies one for 3-SAT. Let us define strong refutation for 3-SAT. We will refer to any variable v_i or its negation $\bar{v}_i$ as a literal. We will use the term *random 3-SAT formula* to refer to a formula where each clause is generated by choosing an ordered triple of literals (y_i, y_j, y_k) uniformly at random (and without replacement) and setting $y_i \vee y_j \vee y_k = 1$.

Definition 15 *An algorithm for strongly refuting random 3-SAT takes as input a 3-SAT formula ϕ and outputs a quantity $alg(\phi)$ that satisfies*

1. *For any 3-SAT formula ϕ, $\mathrm{opt}(\phi) \leq alg(\phi)$*
2. *If ϕ is a random 3-SAT formula with m clauses, then with high probability $alg(\phi) = 7/8 + o(1)$*

The only change from Definition 13 comes from the fact that for 3-SAT a random assignment satisfies a $7/8$ fraction of the clauses in expectation. Our goal here is to certify that the largest fraction of clauses that can be satisfied is $7/8 + o(1)$. The connection between refuting random 3-XOR and 3-SAT is often called "Feige's XOR Trick" (Feige, 2002). The first version of it was used to show that an algorithm for ϵ-refuting 3-XOR can be turned into an algorithm for ϵ-refuting 3-SAT. However we will not use this notion of refutation so for further details we refer the reader to Feige (2002). The reduction was extended later by Coja-Oghlan et al. (2007) to strong refutation, which for us yields the following corollary:

Corollary 16 *Suppose that $\|\cdot\|_{\mathcal{K}}$ is computable in polynomial time and satisfies $\|X\|_{\mathcal{K}} \leq 1$ whenever $X = a \otimes a \otimes a$ and a is a vector with ± 1 entries. Suppose further that for any X with $\|X\|_{\mathcal{K}} \leq 1$ its entries are bounded by C^3 in absolute value and that $R^m(\|\cdot\|_{\mathcal{K}}) = o(1)$. Then there is a polynomial time algorithm for strongly refuting a random 3-SAT formula with $O(C^6 m \log n)$ clauses.*

Now we can get a better understanding of the obstacles to noisy tensor completion by connecting it to the literature on refuting random 3-SAT. Despite a long line of work on refuting random 3-SAT (Goerdt and Krivelevich, 2001; Friedman et al., 2005; Feige and Ofek, 2007; Feige et al., 2006; Coja-Oghlan et al., 2007), there is no known polynomial time algorithm that works with $m = n^{3/2-\epsilon}$ clauses for any $\epsilon > 0$. Feige (2002) conjectured that for any constant C, there is no polynomial time algorithm for refuting random 3-SAT with $m = Cn$ clauses[5]. Daniely et al. (2013) conjectured that there is no polynomial time algorithm for $m = n^{3/2-\epsilon}$ for any $\epsilon > 0$. What we have shown above is that any norm that is a relaxation to the tensor nuclear norm and can be computed in polynomial time but has Rademacher complexity is $R^m(\|\cdot\|_{\mathcal{K}}) = o(1)$ for $m = n^{3/2-\epsilon}$ would disprove the conjecture of Daniely et al. (2013) and would yield much better algorithms for refuting random 3-SAT than we currently know, despite fifteen years of work on the subject.

This leaves open an important question. While there are no known algorithms for strongly refuting random 3-SAT with $m = n^{3/2-\epsilon}$ clauses, there are algorithms that work with roughly $m = n^{3/2}$ clauses (Coja-Oghlan et al., 2007). Do these algorithms have any implications for noisy tensor completion? We will adapt the algorithm of Coja-Oghlan et al. (2007) and embed it within the sum-of-squares hierarchy. In turn, this will give us a norm that we can use to solve noisy tensor completion which uses a polynomial factor fewer observations than known algorithms.

5. In Feige's paper (Feige, 2002) there was no need to make the conjecture any stronger because it was already strong enough for all of the applications in inapproximability.

3. Using Resolution to Bound the Rademacher Complexity

3.1. Pseudo-expectation

Here we introduce the sum-of-squares hierarchy and will use it (at level six) to give a relaxation to the tensor nuclear norm. This will be the norm that we will use in proving our main upper bounds. First we introduce the notion of a pseudo-expectation operator from (Barak et al., 2014, 2015; Barak and Steurer, 2014):

Definition 17 (Pseudo-expectation (Barak et al., 2014)) *Let k be even and let $P_k^{n'}$ denote the linear subspace of all polynomials of degree at most k on n' variables. A linear operator $\widetilde{\mathbf{E}} : P_k^{n'} \rightarrow \mathbb{R}$ is called a* degree k pseudo-expectation operator *if it satisfies the following conditions:*

(1) $\widetilde{\mathbf{E}}[1] = 1$ *(*normalization*)*

(2) $\widetilde{\mathbf{E}}[P^2] \geq 0$*, for any degree at most $k/2$ polynomial P (*nonnegativity*)*

Moreover suppose that $p \in P_k^{n'}$ with $deg(p) = k'$. We say that $\widetilde{\mathbf{E}}$ satisfies the constraint $\{p = 0\}$ if $\widetilde{\mathbf{E}}[pq] = 0$ for every $q \in P_{k-k'}^{n'}$. And we say that $\widetilde{\mathbf{E}}$ satisfies the constraint $\{p \geq 0\}$ if $\widetilde{\mathbf{E}}[pq^2] \geq 0$ for every $q \in P_{\lfloor (k-k')/2 \rfloor}^{n'}$.

The rationale behind this definition is that if μ is a distribution on vectors in $\mathbb{R}^{n'}$ then the operator $\widetilde{\mathbf{E}}[p] = \mathbf{E}_{Y \leftarrow \mu}[p(Y)]$ is a degree d pseudo-expectation operator for every d — i.e. it meets the conditions of Definition 17. However the converse is in general not true. We are now ready to define the norm that will be used in our upper bounds:

Definition 18 (SOS_k norm) *We let $\mathcal{K}_k$ be the set of all $X \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ such that there exists a degree k pseudo-expectation operator on $P_k^{n_1+n_2+n_3}$ satisfying the following polynomial constraints (where the variables are the $Y_i^{(a)}$'s)*

(a) $\{\sum_{i=1}^{n_1} (Y_i^{(1)})^2 = n_1\}$, $\{\sum_{i=1}^{n_2} (Y_i^{(2)})^2 = n_2\}$ *and* $\{\sum_{i=1}^{n_3} (Y_i^{(3)})^2 = n_3\}$

(b) $\{(Y_i^{(1)})^2 \leq C^2\}$, $\{(Y_i^{(2)})^2 \leq C^2\}$ *and* $\{(Y_i^{(3)})^2 \leq C^2\}$ *for all i and*

(c) $X_{i,j,k} = \widetilde{\mathbf{E}}[Y_i^{(1)} Y_j^{(2)} Y_k^{(3)}]$ *for all i, j and k.*

The SOS_k norm *of $X \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ which is denoted by $\|X\|_{\mathcal{K}_k}$ is the infimum over α such that $X/\alpha \in \mathcal{K}_k$.*

The constraints in Definition 17 can be expressed as an $O(n^k)$-sized semidefinite program. This implies that given any set of polynomial constraints of the form $\{p = 0\}$, $\{p \geq 0\}$, one can efficiently find a degree k pseudo-distribution satisfying those constraints if one exists. This is often called the *degree k Sum-of-Squares algorithm* (Shor, 1988; Nesterov, 2000; Lasserre, 2001; Parrilo, 2000). Hence we can compute the norm $\|X\|_{\mathcal{K}_k}$ of any tensor X to within arbitrary accuracy in polynomial time. And because it is a relaxation to the tensor nuclear norm which is defined analogously but over a distribution on C-incoherent vectors instead of a pseudo-distribution over them, we have that $\|X\|_{\mathcal{K}_k} \leq \|X\|_{\mathcal{A}}$ for every tensor X. Throughout most of this paper, we will be interested in the case $k = 6$.

3.2. Resolution in $\mathcal{K}_6$

Recall that any polynomial time computable norm with good Rademacher complexity with m observations yields an algorithm for strong refutation with roughly m clauses too. Here we will use an algorithm for strongly refuting random 3-SAT to guide our search for an appropriate norm. We will adapt an algorithm due to Coja-Oghlan et al. (2007) that strongly refutes random 3-SAT, and will instead give an algorithm that strongly refutes random 3-XOR. Moreover each of the steps in the algorithm embeds into the sixth level of the sum-of-squares hierarchy by mapping resolution operations to applications of Cauchy-Schwartz, that ultimately show how the inequalities that define the norm (Definition 18) can be manipulated to give bounds on its own Rademacher complexity.

Let's return to the task of bounding the Rademacher complexity of $\|\cdot\|_{\mathcal{K}_6}$. Let X be arbitrary but satisfy $\|X\|_{\mathcal{K}_6} \leq 1$. Then there is a degree six pseudo-expectation meeting the conditions of Definition 18. Using Cauchy-Schwartz we have:

$$\Big(\langle Z, X\rangle\Big)^2 = \Big(\sum_i \sum_{j,k} Z_{i,j,k} \widetilde{\mathbf{E}}[Y_i^{(1)} Y_j^{(2)} Y_k^{(3)}]\Big)^2 \leq n_1 \Big(\sum_i \Big(\sum_{j,k} Z_{i,j,k} \widetilde{\mathbf{E}}[Y_i^{(1)} Y_j^{(2)} Y_k^{(3)}]\Big)^2\Big) \tag{4}$$

To simplify our notation, we will define the following polynomial

$$Q_{i,Z}(Y^{(2)}, Y^{(3)}) = \sum_{j,k} Z_{i,j,k} Y_j^{(2)} Y_k^{(3)}$$

which we will use repeatedly. If d is even then any degree d pseudo-expectation operator satisfies the constraint $(\widetilde{\mathbf{E}}[p])^2 \leq \widetilde{\mathbf{E}}[p^2]$ for every polynomial p of degree at most $d/2$ (e.g., see Lemma $A.4$ in Barak et al. (2012)). Hence the right hand side of (4) can be bounded as:

$$n_1 \Big(\sum_i \Big(\widetilde{\mathbf{E}}[Y_i^{(1)} Q_{i,Z}(Y^{(2)}, Y^{(3)})]\Big)^2\Big) \leq n_1 \sum_i \widetilde{\mathbf{E}}\Big[\Big(Y_i^{(1)} Q_{i,Z}(Y^{(2)}, Y^{(3)})\Big)^2\Big] \tag{5}$$

It turns out that bounding the right-hand side of (5) boils down to bounding the spectral norm of the following matrix.

Definition 19 *Let A be the $n_2 n_3 \times n_2 n_3$ matrix whose rows and columns are indexed over ordered pairs (j, k') and (j', k) respectively, defined as*

$$A_{j,k',j',k} = \sum_i Z_{i,j,k} Z_{i,j',k'}$$

We can now make the connection to resolution more explicit: We can think of a pair of observations $Z_{i,j,k}, Z_{i,j',k'}$ as a pair of 3-XOR constraints, as usual. Resolving them (i.e. multiplying them) we obtain a 4-XOR constraint

$$x_j \cdot x_k \cdot x_{j'} \cdot x_{k'} = Z_{i,j,k} Z_{i,j',k'}$$

A captures the effect of resolving certain pairs of 3-XOR constraints into 4-XOR constraints. The challenge is that the entries in A are not independent, so bounding its maximum singular value will require some care. It is important that the rows of A are indexed by (j, k') and the columns are indexed by (j', k), so that j and j' come from different 3-XOR clauses, as do k and k', and

otherwise the spectral bounds that we will want to prove about A would simply not be true! This is perhaps the key insight in Coja-Oghlan et al. (2007).

It will be more convenient to decompose A and reason about its two types of contributions separately. To that end, we let R be the $n_2n_3 \times n_2n_3$ matrix whose non-zero entries are of the form

$$R_{j,k,j,k} = \sum_i Z_{i,j,k}Z_{i,j,k}$$

and all of its other entries are set to zero. Then let B be the $n_2n_3 \times n_2n_3$ matrix whose entries are of the form

$$B_{j,k',j',k} = \begin{cases} 0, \text{ if } j = j' \text{ and } k = k' \\ \sum_i Z_{i,j,k}Z_{i,j',k'} \text{ else} \end{cases}$$

By construction we have $A = B + R$. Finally:

Lemma 20

$$\sum_i \widetilde{\mathbf{E}}\left[\left(Y_i^{(1)}Q_{i,Z}(Y^{(2)}, Y^{(3)})\right)^2\right] \leq C^2n_2n_3\|B\| + C^6m$$

Proof The pseudo-expectation operator satisfies $\{(Y_i^{(1)})^2 \leq C^2\}$ for all i, and hence we have

$$\begin{aligned}\textstyle\sum_i \widetilde{\mathbf{E}}\left[\left(Y_iQ_{i,Z}(Y^{(2)}, Y^{(3)})\right)^2\right] &\leq C^2 \textstyle\sum_i \widetilde{\mathbf{E}}\left[\left(Q_{i,Z}(Y^{(2)}, Y^{(3)})\right)^2\right] \\ &= C^2 \textstyle\sum_i \sum_{j,k,j',k'} \widetilde{\mathbf{E}}\left[Z_{i,j,k}Z_{i,j',k'}Y_j^{(2)}Y_k^{(3)}Y_{j'}^{(2)}Y_{k'}^{(3)}\right]\end{aligned}$$

Now let $Y^{(2)} \in \mathbb{R}^{n_2}$ be a vector of variables where the ith entry is $Y_i^{(2)}$ and similarly for $Y^{(3)}$. Then we can re-write the right hand side as a matrix inner-product:

$$C^2 \sum_i \sum_{j,k,j',k'} Z_{i,j,k}Z_{i,j',k'}\widetilde{\mathbf{E}}[Y_j^{(2)}Y_k^{(3)}Y_{j'}^{(2)}Y_{k'}^{(3)}] = C^2\langle A, \widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]\rangle$$

We will now bound the contribution of B and R separately.

Claim 21 $\widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]$ *is positive semidefinite and has trace at most* n_2n_3

Proof It is easy to see that a quadratic form on $\widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]$ corresponds to $\widetilde{\mathbf{E}}[p^2]$ for some $p \in P_2^{n_2+n_3}$ and this implies the first part of the claim. Finally

$$\text{Tr}(\widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]) = \sum_{j,k} \widetilde{\mathbf{E}}[(Y_j^{(2)})^2(Y_k^{(3)})^2] \leq n_2n_3$$

where the last equality follows because the pseudo-expectation operator satisfies the constraints $\{\sum_{i=1}^{n_2}(Y_i^{(2)})^2 = n_2\}$ and $\{\sum_{i=1}^{n_3}(Y_i^{(3)})^2 = n_3\}$. ■

Hence we can bound the contribution of the first term as $C^2\langle B, \widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]]\rangle \leq C^2n_2n_3\|B\|$. Now we proceed to bound the contribution of the second term:

Claim 22 $\widetilde{\mathbf{E}}[(Y_j^{(2)})^2(Y_k^{(3)})^2] \leq C^4$

Proof It is easy to verify by direction computation that the following equality holds:

$$C^4-(Y_j^{(2)})^2(Y_k^{(3)})^2 = \Big(C^2-(Y_j^{(2)})^2\Big)\Big(C^2-(Y_k^{(3)})^2\Big)+\Big(C^2-(Y_k^{(3)})^2\Big)(Y_j^{(2)})^2+\Big(C^2-(Y_j^{(2)})^2\Big)(Y_k^{(3)})^2$$

Moreover the pseudo-expectation of each of the three terms above is nonnegative, by construction. This implies the claim. ■

Moreover each entry in Z is in the set $\{-1, 0, +1\}$ and there are precisely m non-zeros. Thus the sum of the absolute values of all entries in R is at most m. Now we have:

$$C^2\langle R, \widetilde{\mathbf{E}}[(Y^{(2)} \otimes Y^{(3)})(Y^{(2)} \otimes Y^{(3)})^T]\rangle \leq C^2 \sum_{j,k} R_{j,k,j,k}\widetilde{\mathbf{E}}[(Y_j^{(2)})^2(Y_k^{(3)})^2] \leq C^6 m$$

And this completes the proof of the lemma. ■

4. Spectral Bounds

Recall the definition of B given in the previous section. In fact, for our spectral bounds it will be more convenient to relabel the variables (but keeping the definition intact):

$$B_{j,k,j',k'} = \begin{cases} 0, \text{ if } j = j' \text{ and } k = k' \\ \sum_i Z_{i,j,k'} Z_{i,j',k} \text{ else} \end{cases}$$

Let us consider the following random process: For $r = 1, 2, ..., O(\log n)$ partition the set of all ordered triples (i, j, k) into two sets S_r and T_r. We will use this ensemble of partitions to define an ensemble of matrices $\{\mathsf{B}^r\}_{r=1}^{O(\log n)}$: Set $U^r_{i,j,k'}$ as equal to $Z_{i,j,k'}$ if $(i, j, k') \in S_r$ and zero otherwise. Similarly set $V^r_{i,j',k}$ equal to $Z_{i,j',k}$ if $(i, j', k) \in T_r$ and zero otherwise. Also let $E_{i,j,j',k,k',r}$ be the event that there is no $r' < r$ where $(i, j, k') \in S_{r'}$ and $(i, j', k) \in T_{r'}$ or vice-versa. Now let

$$\mathsf{B}^r_{j,k,j',k'} = \sum_i U^r_{i,j,k'} V^r_{i,j',k} \mathbb{1}_E$$

where $\mathbb{1}_E$ is short-hand for the indicator function of the event $E_{i,j,j',k,k',r}$. The idea behind this construction is that each pair of triples (i, j, k') and (i, j', k) that contributes to B will be contribute to some B^r with high probability. Moreover it will not contribute to any later matrix in the ensemble. Hence with high probability

$$B = \sum_{r=1}^{O(\log n)} \mathsf{B}^r$$

Throughout the rest of this section, we will suppress the superscript r and work with a particular matrix in the ensemble, B. Now let ℓ be even and consider

$$\mathrm{Tr}(\underbrace{\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T}_{\ell \text{ times}})$$

As is standard, we are interested in bounding $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]$ in order to bound $\|\mathsf{B}\|$. But note that B is *not* symmetric. Also note that the random variables U and V are not independent,

however whether or not they are non-zero is non-positively correlated and their signs are mutually independent. Expanding the trace above we have

$$\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T) = \sum_{j_1,k_1}\sum_{j_2,k_2}...\sum_{j_{\ell-1},k_{\ell-1}}\mathsf{B}_{j_1,k_1,j_2,k_2}\mathsf{B}_{j_3,k_3,j_2,k_2}...\mathsf{B}_{j_1,k_1,j_\ell,k_\ell}$$
$$= \sum_{j_1,k_1}\sum_{i_1}\sum_{j_2,k_2}\sum_{i_2}...\sum_{j_\ell,k_\ell}\sum_{i_\ell}U_{i_1,j_1,k_2}V_{i_1,j_2,k_1}\mathbb{1}_{E_1}U_{i_2,j_3,k_2}V_{i_2,j_2,k_3}\mathbb{1}_{E_2}...U_{i_\ell,j_1,k_\ell}V_{i_\ell,j_\ell,k_1}\mathbb{1}_{E_\ell}$$

where $\mathbb{1}_{E_1}$ is the indicator for the event that the entry $\mathsf{B}_{j_1,k_1,j_2,k_2}$ is not covered by an earlier matrix in the ensemble, and similarly for $\mathbb{1}_{E_2}, ..., \mathbb{1}_{E_\ell}$.

Notice that there are 2ℓ random variables in the above sum (ignoring the indicator variables). Moreover if any U or V random variable appears an odd number of times, then the contribution of the term to $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]$ is zero. We will give an encoding for each term that has a non-zero contribution, and we will prove that it is injective.

Fix a particular term in the above sum where each random variable appears an even number of times. Let s be the number of distinct values for i. Moreover let $i_1, i_2, ..., i_s$ be the order that these indices first appear. Now let r_1^j denote the number of distinct values for j that appear with i_1 in U terms — i.e. r_1^j is the number of distinct j's that appear as $U_{i_1,j,*}$. Let r_1^k denote the number of distinct values for k that appear with i_1 in U terms — i.e. r_1^k is the number of distinct k's that appear as or $U_{i_1,*,k}$. Similarly let q_1^j denote the number of distinct values for j that appear with i_1 in V terms — i.e. q_1^j is the number of distinct j's that appear as $V_{i_1,j,*}$. And finally let q_1^k denote the number of distinct values for k that appear with i_1 in V terms — i.e. q_1^k is the number of distinct k's that appear as $V_{i_1,*,k}$.

We give our encoding below. It is more convenient to think of the encoding as any way to answer the following questions about the term.

(a) What is the order $i_1, i_2, ..., i_s$ of the first appearance of each distinct value of i?

(b) For each i that appears, what is the order of each of the distinct values of j's and k's that appear along with it in U? Similarly, what is the order of each of the distinct values of j's and k's that appear along with it in V?

(c) For each step (i.e. a new variable in the term when reading from left to right), has the value of i been visited already? Also, has the value for j or k that appears along with U been visited? Has the value for j or k that appears along with V been visited? Note that whether or not j or k has been visited (together in U) depends on what the value of i is, and if i is a new value then the j or k value must be new too, by definition. Finally, if any value has already been visited, which earlier value is it?

Let $r_j = r_1^j + r_2^j + ... + r_s^j$ and $r_k = r_1^k + r_2^k + ... + r_s^k$. Similarly let $q_j = q_1^j + q_2^j + ...q_s^j$ and $q_k = q_1^k + q_2^k + ...q_s^k$. Then the number of possible answers to (a) and (b) is at most n_1^s and $n_2^{r_j}n_3^{r_k}n_2^{q_j}n_3^{q_k}$ respectively. It is also easy to see that the number of answers to (c) that arise over the sequence of ℓ steps is at most $8^\ell(s(r_j + r_k)(q_j + q_k))^\ell$. We remark that much of the work on bounding the maximum eigenvalue of a random matrix is in removing any ℓ^ℓ type terms, and so one needs to encode re-visiting indices more compactly. However such terms will only cost us polylogarithmic factors in our bound on $\|B\|$.

It is easy to see that this encoding is injective, since given the answers to the above questions one can simulate each step and recover the sequence of random variables. Next we establish some easy facts that allow us to bound $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]$.

Claim 23 *For any term that has a non-zero contribution to* $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]$*, we must have* $s \leq \ell/2$ *and* $r_j + q_j + r_k + q_k \leq \ell$

Proof Recall that there are 2ℓ random variables in the product and precisely ℓ of them correspond to U variables and ℓ of them to V variables. Suppose that $s > \ell/2$. Then there must be at least one U variable and at least one V variable that occur exactly once, which implies that its expectation is zero because the signs of the non-zero entries are mutually independent. Similarly suppose $r_j + q_j + r_k + q_k > \ell$. Then there must be at least one U or V variable that occurs exactly once, which also implies that its expectation is zero. ■

Claim 24 *For any valid encoding,* $s \leq r_j + q_j$ *and* $s \leq r_k + q_k$.

Proof This holds because in each step where the i variable is new and has not been visited before, by definition the j variable is new too (for the current i) and similarly for the k variable. ■

Finally, if s, r_j, q_j, r_k and q_k are defined as above then for any contributing term

$$U_{i_1,j_1,k_2}V_{i_1,j_2,k_1}U_{i_2,j_3,k_2}V_{i_2,j_2,k_3}...U_{i_\ell,j_1,l_\ell}V_{i_\ell,j_\ell,k_1}$$

its expectation is at most $p^{r_j+r_k}p^{q_j+q_k}$ where $p = m/n_1n_2n_3$ because there are exactly $r_j + r_k$ distinct U variables and $q_j + q_k$ distinct V variables whose values are in the set $\{-1, 0, +1\}$ and whether or not a variable is non-zero is non-positively correlated and the signs are mutually independent.

This now implies the main lemma:

Lemma 25 $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] \leq n_1^{\ell/2}(\max(n_2, n_3))^\ell p^\ell(\ell)^{3\ell+3}$

Proof Note that the indicator variables only have the effect of zeroing out some terms that could otherwise contribute to $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]$. Returning to the task at hand, we have

$$\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] \leq \sum_{s,r_j,r_k,q_j,q_k} n_1^s n_2^{r_j} n_3^{r_k} n_2^{q_j} n_3^{q_k} p^{r_j+r_k}p^{q_j+q_k}8^\ell(s(r_j + r_k)(q_j + q_k))^\ell$$

where the sum is over all valid triples s, r_j, r_k, q_j, q_k and hence $s, r, q \leq \ell/2$ and $s \leq r_j + r_k$ and $s \leq q_j + q_k$ using Claim 23 and Claim 24. We can upper bound the above as

$$\begin{aligned}\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] &\leq \sum_{s,r_j,r_k,q_j,q_k} n_1^s(pn_2)^{r_j+q_j}(pn_3)^{r_k+q_k}(\ell)^{3\ell+3} \\ &\leq \sum_{s,r_j,r_k,q_j,q_k} n_1^s(p\max(n_2, n_3))^{r_j+q_j+r_k+q_k}(\ell)^{3\ell+3}\end{aligned}$$

Now if $p\max(n_2, n_3) \leq 1$ then using Claim 24 followed by the first half of Claim 23 we have:

$$\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] \leq n_1^s(p\max(n_2, n_3))^{2s}(\ell)^{3\ell+3} \leq n_1^{\ell/2}(p\max(n_2, n_3))^\ell(\ell)^{3\ell+3}$$

where the last inequality follows because $pn_1^{1/2}\max(n_2,n_3) > 1$. Alternatively if $p\max(n_2,n_3) > 1$ then we can directly invoke the second half of Claim 23 and get:

$$\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] \leq n_1^s(p\max(n_2,n_3))^\ell(\ell)^{3\ell+3} \leq n_1^{\ell/2}(p\max(n_2,n_3))^\ell(\ell)^{3\ell+3}$$

Hence $\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)] \leq n_1^{\ell/2}\max(n_2,n_3)^\ell p^\ell(\ell)^{3\ell+3}$ and this completes the proof. ■

As before, let $n = \max(n_1,n_2,n_3)$. Then the last piece we need to bound the Rademacher complexity is the following spectral bound:

Theorem 26 *With high probability,* $\|B\| \leq O\Big(\frac{m\log^4 n}{n_1^{1/2}\min(n_2,n_3)}\Big)$

Proof We proceed by using Markov's inequality:

$$\begin{aligned}\mathbf{Pr}[\|\mathsf{B}\| \geq n_1^{1/2}\max(n_2,n_3)p(2\ell)^3] &= \mathbf{Pr}\Big[\|\mathsf{B}\|^\ell \geq \Big(n_1^{1/2}\max(n_2,n_3)p(2\ell)^3\Big)^\ell\Big] \\ &\leq \frac{\mathbf{E}[\mathrm{Tr}(\mathsf{B}\mathsf{B}^T\mathsf{B}\mathsf{B}^T...\mathsf{B}\mathsf{B}^T)]}{n_1^{\ell/2}\max(n_2,n_3)^\ell p^\ell(2\ell)^{3\ell}} \leq \frac{\ell^3}{2^{3\ell}}\end{aligned}$$

and hence setting $\ell = \Theta(\log n)$ we conclude that $\|\mathsf{B}\| \leq 8n_1^{1/2}\max(n_2,n_3)p\log^3 n$ holds with high probability. Moreover $B = \sum_{r=1}^{O(\log n)}\mathsf{B}^r$ also holds with high probability. If this equality holds and each B^r satisfies $\|\mathsf{B}^r\| \leq 8n_1^{1/2}\max(n_2,n_3)p\log^3 n$, we have

$$\|B\| \leq \max_r O(\|\mathsf{B}^r\|\log n) = O\Big(\frac{m\log^4 n}{n_1^{1/2}\min(n_2,n_3)}\Big)$$

where we have used the fact that $p = m/n_1n_2n_3$. This completes the proof of the theorem. ■

Proofs of Theorem 1 and Corollary 2

We can now bound the Rademacher complexity of the norm that we get from the six level sum-of-squares relaxation to the tensor nuclear norm:

Theorem 27 $R^m(\|\cdot\|_{\mathcal{K}_6}) \leq O\Big(\sqrt{\frac{(n_1)^{1/2}(n_2+n_3)\log^4 n}{m}}\Big)$

Proof Consider any X with $\|X\|_{\mathcal{K}_6} \leq 1$. Then using Lemma 20 and Theorem 26 we have

$$\begin{aligned}\Big(\langle Z,X\rangle\Big)^2 \leq n_1\Big(\sum_i\Big(\sum_{j,k}Z_{i,j,k}X_{i,j,k}\Big)^2\Big) &\leq C^2n_1n_2n_3\|B\| + C^6mn_1 \\ &= O\Big(mn_1^{1/2}\max(n_2,n_3)\log^4 n + mn_1\Big)\end{aligned}$$

Recall that Z was defined in Definition 8. The Rademacher complexity can now be bounded as

$$\frac{1}{m}(\langle Z,X\rangle) \leq O\Big(\sqrt{\frac{(n_1)^{1/2}(n_2+n_3)\log^4 n}{m}}\Big)$$

which completes the proof of the theorem. ■

Recall that bounds on the Rademacher complexity readily imply bounds on the generalization error (see Theorem 7). We can now prove Theorem 1:

Proof We solve (2) using the norm $\|\cdot\|_{\mathcal{K}_6}$. Since this norm comes from the sixth level of the sum-of-squares hierarchy, it follows that (2) is an n^6-sized semidefinite program and there is an efficient algorithm to solve it to arbitrary accuracy. Moreover we can always plug in $X = T - \Delta$ and the bounds on the maximum magnitude of an entry in Δ together with the Chernoff bound imply that with high probability $X = T - \Delta$ is a feasible solution. Moreover $\|T - \Delta\|_{\mathcal{K}_6} \leq r^*$. Hence with high probability, the minimizer X satisfies $\|X\|_{\mathcal{K}_6} \leq r^*$. Now if we take any such X returned by the convex program, because it is feasible its empirical error is at most 2δ. And since $\|X\|_{\mathcal{K}_6} \leq r^*$ the bounds on the Rademacher complexity (Theorem 27) together with Theorem 7 give the desired bounds on $\mathrm{err}(X)$ and complete the proof of our main theorem. ■

Finally we prove Corollary 2:

Proof Our goal is to lower bound the absolute value of a typical entry in T. To be concrete, suppose that $\mathrm{var}(T_{i,j,k}) \geq f(r,n)$ for a $1 - o(1)$ fraction of the entries where $f(r,n) = r^{1/2}/\log^D n$. Consider $T_{i,j,k}$, which we will view as a degree three polynomial in Gaussian random variables. Then the anti-concentration bounds of Carbery and Wright (2001) now imply that $|T_{i,j,k}| \geq f(r,n)/\log n$ with probability $1 - o(1)$. With this in mind, we define

$$\mathcal{R} = \{(i,j,k) \text{ s.t. } |T_{i,j,k}| \geq f(r,n)/\log n\}$$

and it follows form Markov's bound that that $|\mathcal{R}| \geq (1 - o(1))n_1n_2n_3$. Now consider just those entries in $\mathcal{R}$ which we get substantially wrong:

$$\mathcal{R}' = \{(i,j,k) \text{ s.t. } (i,j,k) \in \mathcal{R} \text{ and } |X_{i,j,k} - T_{i,j,k}| \geq 1/\log n\}$$

We can now invoke Theorem 1 which guarantees that the hypothesis X that results from solving (2) satisfies $\mathrm{err}(X) = o(1/\log n)$ with probability $1 - o(1)$ provided that $m = \widetilde{\Omega}(n^{3/2}r)$. This bound on the error immediately implies that $|\mathcal{R}'| = o(n_1n_2n_3)$ and so $|\mathcal{R} \setminus \mathcal{R}'| = (1 - o(1))n_1n_2n_3$. This completes the proof of the corollary. ■

5. Sum-of-Squares Lower Bounds

Here we will show strong lower bounds on the Rademacher complexity of the sequence of relaxations to the tensor nuclear norm that we get from the sum-of-squares hierarchy. Our lower bounds follow as a corollary from known lower bounds for refuting random instances of 3-XOR (Grigoriev, 2001; Schoenebeck, 2008). First we need to introduce the formulation of the sum-of-squares hierarchy used in Schoenebeck (2008): We will call a Boolean function f a k-junta if there is set $S \subseteq [n]$ of at most k variables so that f is determined by the values in S.

Definition 28 *The k-round Lasserre hierarchy is the following relaxation:*

(a) $\|v_0\|^2 = 1$, $\|v_C\|^2 = 1$ *for all* $C \in \mathcal{C}$

(b) $\langle v_f, v_g \rangle = \langle v_{f'}, v_{g'} \rangle$ *for all* f, g, f', g' *that are* k*-juntas and* $f \cdot g \equiv f' \cdot g'$

(c) $v_f + v_g = v_{f+g}$ *for all* f, g *that are* k*-juntas and satisfy* $f \cdot g \equiv 0$

Here we define a vector v_f for each k-junta, and $\mathcal{C}$ is a class of constraints that must be satisfied by any Boolean solution (and are necessarily k-juntas themselves). See Schoenebeck (2008) for more background, but it is easy to construct a feasible solution to the above convex program given a distribution on feasible solutions for some constraint satisfaction problem. In the above relaxation, we think of functions f as being $\{0,1\}$-valued. It will be more convenient to work with an intermediate relaxation where functions are $\{-1,1\}$-valued and the intuition is that u_S for some set $S \subseteq [n]$ should correspond to the vector for the character χ_S.

Definition 29 *Alternatively, the* k*-round Lasserre hierarchy is the following relaxation:*

(a) $\|u_\emptyset\|^2 = 1$, $\langle u_\emptyset, u_S \rangle = (-1)^{Z_S}$ *for all* $(\oplus_S, Z_S) \in \mathcal{C}$

(b) $\langle u_S, u_T \rangle = \langle u_{S'}, u_{T'} \rangle$ *for sets* S, T, S', T' *that are size at most* k *and satisfy* $S \Delta T = S' \Delta T'$*, where* Δ *is the symmetric difference.*

Here we have explicitly made the switch to XOR-constraints — namely $(\oplus_S, Z_S)$ has $Z_S \in \{0,1\}$ and correspond to the constraint that the parity on the set S is equal to Z_S. Now if we have a feasible solution to the constraints in Definition 28 where all the clauses are XOR-constraints, we can construct a feasible solution to the constraints in Definition 29 as follows. If S is a set of size at most k, we define

$$u_S \equiv v_g - v_f$$

where f is the parity function on S and $g = 1 - f$ is its complement. Moreover let $u_\emptyset = v_0$.

Claim 30 $\{u_S\}$ *is a feasible solution to the constraints in Definition 29*

Proof Consider Constraint (b) in Definition 29, and let S, T, S', T' be sets of size at most k that satisfy $S \oplus T = S' \oplus T'$. Then our goal is to show that

$$\langle v_{g_S} - v_{f_S}, v_{g_T} - v_{f_T} \rangle = \langle v_{g_{S'}} - v_{f_{S'}}, v_{g_{T'}} - v_{f_{T'}} \rangle$$

where f_S is the parity function on S, and similarly for the other functions. Then we have $f_S \cdot f_T \equiv f_{S'} \cdot f_{T'}$ because $S \oplus T = S' \oplus T'$, and this implies that $\langle v_{f_S}, v_{f_T} \rangle = \langle v_{f_{S'}}, v_{f_{T'}} \rangle$. An identical argument holds for the other terms. This implies that all the Constraints (b) hold. Similarly suppose $(\oplus_S, Z_S) \in \mathcal{C}$. Since $f_S \cdot g_S \equiv 0$ and $f_S + g_S \equiv 1$ it is well-known that (1) v_{f_S} and v_{g_S} are orthogonal (2) $v_{f_S} + v_{g_S} = v_0$ and (3) since $f_S \in \mathcal{C}$ in Definition 28, we have $v_{g_S} = 0$ (see Schoenebeck (2008)). Thus

$$\langle u_\emptyset, u_S \rangle = \langle v_0, v_{g_S} \rangle - \langle v_0, v_{f_S} \rangle = -1$$

and this completes the proof. ■

Now following Barak et al. (2012) we can use the constraints in Definition 29 to define the operator $\widetilde{\mathbf{E}}[\cdot]$. In particular, given $p \in P_k^n$ where $p \equiv \sum_S c_S \prod_{i \in S} Y_i$ and p is multilinear, we set

$$\widetilde{\mathbf{E}}[p)] = \sum_S c_S \langle u_\emptyset, u_S \rangle$$

Here we will also need to define $\widetilde{\mathbf{E}}[p]$ when p is not multilinear, and in that case if Y_i appears an even number of times we replace it with 1 and if it appears an odd number of times we replace it by Y_i to get a multilinear polynomial q and then set $\widetilde{\mathbf{E}}[p] = \widetilde{\mathbf{E}}[q]$.

Claim 31 *$\widetilde{\mathbf{E}}[\cdot]$ is a feasible solution to the constraints in Definition 18, and for any $(\oplus_S, Z_S) \in \mathcal{C}$ we have $\widetilde{\mathbf{E}}[\prod_{i \in S} Y_i] = (-1)^{Z_S}$.*

Proof Then by construction $\widetilde{\mathbf{E}}[1] = 1$, and the proof that $\widetilde{\mathbf{E}}[p^2] \geq 0$ is given in Barak et al. (2012), but we repeat it here for completeness. Let $p = \sum_S c_S \prod_{i \in S} Y_i$ be multilinear where we follow the above recipe and replace terms of the form Y_i^2 with $(1/n)$ as needed. Then $p^2 = \sum_{S,T} c_S c_T \prod_{i \in S} Y_i \prod_{i \in T} Y_i$ and moreover

$$\widetilde{\mathbf{E}}[p^2] = \sum_{S,T} c_S c_T \langle u_\emptyset, u_{S \Delta T} \rangle = \sum_{S,T} c_S c_T \langle u_S, u_T \rangle = \Big\| \sum_S c_S u_S \Big\|^2 \geq 0$$

as desired. Next we must verify that $\widetilde{\mathbf{E}}[\cdot]$ satisfies the constraints $\{\sum_{i=1}^n Y_i^2 = n\}$ and $\{Y_i^2 \leq C^2\}$ for all $i \in \{1, 2, ..., n\}$, in accordance with Definition 17. To that end, observe that

$$\widetilde{\mathbf{E}}\Big[\Big(\sum_{i=1}^n Y_i^2 - n\Big) q\Big] = 0$$

which holds for any polynomial $q \in P_{k-2}^n$. Finally consider

$$\widetilde{\mathbf{E}}\Big[\Big(C^2 - Y_i^2\Big) q^2\Big] = \widetilde{\mathbf{E}}\Big[\Big(C^2 - 1\Big) q^2\Big] \geq 0$$

which follows because $C^2 \geq 1$ and holds for any polynomial $q \in P^n_{\lfloor (d-d')/2 \rfloor}$. This completes the proof. ■

Theorem 32 *(Grigoriev, 2001; Schoenebeck, 2008) Let ϕ be a random 3-XOR formula on n variables with $m = n^{3/2-\epsilon}$ clauses. Then for any $\epsilon > 0$ and any $c < 2$, the $k = \Omega(n^{c\epsilon})$ round Lasserre hierarchy given in Definition 28 permits a feasible solution, with probability $1 - o(1)$.*

Note that the constant in the $\Omega(\cdot)$ depends on ϵ and c. Then using the above reductions, we have the following as an immediate corollary:

Corollary 33 *For any $\epsilon > 0$ and any $c < 2$ and $k = \Omega(n^{c\epsilon})$, if $m = n^{3/2-\epsilon}$ the Rademacher complexity $R^m(\| \cdot \|_{\mathcal{K}_k}) = 1 - o(1)$.*

Thus there is a sharp phase transition (as a function of the number of observations) in the Rademacher complexity of the norms derived from the sum-of-squares hierarchy. At level six, $R^m(\| \cdot \|_{\mathcal{K}_6}) = o(1)$ whenever $m = \omega(n^{3/2} \log^4 n)$. In contrast, $R^m(\| \cdot \|_{\mathcal{K}_k}) = 1 - o(1)$ when $m = n^{3/2-\epsilon}$ even for very strong relaxations derived from $n^{2\epsilon}$ rounds of the sum-of-squares hierarchy. These norms require time $2^{n^{2\epsilon}}$ to compute but still achieve essentially no better bounds on their Rademacher complexity.

Acknowledgments

We would like to thank Aram Harrow for many helpful discussions. Part of this work was done while BB was at Microsoft Research, New England. AM is supposed by NSF CAREER Award CCF-1453261, a grant from the MIT NEC Corporation and a Google Faculty Research Award.

References

Anima Anandkumar, Dean P. Foster, Daniel Hsu, Sham M. Kakade, and Yi-Kai Liu. A spectral algorithm for latent dirichlet allocation. *Algorithmica*, 72(1):193–214, 2015. doi: 10.1007/s00453-014-9909-1.

Animashree Anandkumar, Rong Ge, Daniel Hsu, and Sham Kakade. A tensor spectral approach to learning mixed membership community models. In *COLT 2013 - The 26th Annual Conference on Learning Theory, June 12-14, 2013, Princeton University, NJ, USA*, pages 867–881, 2013.

Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. *CoRR*, abs/1404.5236, 2014.

Boaz Barak, Fernando G. S. L. Brandão, Aram Wettroth Harrow, Jonathan A. Kelner, David Steurer, and Yuan Zhou. Hypercontractivity, sum-of-squares proofs, and their applications. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 307–326, 2012. doi: 10.1145/2213977.2214006.

Boaz Barak, Jonathan A. Kelner, and David Steurer. Rounding sum-of-squares relaxations. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 31–40, 2014. doi: 10.1145/2591796.2591886.

Boaz Barak, Jonathan A. Kelner, and David Steurer. Dictionary learning and tensor decomposition via the sum-of-squares method. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 143–151, 2015. doi: 10.1145/2746539.2746605.

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3:463–482, March 2003. ISSN 1532-4435.

Quentin Berthet and Philippe Rigollet. Complexity theoretic lower bounds for sparse principal component detection. In *COLT 2013 - The 26th Annual Conference on Learning Theory, June 12-14, 2013, Princeton University, NJ, USA*, pages 1046–1066, 2013.

Srinadh Bhojanapalli and Sujay Sanghavi. A new sampling technique for tensors. *CoRR*, abs/1502.05023, 2015.

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

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. doi: 10.1109/TIT.2010.2044061.

Anthony Carbery and James Wright. Distributional and l^ q norm inequalities for polynomials over convex bodies in r^ n. *Mathematical Research Letters*, 8(3):233–248, 2001.

Venkat Chandrasekaran, Benjamin Recht, Pablo A. Parrilo, and Alan S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012. doi: 10.1007/s10208-012-9135-7.

Yudong Chen, Srinadh Bhojanapalli, Sujay Sanghavi, and Rachel Ward. Coherent matrix completion. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 674–682, 2014.

Amin Coja-Oghlan, Andreas Goerdt, and André Lanka. Strong refutation heuristics for random k-sat. *Combinatorics, Probability & Computing*, 16(1):5–28, 2007. doi: 10.1017/S096354830600784X.

Amit Daniely, Nati Linial, and Shai Shalev-Shwartz. More data speeds up training time in learning halfspaces over sparse vectors. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 145–153, 2013.

M. Fazel. *Matrix Rank Minimization with Applications*. PhD thesis, Stanford University, 2002.

Uriel Feige. Relations between average case complexity and approximation complexity. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*, pages 534–543, 2002. doi: 10.1145/509907.509985.

Uriel Feige and Eran Ofek. Easily refutable subformulas of large random 3cnf formulas. *Theory of Computing*, 3(1):25–43, 2007. doi: 10.4086/toc.2007.v003a002.

Uriel Feige, Jeong Han Kim, and Eran Ofek. Witnesses for non-satisfiability of dense random 3cnf formulas. In *47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings*, pages 497–508, 2006. doi: 10.1109/FOCS.2006.78.

Joel Friedman, Jeff Kahn, and Endre Szemerédi. On the second eigenvalue in random regular graphs. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washigton, USA*, pages 587–598, 1989. doi: 10.1145/73007.73063.

Joel Friedman, Andreas Goerdt, and Michael Krivelevich. Recognizing more unsatisfiable random k-sat instances efficiently. *SIAM J. Comput.*, 35(2):408–430, 2005. doi: 10.1137/S009753970444096X.

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

Rong Ge and Tengyu Ma. Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2015, August 24-26, 2015, Princeton, NJ, USA*, pages 829–849, 2015. doi: 10.4230/LIPIcs.APPROX-RANDOM.2015.829.

Andreas Goerdt and Michael Krivelevich. Efficient recognition of random unsatisfiable k-sat instances by spectral methods. In *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, Dresden, Germany, February 15-17, 2001, Proceedings*, pages 294–304, 2001. doi: 10.1007/3-540-44693-1_26.

Dima Grigoriev. Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. *Theor. Comput. Sci.*, 259(1-2):613–622, 2001. doi: 10.1016/S0304-3975(00)00157-2.

Leonid Gurvits. Classical deterministic complexity of edmonds' problem and quantum entanglement. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, June 9-11, 2003, San Diego, CA, USA*, pages 10–19, 2003. doi: 10.1145/780542.780545.

Moritz Hardt. Understanding alternating minimization for matrix completion. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 651–660, 2014. doi: 10.1109/FOCS.2014.75.

Aram Wettroth Harrow and Ashley Montanaro. Testing product states, quantum merlin-arthur games and tensor optimization. *J. ACM*, 60(1):3, 2013. doi: 10.1145/2432622.2432625.

Samuel B Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer. Speeding up sum-of-squares for tensor decomposition and planted sparse vectors. *arXiv preprint arXiv:1512.02337*, 2015.

Daniel Hsu and Sham M. Kakade. Learning mixtures of spherical gaussians: moment methods and spectral decompositions. In *Innovations in Theoretical Computer Science, ITCS '13, Berkeley, CA, USA, January 9-12, 2013*, pages 11–20, 2013. doi: 10.1145/2422436.2422439.

Prateek Jain and Sewoong Oh. Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 1431–1439, 2014.

Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 665–674, 2013. doi: 10.1145/2488608.2488693.

Raghunandan H. Keshavan, Andrea Montanari, and Sewoong Oh. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11:2057–2078, 2010.

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

Jean B Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001.

Troy Lee and Adi Shraibman. Matrix completion from any given set of observations. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 1781–1787, 2013.

Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):208–220, 2013.

Jiří Matoušek. *Lectures on discrete geometry*, volume 212. Springer New York, 2002.

Elchanan Mossel and Sébastien Roch. Learning nonsingular phylogenies and hidden markov models. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 366–375, 2005. doi: 10.1145/1060590.1060645.

Cun Mu, Bo Huang, John Wright, and Donald Goldfarb. Square deal: Lower bounds and improved relaxations for tensor recovery. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 73–81, 2014.

Yurii Nesterov. Squared functional systems and optimization problems. In *High performance optimization*, pages 405–440. Springer, 2000.

Pablo A Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, 2000.

Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010. doi: 10.1137/070697835.

Grant Schoenebeck. Linear level lasserre lower bounds for certain k-csps. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 593–602, 2008. doi: 10.1109/FOCS.2008.74.

Naum Zuselevich Shor. An approach to obtaining global extremums in polynomial mathematical programming problems. *Cybernetics*, 23(5):695–700, 1988.

Marco Signoretto, Lieven De Lathauwer, and Johan AK Suykens. Nuclear norms for tensors and their use for convex multilinear estimation. *Tech Report 10-186, K. U. Leuven*, 2010.

Nathan Srebro and Adi Shraibman. Rank, trace-norm and max-norm. In *Learning Theory, 18th Annual Conference on Learning Theory, COLT 2005, Bertinoro, Italy, June 27-30, 2005, Proceedings*, pages 545–560, 2005. doi: 10.1007/11503415_37.

Gongguo Tang, Badri Narayan Bhaskar, Parikshit Shah, and Benjamin Recht. Compressed sensing off the grid. *IEEE Transactions on Information Theory*, 59(11):7465–7490, 2013. doi: 10.1109/TIT.2013.2277451.

Ming Yuan and Cun-Hui Zhang. On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*, pages 1–38, 2014.

Appendix A. Reduction from Asymmetric to Symmetric Tensors

Here we give a general reduction, and show that any algorithm for tensor prediction that works for symmetric tensors can be used to predict the entries of an asymmetric tensor too. Hardt gave a related reduction for the cases of matrices (Hardt, 2014) and it is instructive to first understand this reduction, before proceeding to the tensor case. Suppose we are given a matrix M that is not

necessarily symmetric. Then the approach of Hardt (2014) is to construct the following symmetric matrix:

$$S = \begin{bmatrix} 0 & M^T \\ M & 0 \end{bmatrix}.$$

We have not precisely defined the notion of incoherence that is used in the matrix completion literature, but it turns out to be easy to see that S is low rank and incoherent as well.

The important point is that given m samples generated uniformly at random from M, we can generate random samples from S too. It will be more convenient to think of these random samples as being generated without replacement, but this reduction works just as well without replacement too. Let $M \in \mathbb{R}^{n_1 \times n_2}$. Now for each sample from S, with probability $p = \frac{n_1^2+n_2^2}{(n_1+n_2)^2}$ we reveal a uniformly random entry in the either block of zeros. And with probability $1-p$ we reveal a uniformly random entry from M. Each entry in M appears exactly twice in S, and we choose to reveal this entry of M with probability $1/2$ from the top-right block, and otherwise from the bottom-left block. Thus given m samples from M, we can generate from S (in fact we can generate even more, because some of the revealed entries will be zeros). It is easy to see that this approach works for the case of sampling without replacement to, in that m samples without replacement from M can be used to generate at least m samples without replacement from S.

Now let us proceed to the tensor case. Let us introduce the following definition, for ease of notation:

Definition 34 *Let $m(n, r, \epsilon, f, C)$ be such that, there is an algorithm that on a rank r, order d, size $n \times n \times ... \times n$ symmetric tensor where each factor has norm at most C, the algorithm returns an estimate X with $err(X) = f$ with probability $1-\epsilon$ when it is given $m(n, r, \epsilon, f)$ samples chosen uniformly at random (and without replacement).*

Lemma 35 *For any odd d, suppose we are given $m(\sum_{j=1}^d n_j, r2^{d-1}, \epsilon, f, \sqrt{d})$ samples chosen uniformly at random (and without replacement) from an $n_1 \times n_2 \times ... \times n_d$ tensor*

$$T = \sum_{i=1}^r a_i^1 \otimes a_i^2 \otimes ... \otimes a_i^d$$

where each factor is unit norm. There is an algorithm that with probability at least $1-\epsilon$ returns an estimate Y with

$$err(Y) \leq \frac{(\sum_{j=1}^d n_j)^d}{d! 2^{d-1} \prod_{j=1}^d n_j} f$$

Proof Our goal is to symmetrize an asymmetric tensor, and in such a way that each entry in the symmetrized tensor is either zero or else corresponds to an entry in the original tensor. Our reduction will work for any odd order d tensor. In particular let

$$T = \sum_{i=1}^r a_i^1 \otimes a_i^2 \otimes ... \otimes a_i^d$$

be an order d tensor where the dimension of a^j is n_j. Also let $n = \sum_{j=1}^d n_j$. Then we will construct a symmetric, order d tensor as follows. Let $\sigma_1, \sigma_2, ...\sigma_d$ be a collection of d random $\pm$ variables

that are chosen uniformly at random from the 2^{d-1} configurations where $\prod_{j=1}^{d} \sigma_j = 1$. Then we consider the following random vector

$$a_i(\sigma_1, \sigma_2, ...\sigma_d) = [\sigma_1 a_i^1, \sigma_2 a_i^2, ..., \sigma_d a_i^d]$$

Here $a_i(\sigma_1, \sigma_2, ...\sigma_d)$ is an n-dimensional vector that results from concatenating the vectors $a_i^1, a_i^2, ..., a_i^d$ but after flipping some of their signs according to $\sigma_1, \sigma_2, ...\sigma_d$. Then we set

$$S = \mathop{\mathbf{E}}_{\sigma_1,\sigma_2,...\sigma_d}[\sum_{i=1}^{r} \Big(a_i(\sigma_1, \sigma_2, ...\sigma_d)\Big)^{\otimes d}]$$

It is immediate that S is symmetric and has rank at most $2^{d-1}r$ by expanding out the expectation into a sum over the valid sign configurations. Moreover each rank one term in the decomposition is of the form $a^{\otimes d}$ where $\|a\|_2^2 = d$ because it is the concatenation of d unit vectors.

If $\sigma_1, \sigma_2, ...\sigma_d$ is fixed, then each entry in S is itself a degree d polynomial in the σ_j variables. By our construction of the σ_j variables, and because d is odd so there are no terms where every variable appears to an even power, it follows that all the terms vanish in expectation except for the terms which have a factor of $\prod_{j=1}^{d} \sigma_j$, and these are exactly terms that correspond to some permutation $\pi : [d] \to [d]$, and a term of the form

$$\sum_{i=1}^{d} a_i^{\pi(1)} \otimes a_i^{\pi(2)} \otimes, ..., \otimes a_i^{\pi(d)}$$

Hence all of the entries in S are either zero or are 2^{d-1} times an entry in T. As before, we can generate m uniformly random samples from S given m uniformly random samples from T, by simply choosing to sample an entry from one of the blocks of zeros with the appropriate probability, or else revealing an entry of T and choosing where in S to reveal this entry uniformly at random. Hence:

$$\frac{1}{(\sum_{j=1}^{d} n_j)^d} \sum_{(i_1,i_2,...,i_d)\in\Gamma} |Y_{i_1,i_2,...,i_d} - S_{i_1,i_2,...,i_d}| \leq \frac{1}{(\sum_{j=1}^{d} n_j)^d} \sum_{i_1,i_2,...,i_d} |Y_{i_1,i_2,...,i_d} - S_{i_1,i_2,...,i_d}|$$

where Γ represents the locations in S where an entry of T appears. The right hand side above is at most f with probability $1 - \epsilon$. Moreover each entry in T appears in exactly $d!$ locations in S. And when it does appear, it is scaled by 2^{d-1}. And hence if we multiply the left hand side by

$$\frac{(\sum_{j=1}^{d} n_j)^d}{d! 2^{d-1} \prod_{j=1}^{d} n_j}$$

we obtain $\text{err}(Y)$. This completes the reduction. ■

Note that in the case where $n_1 = n_2 = n_3 ... = n_d$, the error and the rank in this reduction increase only by at most an e^d and 2^d factor respectively.