Proceedings of Machine Learning Research vol 99:1–40, 2019 32nd Annual Conference on Learning Theory

A Robust Spectral Algorithm for Overcomplete Tensor Decomposition

Samuel B. Hopkins HOPKINS@BERKELEY.EDU
Cornell University and UC Berkeley

Tselil Schramm TSELIL@MIT.EDU
MIT and Harvard

Jonathan Shi JSHI@CS.CORNELL.EDU
Cornell University

Editors: Alina Beygelzimer and Daniel Hsu

Abstract

We give a spectral algorithm for decomposing overcomplete order-4 tensors, so long as their components satisfy an algebraic non-degeneracy condition that holds for nearly all (all but an algebraic set of measure 0) tensors over $(\mathbb{R}^d)^{\otimes 4}$ with rank $n \leqslant d^2$. Our algorithm is robust to adversarial perturbations of bounded spectral norm.

Our algorithm is inspired by one which uses the sum-of-squares semidefinite programming hierarchy (Ma, Shi, and Steurer STOC'16), and we achieve comparable robustness and overcompleteness guarantees under similar algebraic assumptions. However, our algorithm avoids semidefinite programming and may be implemented as a series of basic linear-algebraic operations. We consequently obtain a much faster running time than semidefinite programming methods: our algorithm runs in time $\tilde{O}(n^2 d^3) \leqslant \tilde{O}(d^7)$, which is subquadratic in the input size d^4 (where we have suppressed factors related to the condition number of the input tensor).

Keywords: overcomplete tensors, tensor rank decomposition, CP decomposition, spectral algorithms, algebraic non-degeneracy, sum-of-squares proofs

1. Introduction

Tensors are higher-order analogues of matrices: multidimensional arrays of numbers. They have broad expressive power: tensors may represent higher-order moments of a probability distribution Anandkumar et al. (2014b), they are natural representations of cubic, quartic, and higher-degree polynomials Richard and Montanari (2014); Hopkins et al. (2015), and they appear whenever data is multimodal (e.g. in medical studies, where many factors are measured) Acar et al. (2007); Beckmann and Smith (2005); Hai-Long et al.. Due to these reasons, in recent decades tensors have emerged as fundamental structures in machine learning and signal processing.

The notion of *rank* extends from matrices to tensors: a rank-1 tensor in $(\mathbb{R}^d)^{\otimes k}$ is a tensor that can be written as a tensor product $u^{(1)} \otimes \cdots \otimes u^{(k)}$ of vectors $u^{(1)}, \ldots, u^{(k)} \in \mathbb{R}^d$. Any tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes k}$ can be expressed as a sum of rank-1 tensors, and the *rank* of $\mathbf{T}$ is the minimum number of terms needed in such a sum. As is the case for matrices, we are often interested in tensors of low rank: low-rank structure in tensors often carries interpretable

© 2019 S.B. Hopkins, T. Schramm & J. Shi.

meaning about underlying data sets or probability distributions, and the tensors that arise in many applications are low-rank Anandkumar et al. (2014b).

Tensor decomposition is the natural inverse problem in the context of tensor rank: given a d-dimensional symmetric k-tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes k}$ of the form

$$\mathbf{T} = \sum_{i \leqslant n} a_i^{\otimes k} + \mathbf{E},$$

for vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ and an (optional) error tensor $\mathbf{E} \in (\mathbb{R}^d)^{\otimes k}$, we are asked to output vectors $b_1, \ldots, b_n$ as close as possible to $a_1, \ldots, a_n$ (e.g. minimizing the Euclidean distance $\|b_i - a_i\|$). The goal is to accomplish this with an algorithm that is as efficient as possible, under the mildest-possible assumptions on k,$a_1, \ldots, a_n$, and $\mathbf{E}$.

While tensor rank decomposition is a generalization of rank decomposition for matrices, decomposition for tensors of order $k \geqslant 3$ differs from the matrix case in several key ways.

1. (Uniqueness) Under mild assumptions on the vectors $a_1, \ldots, a_n$, tensor decompositions are unique (up to permutations of $[n]$), while matrix decompositions are often unique only up to unitary transformation.

2. (Overcompleteness) Tensor decompositions often remain unique even when the number of factors n is larger than the ambient dimension d (up to $n = O(d^{k-1})$), while a $d \times d$ matrix can have only d eigenvectors or $2d$ singular vectors.

These features make tensor decompositions suitable for many applications where matrix factorizations are insufficient. However, there is another major difference:

3. (Computational Intractability) While many matrix decompositions — eigendecompositions, singular value decompositions, LU-factorizations, and so on — can be found in polynomial time, tensor decomposition is NP-hard in general Hillar and Lim (2013).

In spite of the NP-hardness of general tensor decomposition, many special cases admit polynomial-time algorithms. A classic algorithm, often called *Jennrich's algorithm*, recovers the components $a_1, \ldots, a_n$ from $\mathbf{T}$ when they are linearly independent (which requires $n \leqslant d$) and $\mathbf{E} \approx 0$ using simultaneous diagonalization Harshman (1970); De Lathauwer et al. (1996).

More sophisticated algorithms improve on Jennrich's in their tolerance to overcompleteness (and the resulting lack of linear independence) and robustness to nontrivial error tensors $\mathbf{E}$. The literature now contains a wide variety of techniques for tensor decomposition: the major players are iterative methods (tensor power iteration, stochastic gradient descent, and alternating minimization), spectral algorithms, and convex programs. Convex programs, and in particular the *sum-of-squares* semidefinite programming hierarchy (SoS), require the mildest assumptions on $k, a_1, \ldots, a_n, \mathbf{E}$ among known polynomial-time algorithms Ma et al. (2016). In pushing the boundaries of what is known to be achievable in polynomial time, SoS-based algorithms have been crucial. However, the running times of these algorithms are large polynomials in the input, making them utterly impractical for applications.

The main contribution of this work is a tensor decomposition algorithm whose robustness to errors and tolerance for overcompleteness are similar to those of the SoS-based algorithms, but with *subquadratic running time*. Other algorithms with comparable running times require

either higher-order tensors,[1] an exponentially small input error $\mathbf{E}$ (and hence are not robust), or linear independence of the components $a_1, \ldots, a_n$ and hence $n \leqslant d$.[2]

Our algorithm is comparatively simple, and can be implemented with a small number of dense matrix and matrix-vector multiplication operations, which are fast not only asymptotically but also in practice.

Concretely, we study tensor decomposition of overcomplete 4-tensors under algebraic nondegeneracy conditions on the tensor components $a_1, \ldots, a_n$. Algebraic conditions like ours are the mildest type of assumption on $a_1, \ldots, a_n$ known to lead to polynomial time algorithms – our algorithm can decompose all but a measure-zero set of 4-tensors of rank $n \ll d^2$, and in particular we make no assumption that the components $a_1, \ldots, a_n$ are random.[3]

When $n \ll d^2$, our algorithm approximately recovers a $1 - o(1)$ fraction of $a_1, \ldots, a_n$ (up to their signs) from $T = \sum_{i \leqslant n} a_i^{\otimes 4} + E$, so long as the spectral norm $\|E\|$ is (significantly) less than the minimum singular value of a certain matrix associated to the $\{a_i\}$. (In particular, nonsingularity of this matrix is our nondegeneracy condition on $a_1, \ldots, a_n$.) The algorithm requires time $\tilde{O}(n^2 d^3) \leqslant O(d^7)$, which is subquadratic in the input size d^4.

Robustness, Overcompleteness, and Applications to Machine Learning Tensor decomposition is a common primitive in algorithms for statistical inference that leverage the *method of moments* to learn parameters of latent variable models. Examples of such algorithms exist for independent component analysis / blind source separation De Lathauwer et al. (2007), dictionary learning Barak et al. (2015); Ma et al. (2016); Schramm and Steurer (2017), overlapping community detection Anandkumar et al. (2013); Hopkins and Steurer (2017), mixtures of Gaussians Ge et al. (2015b), and more.

In these applications, we receive samples $x \in \mathbb{R}^d$ from a model distribution $\mathcal{D}(\rho)$ that is a function of parameters ρ. The goal is to estimate ρ using the samples. The method-of-moments strategy is to construct the third- or fourth-order moment tensor $\mathbb{E}\, x^{\otimes k}$ $(k = 3, 4)$ from samples whose expectation $\mathbb{E}\, x^{\otimes k} = \sum_{i \leqslant n} a_i^{\otimes k}$ is a low rank tensor with components $a_1, \ldots, a_n$, from which the model parameters ρ can be deduced.[4] Since $\mathbb{E}\, x^{\otimes k}$ is estimated using samples, the tensor decomposition algorithm used to extract $a_1, \ldots, a_n$ from $\mathbb{E}\, x^{\otimes k}$ must be *robust* to error from sampling. The sample complexity of the resulting algorithm depends directly on the magnitude of errors tolerated by the decomposition algorithm.

Some model classes give rise to *overcomplete* tensors; roughly speaking, this occurs when the number of parameters (the size of the description of ρ) far exceeds d^2, where d is the ambient dimension. Typically, in such cases, ρ consists of a collection of vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ with $n \gg d$. Such overcomplete models are widely used; for example, in the *dictionary learning* setting, we are given a data set S and are asked to find a *sparse representation* of S. This is a powerful preprocessing tool, and the resulting representations

1. Higher-order tensors are costly because they are larger objects, and for learning applications they often require a polynomial increase in sample complexity.
2. There are also existing robust algorithms which tolerate some overcompleteness when $a_1, \ldots, a_n$ are assumed to be random; in this paper we study *generic* $a_1, \ldots, a_n$, which is a much more challenging setting than random $a_1, \ldots, a_n$ Anandkumar et al. (2014a); Hopkins et al. (2016).
3. Although decompositions of 4-th order tensors can remain unique up to $n \approx d^3$, no polynomial-time algorithms are known which successfully decompose tensors of overcompleteness $n \gg d^2$.
4. Any constant k, rather than just $k = 3, 4$, may lead to polynomial-time learning algorithms, but the cost is typically gigantic polynomial sample complexity and running time, scaling like d^k, to estimate and store a k-th order tensor.

are more robust to perturbations, but assembling a truly sparse, effective dictionary often requires representing d-dimensional data in a basis with $n \gg d$ elements Lewicki and Sejnowski (2000); Elad (2010). Recent works also relate the problem of learning neural networks with good generalization error to tensor decomposition, showing a connection between overcompleteness and the width of the network Mondelli and Montanari (2018).[5]

Using tensor decomposition in such settings requires algorithms with practical running times, error robustness, and tolerance to overcompleteness. The strongest polynomial-time guarantees for overcomplete dictionary learning and similar models currently rely on overcomplete tensor decomposition via the SoS method Ma et al. (2016); our work is an important step towards giving lightweight, spectral algorithms for such problems.

1.1. Our Results

Our contribution is a robust, lightweight spectral algorithm for tensor decomposition in the overcomplete regime. We require that the components satisfy an algebraic non-degeneracy assumption satisfied by all but a measure-0 set of inputs. At a high level, we require that a certain matrix associated with the components of the tensor have full rank. Though the assumption may at first seem complicated, we give it formally here:

Definition 1 *Let $\Pi_{2,3}^{\perp}$ be the projector to the orthogonal complement of the subspace of $(\mathbb{R}^d)^{\otimes 3}$ that is symmetric in its latter two tensor modes. Equivalently, $\Pi_{2,3}^{\perp} = \frac{1}{2}(\mathrm{Id} - P_{2,3})$, where $P_{2,3}$ is the linear operator that interchanges the second and third modes of $(\mathbb{R}^d)^{\otimes 3}$.*

Definition 2 *Let $\Pi_{\mathrm{img}(M)}$ denote the projector to the column space of the matrix M. Equivalently, $\Pi_{\mathrm{img}(M)} = (MM^\top)^{-1/2}M = M(M^\top M)^{-1/2}$, where $(MM^\top)^{-1/2}$ is the* whitening *transform of M and is equal to the Moore-Penrose pseudoinverse of $(MM^\top)^{1/2}$.*

Definition 3 *Vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ are κ-*non-degenerate *if the matrix $K(a_1, \ldots, a_n)$, defined below, has minimum singular value at least $\kappa > 0$. If $\kappa = 0$, we say that the $\{a_i\}$ are* degenerate.

The matrix $K(a_1, \ldots, a_n)$ is given by choosing for each $i \in [n]$ a matrix B_i whose columns form a basis for the orthogonal complement of a_i in $\mathbb{R}^d$, assembling the $d^3 \times n(d-1)$ matrix H whose rows are given by $a_i \otimes a_i \otimes B_i^{(j)}$ as

$$H = \begin{bmatrix} a_1^\top \otimes a_1^\top \otimes B_1^\top \\ \vdots \\ a_n^\top \otimes a_n^\top \otimes B_n^\top \end{bmatrix}$$

and letting $K(a_1, \ldots, a_n) = \Pi_{2,3}^{\perp}\Pi_{\mathrm{img}(H^\top)}$.

We note that when $n \ll d^2$ then all but a measure-zero set of unit $(a_1, \ldots, a_n) \in \mathbb{R}^{dn}$ satisfy the condition that $\kappa > 0$. We expect also that for $n \ll d^2$, if $a_1, \ldots, a_n \in \mathbb{R}^d$ are

5. Strictly speaking, this work shows a reduction *from* tensor decomposition *to* learning neural nets, but the connection between width and overcompleteness is direct regardless.

independent uniformly random unit vectors then $\kappa \geqslant \Omega(1)$ – we provide simulations in support of this, and for a variety of other families of random vectors, in Appendix H.[6]

Some previous works on tensor decomposition under algebraic nondegeneracy assumptions also give smoothed analyses of nondegeneracy, showing that small random perturbations of arbitrary vectors are $\frac{1}{\mathrm{poly}(d)}$-well-conditioned (for differing notions of well-conditioned-ness) Bhaskara et al. (2014); Ma et al. (2016). We expect that a similar smoothed analysis is possible for κ-non-degeneracy, though because of the specific form of the matrix $K(a_1, \ldots, a_n)$ it does not follow immediately from known results. We defer this to future work.

Given this non-degeneracy condition, we robustly decompose the input tensor in time $\tilde{O}(\frac{n^2 d^3}{\kappa})$, where we have suppressed factors depending on the smallest singular value of a matrix flattening of our tensor.

Theorem *[Special case of Theorem 25] Suppose that $d \leqslant n \leqslant d^2$, and that $a_1, \ldots, a_n \in \mathbb{R}^d$ are κ-non-degenerate unit vectors for $\kappa > 0$, and suppose that $\mathbf{T}$ is their 4-tensor perturbed by noise, $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 4}$ such that $\mathbf{T} = \sum_{i \in [n]} a_i^{\otimes 4} + E$, where E is a perturbation such that $\|E\| \leqslant \frac{\varepsilon}{\log d}$ in its $d^2 \times d^2$ reshaping. Suppose further that when reshaped to a $d^2 \times d^2$ matrix, $\|T^{-1}\| \leqslant O(1)$ and that $\| \sum_{i \in [n]} (a_i^{\otimes 3})(a_i^{\otimes 3})^\top \| \leqslant O(1)$.*

There exists an algorithm DECOMPOSE *with running time $\tilde{O}(n^2 d^3 \kappa^{-1})$, so that for every such $\mathbf{T}$ there exists a subset $S \subseteq \{a_1, \ldots, a_n\}$ of size $|S| \geqslant 0.99n$, such that* DECOMPOSE($\mathbf{T}$) *with high probability returns a set of $t = \tilde{O}(n)$ unit vectors $b_1, \ldots, b_t$ where every $a_i \in S$ is close to some b_j, and each b_j is close to some $a_i \in S$:*

$$\forall a_i \in S,\ \max_j |\langle b_j, a_i \rangle| \geqslant 1 - O\left(\frac{\varepsilon}{\kappa^2}\right)^{1/8}, \qquad \text{and} \qquad \forall j \in [t],\ \max_{a_i \in S} |\langle b_j, a_i \rangle| \geqslant 1 - O\left(\frac{\varepsilon}{\kappa^2}\right)^{1/8}.$$

Furthermore, if $a_1, \ldots, a_n$ are random unit vectors, then with high probability they satisfy the conditions of this theorem with $\kappa = \Omega(1)$.

When $n \leqslant d$, our algorithm still obtains nontrivial guarantees (though the runtime asymptotics are dominated by other terms); however in this regime, a combination of the simpler algorithm of Schramm and Steurer (2017) and a whitening procedure gives comparable guarantees.

We remark that our full theorem, Theorem 25, does not pose as many restrictions on the $\{a_i\}$; we do not generally require that $\|T^{-1}\| \leqslant O(1)$ or that $\| \sum_i (a_i^{\otimes 3})(a_i^{\otimes 3})^\top \| \leqslant O(1)$. However, allowing these quantities to depend on d and n affects our runtime and approximation guarantees, and so to simplify presentation we have made these restrictions here; we refer the reader to Theorem 25 for details.

Furthermore, in the theorem stated above we recover only a 0.99-fraction of the vectors, and we require the perturbation to have magnitude $O(\frac{1}{\log d})$. This is again a particular choice of parameters in Theorem 25, which allows for a four-way tradeoff among accuracy, magnitude of perturbation, fraction of components recovered, and runtime. For example, if the perturbation is $\frac{1}{\mathrm{poly}(d)}$ in spectral norm, then we may recover all components in time

6. Furthermore, standard techniques in random matrix theory prove that when $a_1, \ldots, a_n$ are random then matrices closely related to $K(a_1, \ldots, a_n)$ are well-conditioned; for instance this holds (roughly speaking) if $(H_1^\top H)^{-1/2}$ and $(H_2^\top H)^{-1/2}$ are removed. However, inverses and pseudoinverses of random matrices, especially those with dependent entries like ours, are infamously challenging to analyze – we leave this challenge to future work. See Appendix H for details.

Algorithm	Type	Rank	Robustness	Assumptions	Runtime
Lathauwer et al. (2007)	algebraic	$n \leqslant d^2$	$\|E\|_\infty \leqslant 2^{-O(d)}$	algebraic	$\tilde{O}(n^3 d^4)$
Anandkumar et al. (2017)	iterative	$n \leqslant o(d^{1.5})$	$\|E\| \leqslant o(\frac{n}{d^2})$	random, warm start	$\tilde{O}(nd^3)$
Ge and Ma (2017)	iterative	$n \leqslant O(d^2)$	$E = 0$	random, warm start	$\tilde{O}(nd^4)$
Ma et al. (2016)	SDP	$n \leqslant d^2$	$\|E\| \leqslant 0.01$	algebraic	$\geqslant nd^{24}$
Schramm and Steurer (2017)	spectral	$n \leqslant d$	$\|E\| \leqslant O(\frac{\log\log d}{\log d})$	orthogonal	$\tilde{O}(d^{2+\omega})$
this paper	spectral	$n \leqslant d^2$	$\|E\| \leqslant O(\frac{1}{\log d})$	algebraic	$\tilde{O}(n^2 d^3)$

Table 1: A comparison of tensor decomposition algorithms for rank-n 4-tensors in $(\mathbb{R}^d)^{\otimes 4}$. Here ω denotes the matrix multiplication constant. A robustness bound $\|E\| \leqslant \eta$ refers to the requirement that a $d^2 \times d^2$ reshaping of the error tensor E have spectral norm at most η. Some of the algorithms' guarantees involve a tradeoff between robustness, runtime, and assumptions; where this is the case, we have chosen one representative setting of parameters. See Section G for details. Above, "random" indicates that the algorithm assumes $a_1, \ldots, a_n$ are independent unit vectors (or Gaussians) and "algebraic" indicates that the algorithm assumes that the vectors avoid an algebraic set of measure 0.

$\tilde{O}(n^2 d^3 \kappa^{-1})$; alternatively, if the perturbation has spectral norm $\eta^2 = \Theta(1)$, then we may recover an 0.99-fraction of components in time $\tilde{O}(n^{2+O(\eta)} d^3 \kappa^{-1})$ up to accuracy $1 - O(\frac{\eta}{\kappa^2})^{1/8}$. Again, we refer the reader to Theorem 25 for the full tradeoff.

Finally, a note about our recovery guarantee: we guarantee that every vector returned by the algorithm is close to *some* component, and furthermore that most components will be close to some vector. It is possible to run a clean-up procedure after our algorithm, in which nearby approximate components b_j are clustered to correspond to a specific a_i; depending on the proximity of the a_i to each other, this may require stronger accuracy guarantees, and so we leave this procedure as an independent step. Our guarantee does not include signs, but this is because the tensor $\mathbf{T}$ is an even-order tensor, so the decomposition is only unique up to signings as $(-a_i)^{\otimes 4} = a_i^{\otimes 4}$.

1.2. Related works

The literature on tensor decomposition is broad and varied, and we will not attempt to survey it fully here (see e.g. the survey Kolda and Bader (2009) or the references within Anandkumar et al. (2014b); Ge and Ma (2017) for a fuller picture). We will give an idea of the relationship between our algorithm and others with provable guarantees.

For simplicity, we focus on order-4 tensors. Algorithms with provable guarantees for tensor decomposition fall broadly into three classes: iterative methods, convex programs, and spectral algorithms. For a brief comparison to previous works, we include Table 1.

Iterative Methods. Iterative methods are a class of algorithms that maintain one (or sometimes several) estimated component(s) b, and update the estimate using a variety of update rules. Some popular update rules include tensor power iteration Anandkumar et al. (2014b), gradient descent Ge and Ma (2017), and alternating-minimization Anandkumar

et al. (2014c). Most of these methods have the advantage that they are fast; the update steps usually run in time linear in the input size, and the number of updates to convergence is often polylogarithmic in the input size.

The performance of the most popular iterative methods has been well-characterized in some restricted settings; for example, when the components $\{a_i\}$ are orthogonal or linearly independent Anandkumar et al. (2014b); Ge et al. (2015a); Sharan and Valiant (2017), or are independently drawn random vectors Anandkumar et al. (2017); Ge and Ma (2017). Many of these analyses require a "warm start," or an initial estimate b that is more correlated with a component than a typical random starting point. Few provable guarantees are known for the non-random overcomplete regime, or in the presence of arbitrary perturbations.

Convex Programming. Convex programs based on the sum-of-squares (SoS) semidefinite programming (SDP) relaxation yield the most general provable guarantees for tensor decomposition. These works broadly follow a *method of pseudo-moments*: interpreting the input tensor $\sum_{i\in[n]} a_i^{\otimes k}$ as the k-th moment tensor $\mathbb{E}\, X^{\otimes k}$ of a distribution X on $\mathbb{R}^d$, this approach uses SoS to generate *surrogates* (or *pseudo-moments*) for higher moment tensors, like $\mathbb{E}\, X^{\otimes 100k} = \sum_{i\in[n]} a_i^{\otimes 100k}$. It is generally easier to extract the components $a_1, \ldots, a_n$ from $\sum_{i\in[n]} (a_i^{\otimes 100})^{\otimes k}$ than from $\sum_{i\in[n]} a_i^{\otimes k}$, because the vectors $\{a_i^{\otimes 100}\}$ have fewer algebraic dependencies than the vectors $\{a_i\}$, and are farther apart in Euclidean distance. Of course, $\mathbb{E}\, X^{\otimes 100k} = \sum_{i\in[n]} a_i^{\otimes 100k}$ is not given as input, and even in applications where the input is negotiable, it may be expensive or impossible to obtain such a high-order tensor. The SoS method uses semidefinite programming to generate a surrogate which is good enough to be used to find the vectors $a_1, \ldots, a_n$

Work on sum-of-squares relaxations for tensor decomposition began with the quasi-polynomial time algorithm of Barak et al. (2015); this algorithm requires only mild well-conditioned-ness assumptions, but also requires high-order tensors as input, and runs in quasi-polynomial time. This was followed by an analysis showing that, at least in the setting of random $a_1, \ldots, a_n$, the SoS algorithm can decompose substantially overcomplete tensors of order 3 Ge and Ma (2015). This line of work finally concluded with the work of Ma, Shi, and Steurer Ma et al. (2016), who give sum-of-squares based polynomial-time algorithms for tensor decomposition in the most general known settings: under mild algebraic assumptions on the components, and in the presence of adversarial noise, so long as the noise tensor has bounded spectral norm in its matrix reshapings.

These SoS algorithms have the best known polynomial-time guarantees, but they are formidably slow. The work of Ma et al. (2016) uses the degree-8 sum-of-squares relaxation, meaning that to find each of the n components, one must solve an SDP in $\Omega(d^8)$ variables. While these results are important in establishing that polynomial-time algorithms *exist* for these settings, their runtimes are far from efficient.

Spectral algorithms from Sum-of-Squares Analyses. Inspired by the mild assumptions needed by SoS algorithms, a line of work has used the *analyses* of SoS in order to design more efficient *spectral* algorithms, which ideally work for similarly broad classes of tensors.

At a high level, these spectral algorithms use eigendecompositions of specific matrix polynomials to directly construct approximate primal and dual solutions to the SoS semidefinite programs, thereby obtaining the previously mentioned "surrogate moments" without having to solve an SDP. Since the SoS SDPs are quite powerful, constructing (even approximate)

solutions to them directly and efficiently is a nontrivial endeavor. The resulting matrices are only approximately SDP solutions — in fact, they are often far from satisfying most of the constraints of the SoS SDPs. There is a tradeoff between how well these spectrally constructed solutions approximate the SoS output and how efficiently the algorithm can be implemented. However, by carefully choosing which constraints to satisfy, these works are able to apply the SDP rounding algorithms to the approximate spectrally-constructed solutions (often with new analyses) to obtain similar algorithmic guarantees.

The work of Hopkins et al. (2016) was the first to adapt the analysis of SoS for random $a_1, \ldots, a_n$ presented by Ge and Ma (2015) to obtain spectral algorithms for tensor decomposition, giving subquadratic algorithms for decomposing random overcomplete tensors with $n \leqslant O(d^{4/3})$. As SoS algorithms have developed, so too have their faster spectral counterparts. In particular, Schramm and Steurer (2017) adapted some of the SoS arguments presented in Ma et al. (2016) to give robust subquadratic algorithms for decomposing orthogonal 4-tensors in the presence of *adversarial* noise bounded only in spectral norm.

Our result builds on the progress of both Ma et al. (2016); Schramm and Steurer (2017). The SoS algorithm of Ma et al. (2016) was the first to robustly decompose generic overcomplete tensors in polynomial time. The spectral algorithm of Schramm and Steurer (2017) obtains a much faster running time for robust tensor decomposition, but sacrifices overcompleteness. Our work adapts (and improves upon) the SoS analysis of Ma et al. (2016) to give a spectral algorithm for the robust *and* overcomplete regime. Our primary technical contribution is the efficient implementation of the *lifting* step in the SoS analysis of Ma et al. (2016) as an efficient spectral algorithm to generate surrogate 6th order moments ; this is the subject of Appendix C, and we give an informal description in Section 2.

FOOBI The innovative FOOBI (Fourth-Order Cumulant-Based Blind Identification) algorithm of Lathauwer et al. (2007) was the first method with provable guarantees for overcomplete 4-th order tensor decomposition under algebraic nondegeneracy assumptions. Like our algorithm, FOOBI can be seen as a *lifting* procedure (to an 8-th order tensor) followed by a *rounding* procedure. The FOOBI lifting procedure inspires ours – although ours runs faster because we lift to a 6-tensor rather than an 8-tensor – but the FOOBI rounding step is quite different, and proceeds via a clever simultaneous diagonalization approach. The advantage our algorithm offers over FOOBI is twofold: first, it provides formal, strong robustness guarantees, and second, it has a faster asymptotic runtime.

To the first point: for a litmus test, consider the case that $n = d$ and $a_1, \ldots, a_n \in \mathbb{R}^d$ are orthonormal. On input $T = \sum_{i=1}^n a_i^{\otimes 4} + E$, our algorithm recovers the a_i for arbitrary perturbations E so long as they are bounded in spectral norm by $\|E\| \leqslant 1/\operatorname{poly}\log d$.[7] We are not aware of any formal analyses of FOOBI when run on tensors with arbitrary perturbations of this form. Precisely what degree of robustness should be expected from this modified FOOBI algorithm is unclear. The authors of Lathauwer et al. (2007) do suggest (without analysis) a modification of their algorithm for the setting of nonzero error tensors E, involving an alternating-minimization method for computing an *approximate* simultaneous diagonalization. Because the problem of approximate simultaneous diagonalization is non-convex, establishing robustness guarantees for the FOOBI algorithm when augmented with

7. In contrast, most iterative methods, such as power iteration, can only handle perturbations of spectral norm at most $\|E\| \leqslant 1/\operatorname{poly}(d)$.

the approximate simultaneous diagonalization step appears to be a nontrivial technical endeavor. We think this is an interesting and potentially challenging open question.

Further, while the running time of FOOBI depends on the specific implementation of its linear-algebraic operations, we are unaware of any technique to implement it in time faster than $\tilde{O}(n^3 d^4)$. In particular, the factor of d^4 appears essential to any implementation of FOOBI; it represents the side-length of a $d^4 \times d^4$ square unfolding of a d-dimensional 8-tensor, which FOOBI employs extensively. By contrast, our algorithm runs in time $\tilde{O}(n^2 d^3)$, which is (up to logarithmic factors) faster by a factor of nd.

2. Overview of algorithm

We now describe a simple decomposition algorithm for orthogonal 3-tensors: Gaussian rounding (Harshman (1970)). We then build on that intuition to describe our algorithm.

Orthogonal, undercomplete tensors. Suppose that $u_1, \dots, u_d \in \mathbb{R}^d$ are orthonormal vectors, and that we are given $T = \sum_{i \in [d]} u_i^{\otimes 3}$. As a first attempt at recovering the u_i, one might be tempted to choose the first "slice" of T, the $d \times d$ matrix $T_1 = \sum_i u_i(1) \cdot u_i u_i^\top$, and compute its singular value decomposition (SVD). However, if $|u_i(1)| = |u_j(1)|$ for some $i \neq j \in [d]$, the SVD will not allow us to recover these components. In this setting, Gaussian rounding allows us to exploit the additional mode of T: If we sample $g \sim \mathcal{N}(0, \mathrm{Id}_d)$, then we can take the random flattening $T(g) = \sum_i \langle g, u_i \rangle \cdot u_i u_i^\top$; because the $\langle g, u_i \rangle$ are independent standard Gaussians, they are distinct with probability 1, and an SVD will recover the u_i exactly. Moreover, this algorithm also solves k-tensor decomposition for orthogonal tensors with $k \geqslant 4$, by treating $\sum_{i \in [d]} u_i^{\otimes k}$ as the 3-tensor $\sum_{i \in [d]} u_i^{\otimes k-1} \otimes u_i \otimes u_i$.

Challenges of overcomplete tensors. In our setting, we have unit vectors $\{a_i\}_{i \in [n]} \subset \mathbb{R}^d$ with $n > d$, and $T = \sum_i a_i^{\otimes 4}$ (focusing for now on the unperturbed case). Since $n > d$, the components $a_1, \dots, a_n$ are *not* orthogonal: they are not even linearly independent. So, we cannot hope to use Gaussian rounding as a black box. While the vectors $a_1 \otimes a_1, \dots, a_n \otimes a_n$ may be linearly independent, the spectral decompositions of the matrix $\sum_{i \in [n]} (a_i^{\otimes 2})(a_i^{\otimes 2})^\top$ are not necessarily useful, since its eigenvectors may not be close to any of the vectors a_i, and may be unique only up to rotation.

Challenges of perturbations. Returning momentarily to the orthogonal setting with $n \leqslant d$, new challenges arise when the perturbation tensor E is nonzero. For an orthogonal 4-tensor $T = \sum_{i \in [d]} u_i^{\otimes 4} + E$, the Gaussian rounding algorithm produces the matrix $\sum_{i \in [d]} \langle g, u_i^{\otimes 2} \rangle u_i u_i^\mathsf{T} + E_g$ for some $d \times d$ matrix E_g. The difficulty is that even if the spectral norm $\|E\| \ll \sigma_{\min}(\sum_{i \in [d]} (u_i^{\otimes 2})(u_i^{\otimes 2})^\mathsf{T}) = 1$, the matrix E_g sums many slices of the tensor E, and so the spectrum of E_g can overwhelm that of $\sum_{i \in [d]} \langle g, u_i^{\otimes 2} \rangle u_i u_i^\mathsf{T}$.

This difficulty is studied in Schramm and Steurer (2017), where it is resolved by SoS-inspired preprocessing of the tensor T. We borrow many of those ideas in this work.

Algorithmic strategy. We now give an overview of our algorithm. Algorithm 1 summarizes the algorithm, omitting details concerning robustness and fast implementation.

There are two main stages to the algorithm: the first stage is *lifting*, where the input rank-n 4-tensor over $\mathbb{R}^d$ is lifted to a corresponding rank-n 3-tensor over a higher dimensional

space $\mathbb{R}^{d^2}$; this creates an opportunity to use Gaussian rounding on the newly-created tensor modes. In the second *rounding* stage, the components of the lifted tensor are recovered using a strategy similar to Gaussian rounding and then used to find the components of the input.

This parallels the form of the SoS-based overcomplete tensor decomposition algorithm of Ma et al. (2016), where both stages rely on SoS semidefinite programming. Our main technical contribution is a spectral implementation of the lifting stage; our spectral implementation of the rounding stage reuses many ideas of Schramm and Steurer (2017), adapted round the output of our new lifting stage.

Lifting. The goal of the lifting stage is to transform the input $T = \sum_{i\in[n]}(a_i^{\otimes 2})(a_i^{\otimes 2})^\mathsf{T}$ to an orthogonal 3-tensor. Let $W = T^{-1/2}$ and observe that the *whitened* vectors $W(a_i^{\otimes 2})$ are orthonormal; therefore we will want to use T to find the orthogonal 3-tensor $\sum_{i\in[n]}(Wa_i^{\otimes 2})^{\otimes 3}$.

The lifting works by deriving $\mathrm{Span}(a_i^{\otimes 3})_{i\in[n]}$ from $\mathrm{Span}(a_i^{\otimes 2})_{i\in[n]}$, where the latter is simply the column space of the input T. By transforming $\mathrm{Span}(a_i^{\otimes 3})$ using $W = T^{-1/2}$, we obtain $\mathrm{Span}(W(a_i^{\otimes 2}) \otimes a_i)$. Since $\{W(a_i^{\otimes 2}) \otimes a_i\}_{i\in[n]}$ are orthonormal, the orthogonal projector to their span is in fact equal to $\sum_i (W(a_i^{\otimes 2}) \otimes a_i)(W(a_i^{\otimes 2}) \otimes a_i)^\mathsf{T}$, which is only a reshaping and a final multiplication by W away from the orthogonal tensor $\sum_i (W(a_i^{\otimes 2}))^{\otimes 3}$.

The key step is the operation which obtains $\mathrm{Span}(a_i^{\otimes 3})$ from $\mathrm{Span}(a_i^{\otimes 2})$. It rests on an algebraic "identifiability" argument, which establishes that for almost all problem instances (all but an algebraic set of measure 0), the subspace $\mathrm{Span}(a_i^{\otimes 3})$ is equal to $\mathrm{Span}(a_i^{\otimes 2}) \otimes \mathbb{R}^d$ intersected with the symmetric subspace $\mathrm{Span}(\{x \otimes x \otimes x\}_{x\in\mathbb{R}^d})$. Since we can compute $\mathrm{Span}(a_i \otimes a_i)$ from the input and since the symmetric subspace is easy to describe, we are able to perform this lifting step efficiently. The simplest version of the identifiability argument is given in Lemma 4, and a more robust version that includes a condition number analysis is given in Section C.1.

Lemma 4 (Simple Identifiability) *Let $a_1, \ldots, a_n \in \mathbb{R}^d$ with $n \leqslant d^2$. Let S denote $\mathrm{Span}(\{a_i^{\otimes 2}\})$ and let T denote $\mathrm{Span}(\{a_i^{\otimes 3}\})$ and assume both have dimension n. Let $\mathrm{sym} \subseteq (\mathbb{R}^d)^{\otimes 3}$ be the linear subspace $\mathrm{sym} = \mathrm{Span}(\{x \otimes x \otimes x\}_{x\in\mathbb{R}^d})$. For each i, let $\{b_{i,j}\}_{j\in[d-1]}$ be an arbitrary orthonormal basis the orthogonal complement of a_i in $\mathbb{R}^d$. Let also*

$$K'^\mathsf{T} := \begin{bmatrix} a_1 \otimes a_1 \otimes b_{1,1} & - & a_1 \otimes b_{1,1} \otimes a_1 \\ & \vdots & \\ a_i \otimes a_i \otimes b_{i,j} & - & a_i \otimes b_{i,j} \otimes a_i \\ & \vdots & \\ a_n \otimes a_n \otimes b_{n,d-1} & - & a_n \otimes b_{n,d-1} \otimes a_n \end{bmatrix},$$

Then if K' has full rank $n(d-1)$, it follows that $(S \otimes \mathbb{R}^d) \cap \mathrm{sym} = T$.

Proof To show that $T \subseteq (S \otimes \mathbb{R}^d) \cap \mathrm{sym}$, we simply note that $\{a_i \otimes a_i \otimes a_i\}_{i\in[n]}$ form a basis for T and are also each in both $S \otimes \mathbb{R}^d$ and sym.

To show that $(S \otimes \mathbb{R}^d) \cap \mathrm{sym} \subseteq T$, we take some $y \in (S \otimes \mathbb{R}^d) \cap \mathrm{sym}$. Since y is symmetric under mode interchange, we express y in two ways as

$$y = \sum_i a_i \otimes a_i \otimes c_i = \sum_i a_i \otimes c_i \otimes a_i \,.$$

Then by subtracting these two expressions for y from each other, we find

$$0 = \sum_i a_i \otimes (a_i \otimes c_i - c_i \otimes a_i).$$

We express $c_i = \langle a_i, c_i \rangle a_i + \sum_j \gamma_{ij} b_{ij}$ for some vector γ. Then the symmetric parts cancel out, leaving

$$0 = \sum_{ij} \gamma_{ij}\, a_i \otimes (a_i \otimes b_{ij} - b_{ij} \otimes a_i) = K'\gamma\,.$$

Since K' is full rank by assumption, this is only possible when $\gamma = 0$. Therefore, $c_i \propto a_i$ for all i, so that $y \in T$. ■

Remark 5 *Although the condition number from the matrix K' here is not the same as the one derived from K from Definition 3, it is off by at most a multiplicative factor of $2\|T^{-1}\|^{-1/2}$. To see this, K in Definition 3 is given as $K = \Pi^{\perp}_{2,3}\Pi_{\mathrm{img}(H^{\mathsf{T}})}$, whereas we may write $K' = 2\Pi^{\perp}_{2,3}H^{\mathsf{T}} = 2\Pi^{\perp}_{2,3}\Pi_{\mathrm{img}(H^{\mathsf{T}})}(HH^{\mathsf{T}})^{1/2} = 2K(HH^{\mathsf{T}})^{1/2}$. Therefore, $\|K'^{-1}\| \geqslant \frac{1}{2}\|K^{-1}\|\,\|H^{-1}\|$. By (Ma et al., 2016, Lemma 6.3), $\|H^{-1}\|^2 \geqslant \|T^{-1}\|$.*

Robustness. To ensure that our algorithm is robust to perturbations E, we must argue that the column span of T and $T + E$ are close to each other so long as E is bounded in spectral norm, and furthermore than the lifting operation still produces a subspace V which is close to $\mathrm{Span}(\{W(a_i \otimes a_i) \otimes a_i\})$. This is done via careful application of matrix perturbation analysis to the identifiability argument. By operating with W *only* on third-order vectors and matrices over $(\mathbb{R}^d)^{\otimes 3}$, we also avoid incurring factors of the fourth-order operator norm $\|T\|$ in the condition numbers, instead only incurring a much milder *sixth*-order penalty $\|\sum a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}\|$. For details, see Section C.2.

Rounding. If we are given direct access to T in the absence of noise, the rounding stage can be accomplished with Gaussian rounding. However when we allow T to be adversarially perturbed the situation becomes more delicate. Our rounding stage is an adaptation of Schramm and Steurer (2017), though some modifications are required for the additional challenges of the overcomplete setting. It recovers the components of an approximation of a 3-tensor with n orthonormal components, provided that said approximation is within $\varepsilon\sqrt{n}$ in Frobenius norm distance. The technique is built around Gaussian rounding, but in order to have this succeed in the presence of $\varepsilon\sqrt{n}$ Frobenius norm noise, the large singular values are truncated from the rectangular matrix reshapings of the 3-tensor: this ensures that the rounding procedure is not entirely dominated by any spectrally large terms in the noise.

After we recover approximations of the orthonormal components $b_i \approx W a_i^{\otimes 2}$, we wish to extract the a_i. Naively one could simply apply W^{-1}, but this can cause errors in the recovered vectors to blow up by a factor of $\|W^{-1}\|$. Even when the $\{a_i\}$ are random vectors, $\|W^{-1}\| = \Omega(\mathrm{poly}(d))$.[8] Instead, we utilize the projector to $\mathrm{Span}\{W(a_i \otimes a_i) \otimes a_i\}$ computed in the lifting step: we *lift* b_i, project it into the span to obtain a vector close to $W(a_i \otimes a_i) \otimes a_i$, and reshape it to a $d^2 \times d$ matrix whose top right-singular vector is correlated with a_i. This extraction-via-lifting step allows us to circumvent a loss of $\|W^{-1}\|$ in the error.

8. This is in contrast to $\|W\|$, which is $O(1)$ in the random case.

Algorithm 1 Sketch of full algorithm, in the absence of noise

Input: A 4-tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 4}$, so that $\mathbf{T} = \sum_{i=1}^n a_i^{\otimes 4}$ for unit vectors $a_i \in \mathbb{R}^d$.

1. Take the square reshaping $T \in \mathbb{R}^{d^2 \times d^2}$ of $\mathbf{T}$ and compute its whitening $W = T^{-1/2}$ (where $T^{-1/2}$ refers to the Moore-Penrose pseudo-inverse of the positive-semidefinite square root of T) and the projector $\Pi_2 = WTW$ to the image of T.
2. *Lifting*: Compute the lifted tensor $\mathbf{T}' \in (\mathbb{R}^{d^2})^{\otimes 3}$ so that $\mathbf{T}' = \sum_i (W a_i^{\otimes 2})^{\otimes 3}$. (See Algorithm 2 for full details).
   (a) Find a basis for the subspace $S_3 = (\operatorname{img} T) \otimes \mathbb{R}^d \cap \operatorname{sym}$: take S_3 to be the top-n eigenspace of $(\Pi_2 \otimes \mathrm{Id})\Pi_{\mathrm{sym}}(\Pi_2 \otimes \mathrm{Id})$. Then by Lemma 4, $S_3 = \operatorname{Span}(a_i^{\otimes 3})$.
   (b) Find the projector Π_3 to the space $(W \otimes \mathrm{Id})\, S_3 = \operatorname{Span}(W a_i^{\otimes 2} \otimes a_i)$.
   (c) Compute the orthogonal 3-tensor: since $\{W a_i^{\otimes 2} \otimes a_i\}$ is an orthonormal basis,
   $$\Pi_3 = \sum\nolimits_i (W a_i^{\otimes 2} \otimes a_i)(W a_i^{\otimes 2} \otimes a_i)^{\mathsf{T}} .$$
   Therefore, reshape Π_3 as $\sum_i (W a_i^{\otimes 2}) \otimes (W a_i^{\otimes 2}) \otimes (a_i^{\otimes 2})$ and multiply W into the third mode to obtain $\mathbf{T}'$.
3. *Rounding*: Use Gaussian rounding to find the components a_i. (In the presence of noise, this step becomes substantially more delicate; see Algorithms 3 to 5).
   (a) Compute a random flattening of $\mathbf{T}'$ by contracting with $g \sim \mathcal{N}(0, \mathrm{Id}_{d^2})$ along the first mode, $T'(g) = \sum_i \langle g, (W a_i^{\otimes 2})\rangle \cdot (W a_i^{\otimes 2})(W a_i^{\otimes 2})^{\top}$
   (b) Perform an SVD on $T'(g)$ to recover the eigenvectors $(W a_1^{\otimes 2}), \ldots, (W a_n^{\otimes 2})$.
   (c) Apply W^{-1} to each eigenvector to obtain the $a_i^{\otimes 2}$, and re-shape $a_i^{\otimes 2}$ to a matrix and compute its eigenvector to obtain a_i.

Organization of technical details.

The full implementation details and the analysis of our algorithm are given in the following sections of the appendix. First, Appendix B sets up some primitives for spectral subspace perturbation analysis and linear-algebraic procedures on which we build the full algorithm and its analysis. Then Appendix C covers the lifting stage of the algorithm in detail, while Appendix D elaborates on the rounding stage. Finally, in Appendix E we combine these tools to prove Theorem 25.

In Appendix H, we detail simulations strongly suggesting that various families of random tensors (uniform, discrete, sparse, and spiked) with $n \ll d^2$ components have constant condition number κ.

3. Acknowledgements

We thank David Steurer for many helpful conversations regarding the technical content and presentation of this work. Sam Hopkins was supported by NSF awards CFF-1350196 and

CFF-1408673, and a Microsoft PhD Fellowship. Tselil Schramm was supported by NSF awards CCF-1565264 and CNS-1618026. Jonathan Shi was supported by David Steurer's NSF CAREER grant CCF-1350196 and CFF-1408673.

References

Evrim Acar, Canan Aykut-Bingol, Haluk Bingol, Rasmus Bro, and Bülent Yener. Multiway analysis of epilepsy tensors. *Bioinformatics*, 23(13):i10–i18, 2007. doi: 10.1093/bioinformatics/btm210. URL http://dx.doi.org/10.1093/bioinformatics/btm210.

Zeyuan Allen-Zhu and Yuanzhi Li. Lazysvd: Even faster svd decomposition yet without agonizing pain. In *Advances in Neural Information Processing Systems*, pages 974–982, 2016.

Anima Anandkumar, Rong Ge, and Majid Janzamin. Analyzing tensor power method dynamics: Applications to learning overcomplete latent variable models. *CoRR*, abs/1411.1488, 2014a.

Animashree Anandkumar, Rong Ge, Daniel J. Hsu, and Sham Kakade. A tensor spectral approach to learning mixed membership community models. In *COLT*, volume 30 of *JMLR Workshop and Conference Proceedings*, pages 867–881. JMLR.org, 2013.

Animashree Anandkumar, Rong Ge, Daniel J. Hsu, Sham M. Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, 15(1):2773–2832, 2014b.

Animashree Anandkumar, Rong Ge, and Majid Janzamin. Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates. *CoRR*, abs/1402.5180, 2014c.

Animashree Anandkumar, Rong Ge, and Majid Janzamin. Analyzing tensor power method dynamics in overcomplete regime. *Journal of Machine Learning Research*, 18:22:1–22:40, 2017.

Boaz Barak, Jonathan A. Kelner, and David Steurer. Dictionary learning and tensor decomposition via the sum-of-squares method. In *STOC*, pages 143–151. ACM, 2015.

C.F. Beckmann and S.M. Smith. Tensorial extensions of independent component analysis for multisubject fmri analysis. *NeuroImage*, 25(1):294 – 311, 2005. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2004.10.043. URL http://www.sciencedirect.com/science/article/pii/S1053811904006378.

Aditya Bhaskara, Moses Charikar, Ankur Moitra, and Aravindan Vijayaraghavan. Smoothed analysis of tensor decompositions. In *STOC*, pages 594–603. ACM, 2014.

Chandler Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.*, 7:1–46, 1970. ISSN 0036-1429. doi: 10.1137/0707001. URL http://dx.doi.org/10.1137/0707001.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. Blind source separation by simultaneous third-order tensor diagonalization. In *European Signal Processing Conference, 1996. EUSIPCO 1996. 8th*, pages 1–4. IEEE, 1996.

Lieven De Lathauwer, Josphine Castaing, and Jean-Franois Cardoso. Fourth-order cumulant-based blind identification of underdetermined mixtures. *IEEE Transactions on Signal Processing*, 55(6):2965–2973, 2007.

Michael Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN 144197010X, 9781441970107.

Rong Ge and Tengyu Ma. Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms. In *APPROX-RANDOM*, volume 40 of *LIPIcs*, pages 829–849. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

Rong Ge and Tengyu Ma. On the optimization landscape of tensor decompositions. In *Advances in Neural Information Processing Systems*, pages 3653–3663, 2017.

Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points - online stochastic gradient for tensor decomposition. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, volume 40 of *JMLR Workshop and Conference Proceedings*, pages 797–842. JMLR.org, 2015a. URL http://jmlr.org/proceedings/papers/v40/Ge15.html.

Rong Ge, Qingqing Huang, and Sham M. Kakade. Learning mixtures of Gaussians in high dimensions [extended abstract]. In *STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing*, pages 761–770. ACM, New York, 2015b.

Wu Hai-Long, Shibukawa Masami, and Oguma Koichi. An alternating trilinear decomposition algorithm with application to calibration of HPLC-DAD for simultaneous determination of overlapped chlorinated aromatic hydrocarbons. *Journal of Chemometrics*, 12(1):1–26. doi: 10.1002/(SICI)1099-128X(199801/02)12:1<1::AID-CEM492>3.0.CO;2-4.

Richard A Harshman. Foundations of the parafac procedure: Models and conditions for an" explanatory" multi-modal factor analysis. 1970.

Christopher J. Hillar and Lek-Heng Lim. Most tensor problems are np-hard. *J. ACM*, 60(6): 45:1–45:39, 2013.

Samuel B Hopkins and David Steurer. Efficient bayesian estimation from few samples: community detection and related problems. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 379–390. IEEE, 2017.

Samuel B. Hopkins, Jonathan Shi, and David Steurer. Tensor principal component analysis via sum-of-square proofs. In *COLT*, volume 40 of *JMLR Workshop and Conference Proceedings*, pages 956–1006. JMLR.org, 2015.

Samuel B. Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer. Fast spectral algorithms from sum-of-squares proofs: tensor decomposition and planted sparse vectors. In *STOC*, pages 178–191. ACM, 2016.

Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

Lieven De Lathauwer, Joséphine Castaing, and Jean-François Cardoso. Fourth-order cumulant-based blind identification of underdetermined mixtures. *IEEE Trans. Signal Processing*, 55(6-2):2965–2973, 2007.

Michael S. Lewicki and Terrence J. Sejnowski. Learning overcomplete representations. *Neural Comput.*, 12(2):337–365, February 2000. ISSN 0899-7667. doi: 10.1162/089976600300015826. URL http://dx.doi.org/10.1162/089976600300015826.

Tengyu Ma, Jonathan Shi, and David Steurer. Polynomial-time tensor decompositions with sum-of-squares. In *FOCS*, pages 438–446. IEEE Computer Society, 2016.

Marco Mondelli and Andrea Montanari. On the connection between learning two-layers neural networks and tensor decomposition. *CoRR*, abs/1802.07301, 2018. URL http://arxiv.org/abs/1802.07301.

Emile Richard and Andrea Montanari. A statistical model for tensor PCA. In *NIPS*, pages 2897–2905, 2014.

Tselil Schramm and David Steurer. Fast and robust tensor decomposition with applications to dictionary learning. In *COLT*, volume 65 of *Proceedings of Machine Learning Research*, pages 1760–1793. PMLR, 2017.

Vatsal Sharan and Gregory Valiant. Orthogonalized ALS: A theoretically principled tensor decomposition algorithm for practical use. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 3095–3104. PMLR, 2017.

Hermann Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). *Mathematische Annalen*, 71(4):441–479, Dec 1912. ISSN 1432-1807. doi: 10.1007/BF01456804. URL https://doi.org/10.1007/BF01456804.

Appendix A. Preliminaries

Linear algebra We use Id_d to denote the $d \times d$ identity matrix, or just Id if the dimension is clear from context. For any subspace S, we use Π_S to denote the projector to that subspace. For M a matrix, $\mathrm{img}(M)$ refers to the image, or columnspace, of M.

We will, in a slight abuse of notation, use M^{-1} to denote the Moore-Penrose pseudo-inverse of M. Except where explicitly specified, this will never be assumed to be equal to the proper inverse, so that, e.g., in general $MM^{-1} = \Pi_{\mathrm{img}(M)} \neq \mathrm{Id}$ and $(AB)^{-1} \neq B^{-1}A^{-1}$.

For a matrix $B \in \mathbb{R}^{m \times n}$, we will use the whitening matrix $W = (BB)^{-1/2}$, which maps the columns of B to an orthonormal basis for $\mathrm{img}(B)$, so that $(WB)(WB)^\top = \Pi_{\mathrm{img}(B)}$.

We denote by $\mathrm{sym} \subseteq (\mathbb{R}^d)^{\otimes 3}$ the linear subspace

$$\mathrm{sym} = \mathrm{Span}(\{x \otimes x \otimes x\}_{x \in \mathbb{R}^d}) .$$

Note that $(\Pi_{\mathrm{sym}})_{(i,j,k);(i',j',k')}$ is $(|\{i,j,k\}|!)^{-1}$ when $\{i,j,k\} = \{i',j',k'\}$ and zero otherwise.

Tensor manipulations When working with tensors $T \in (\mathbb{R}^d)^{\otimes k}$, we will sometimes reshape the tensors to lower-order tensors or matrices; in this case, if $S_1, \ldots, S_m$ are a partition of k, then $T_{(S_1,\ldots,S_m)}$ is the tensor given by identifying the modes in each S_i into a single mode. For $S \subset [d]^k$, we will also sometimes use the notation $T(S)$ to refer to the entry of T indexed by S.

A useful property of matrix reshapings is that $u \otimes v$ reshapes into the outer product uv^T. Linearity allows us to generalize this so, e.g., the reshaping of $(U \otimes V)M$ for $U \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{m \times m}$ and $M \in \mathbb{R}^{(n \otimes m) \times q}$ is equal to $UM'(V \otimes \mathrm{Id}_q)$, where $M' \in \mathbb{R}^{n \times (m \otimes q)}$ is the reshaping of M. Since reshapings can be easily done and undone by exchanging indices, these identities will sometimes allow more efficient computation of matrix products over tensor spaces.

We will on occasion use a $\cdot$ as a placeholder in a partially applied multiple-argument function: for instance $\frac{\partial}{\partial y} f(\cdot, y) = \lim_{h \to 0} \frac{1}{h}(f(\cdot, y+h) - f(\cdot, y))$.

Appendix B. Tools for analysis and implementation

In this section, we briefly introduce some tools which we will use often in our analysis.

B.1. Robustness and spectral perturbation

A key tool in our analysis of the robustness of Algorithm 1 comes from the theory of the perturbation of eigenvalues and eigenvectors.

The lemma below combines the Davis-Kahan sin-Θ theorem with Weyl's inequality to characterize how top eigenspaces are affected by spectral perturbation.

Theorem 6 (Perturbation of top eigenspace) *Suppose $Q \in \mathbb{R}^{D \times D}$ is a symmetric matrix with eigenvalues $\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_D$. Suppose also $\widetilde{Q} \in \mathbb{R}^{D \times D}$ is a symmetric matrix with $\|Q - \widetilde{Q}\| \leqslant \varepsilon$. Let S and $\widetilde{S}$ be the spaces generated by the top n eigenvectors of Q and $\widetilde{Q}$ respectively. Then,*

$$\sin(S, \widetilde{S}) \stackrel{\mathrm{def}}{=} \|\Pi_S - \Pi_{\widetilde{S}} \Pi_S\| = \|\Pi_{\widetilde{S}} - \Pi_S \Pi_{\widetilde{S}}\| \leqslant \frac{\varepsilon}{\lambda_n - \lambda_{n+1} - 2\varepsilon} . \tag{B.1}$$

Consequently,

$$\|\Pi_S - \Pi_{\widetilde{S}}\| \leqslant \frac{2\varepsilon}{\lambda_n - \lambda_{n+1} - 2\varepsilon} . \tag{B.2}$$

Proof We first prove the theorem assuming that Q and $\widetilde{Q}$ are symmetric. By Weyl's inequality for matrices Weyl (1912), the nth eigenvalue of $\widetilde{Q}$ is at least $\lambda_n - 2\varepsilon$. By Davis and Kahan's sin-Θ theorem Davis and Kahan (1970), since the top-n eigenvalues of $\widetilde{Q}$ are all at least $\lambda_n - 2\varepsilon$ and the lower-than-n eigenvalues of Q are all at most λ_{n+1}, the sine of the angle between S and $\widetilde{S}$ is at most $\|Q - \widetilde{Q}\|/(\lambda_n - \lambda_{n+1} - 2\varepsilon)$. The final bound on $\|\Pi_S - \Pi_{\widetilde{S}}\|$ follows by triangle inequality. ∎

B.2. Efficient implementation and runtime analysis

It is not immediately obvious how to implement Algorithm 1 in time $\tilde{O}(n^2 d^3)$, since there are steps that require we multiply or eigendecompose $d^3 \times d^3$ matrices, which if done naively might take up to $\Omega(d^9)$ time.

To accelerate our runtime, we must take advantage of the fact that our matrices have additional structure. We exploit the fact that in certain reshapings our tensors have low-rank representations. This allows us to perform matrix multiplication and eigendecomposition (via power iteration) efficiently, and obtain a runtime that is depends on the rank rather than on the dimension.

For example, the following lemma, based upon a result of Allen-Zhu and Li (2016), captures our eigendecomposition strategy in a general sense.

Lemma 7 (Implicit gapped eigendecomposition) *Suppose a symmetric matrix $M \in \mathbb{R}^{d \times d}$ has an eigendecomposition $M = \sum_j \lambda_j v_j v_j^\mathsf{T}$, and that Mx may be computed within t time steps for $x \in \mathbb{R}^d$. Then $v_1, \ldots, v_n$ and $\lambda_1, \ldots, \lambda_n$ may be computed in time $\tilde{O}(\min(n(t + nd)\delta^{-1/2}, d^3))$, where $\delta = (\lambda_n - \lambda_{n+1})/\lambda_n$. The dependence on the desired precision is polylogarithmic.*

Proof The $n(t + nd)\delta^{-1/2}$ runtime is attained by LazySVD in (Allen-Zhu and Li, 2016, Corollary 4.3). While LazySVD's runtime depends on $\mathrm{nnz}(M)$ where nnz denotes the number of non-zero elements in the matrix, in the non-stochastic setting $\mathrm{nnz}(M)$ is used only as a bound on the time cost of multiplying a vector by M, so in our case we may substitute $O(t)$ instead.

The d^3 time is attained by iterated squaring of M: in this case, all runtime dependence on condition numbers is polylogarithmic. ■

The following lemma lists some primitives for operations with the tensor $\mathbf{T}' \in (\mathbb{R}^{d^2})^{\otimes 3}$ in Algorithm 1, by interpreting it as a 6-tensor in $(\mathbb{R}^d)^{\otimes 6}$ and using a low-rank factorization of the square reshaping of that 6-tensor.

Lemma 8 (Implicit tensors) *For a tensor $\mathbf{T} \in (\mathbb{R}^{[d]^2})^{\otimes 3}$, suppose that the matrix $T \in \mathbb{R}^{[d]^3 \times [d]^3}$ given by $T_{(i,i',j),(k,k',j')} = \mathbf{T}_{(i,i'),(j,j'),(k,k')}$ has a rank-n decomposition $T = UV^\mathsf{T}$ with $U, V \in \mathbb{R}^{d^3 \times n}$ and $n \leqslant d^2$. Such a rank decomposition provides an implicit representation of the tensor $\mathbf{T}$. This implicit representation supports:*

Tensor contraction: *For vectors $x, y \in \mathbb{R}^{[d]^2}$, the computation of $(x^\mathsf{T} \otimes y^\mathsf{T} \otimes \mathrm{Id})\mathbf{T}$ or $(x^\mathsf{T} \otimes \mathrm{Id} \otimes y^\mathsf{T})\mathbf{T}$ or $(\mathrm{Id} \otimes x^\mathsf{T} \otimes y^\mathsf{T})\mathbf{T}$ in time $O(nd^3)$ to obtain an output vector in $\mathbb{R}^{d^2}$.*

Spectral truncation: *For $R \in \mathbb{R}^{d^2 \times d^4}$ equal to one of the two matrix reshapings $T_{\{1,2\}\{3\}}$ or $T_{\{2,3\}\{1\}}$ of $\mathbf{T}$, an approximation to the tensor $\mathbf{T}^{\leqslant 1}$, defined as $\mathbf{T}$ after all larger-than-1 singular values in its reshaping R are truncated down to 1. Specifically, letting ρ_k be the kth largest singular value of R for $k \leqslant O(n)$, this returns an implicit representation of a tensor $\mathbf{T}'$ such that $\|\mathbf{T}' - \mathbf{T}^{\leqslant 1}\|_F \leqslant (1+\delta)\rho_k \|\mathbf{T}\|_F$ and the reshaping of $\mathbf{T}'$ corresponding to R has largest singular value no more than $1 + (1+\delta)\rho_k$. The representation of $\mathbf{T}'$ also supports the tensor contraction, spectral truncation, and implicit matrix multiplication operations, with no more than a constant factor increase in runtime. This takes time $\tilde{O}(n^2 d^3 + k(nd^3 + kd^2)\delta^{-1/2})$.*

Implicit matrix multiplication: *For a matrix $R \in \mathbb{R}^{[d]^2 \times [d]^2}$ with rank at most $O(n)$, an implicit representation of the tensor $(R^\mathsf{T} \otimes \mathrm{Id} \otimes \mathrm{Id})\mathbf{T}$ or $(\mathrm{Id} \otimes \mathrm{Id} \otimes R^\mathsf{T})\mathbf{T}$, in time $O(nd^4)$. This output also supports the tensor contraction, spectral truncation, and implicit matrix multiplication operations, with no more than a constant factor increase in runtime. Multiplication into the second mode $(\mathrm{Id} \otimes R^\mathsf{T} \otimes \mathrm{Id})\mathbf{T}$ may also be implicitly represented, but without support for the spectral truncation operation.*

The implementation of these implicit tensor operations consists solely of tensor reshapings, singular value decompositions, and matrix multiplication. However, the details get involved and lengthy, and so we defer their exposition to Section F.

Appendix C. Lifting

This section presents Algorithm 2, which lifts a well-conditioned 4-tensor $\mathbf{T}$ of rank at most d^2 in $(\mathbb{R}^d)^{\otimes 3}$ to $\mathbf{T}'$, an orthogonalized version of the 6-tensor in the same components in $(\mathbb{R}^{d^2})^{\otimes 3}$; that is, we obtain an orthogonal 3-tensor $\mathbf{T}'$ whose components correspond to the orthogonalized Kronecker squares of the components of $\mathbf{T}$. Section C.1 presents the identifiability argument giving robust algebraic non-degeneracy conditions under which the algorithm succeeds.

Although we assume that the tensor components a_i are unit vectors, throughout this section we will keep track of factors of $\|a_i\|$ so as to better elucidate the scaling and dimensional analysis.

Algorithm 2 Function $\textsc{lift}(\mathbf{T}, n)$

Input: $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 4}, n \in \mathbb{N}$ with $n \leqslant d^2$.

1. Use Lemma 7 to find the top-n eigenvalues and corresponding eigenvectors of the square matrix reshaping of $\mathbf{T}$, and call the eigendecomposition $T = Q\Lambda Q^\mathsf{T}$. This also yields $W = Q\Lambda^{-1/2}Q^\mathsf{T}$ and $\Pi_S = QQ^\mathsf{T}$.

2. Use Lemma 7 again to find the top-n eigendecomposition of $\Pi_{S\otimes\mathbb{R}^d}\Pi_{\mathrm{sym}}\Pi_{S\otimes\mathbb{R}^d}$, implementing multiplication by $\Pi_{S\otimes\mathbb{R}^d}$ as $(\Pi_{S\otimes\mathbb{R}^d}v)_{(\cdot,\cdot,i)} = QQ^\mathsf{T}v_{(\cdot,\cdot,i)}$ and implementing Π_{sym} as a sparse matrix. Call the result $R\Sigma R^\mathsf{T}$ and take $\Pi_{S_3} = RR^\mathsf{T}$.

3. Find a basis B' for the columnspace of $M_3 = (W \otimes \mathrm{Id})\Pi_{S_3}(W \otimes \mathrm{Id})$. Implement this as
$$(B')_{(\,\cdot\,,\,\cdot\,,i);\,\cdot} = Q\Lambda^{-1/2}Q^\mathsf{T}R_{(\,\cdot\,,\,\cdot\,,i);\,\cdot}\,.$$

4. Use Gram-Schmidt orthogonalization to find an orthonormalization B of B'. Call the projection operator to this basis $\Pi_3 = BB^\mathsf{T}$.

5. Instantiate an implicit tensor in $(\mathbb{R}^{d^2})^{\otimes 3}$ with Lemma 8, using BB^T as the SVD of its underlying $d^3 \times d^3$ reshaping. Output this as $(\mathrm{Id} \otimes W^{-1} \otimes \mathrm{Id})\mathbf{T}'$, meaning a tensor which, when W^{-1} is multiplied into its second mode, becomes equal to $\mathbf{T}'$.

Output: $(\mathrm{Id} \otimes W^{-1} \otimes \mathrm{Id})\mathbf{T}' \in (\mathbb{R}^{d^2})^{\otimes 3}$, implicitly as specified by Lemma 8, and $\Pi_3 \in \mathbb{R}^{d^3 \times d^3}$.

The following two lemmas will argue that the algorithm is correct, and that it is fast. First, Lemma 9 states that the output of Algorithm 2 is an orthogonal 3-tensor whose components are $W(a_i \otimes a_i)$, where the a_i are the components of the original 4-tensor and W is the whitening matrix for the $a_i \otimes a_i$. Furthermore, if the error in the input is small in spectral norm compared to some condition numbers, the Frobenius norm error in the output robustly remains within a small constant of $\sqrt{n}$.

The main work of the lemma is deferred to Lemma 13 in Section C.2, which repeatedly applies Davis and Kahan's sin-Θ theorem (Theorem 6) to say that the top eigenspaces of various matrices in the algorithm are relatively unperturbed in spectral norm by small spectral norm error in the matrices. After that, we simply bound the Frobenius norm error of a rank-n matrix by $2\sqrt{n}$ times its spectral norm error, and reason that Frobenius norms are unchanged by tensor reshapings.

Lemma 9 (Correctness of LIFT) *Let $a_1, \dots, a_n \in \mathbb{R}^d$ and suppose that $\mathbf{T} = \sum_{i\in[n]} a_i^{\otimes 4} + \mathbf{E}$ satifies $\|E_{12;34}\| \leqslant \varepsilon \sigma_n^2 \mu^{-1} \kappa^2$ for some $\varepsilon < 1/63$, where σ_n is the nth eigenvalue of $\sum_{i\in[n]} a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}}$ and μ is the operator norm of $\sum \|a_i\|^{-2} a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}$ and κ is the condition number from Lemma 11. Then the outputs $(\mathrm{Id} \otimes W^{-1} \otimes \mathrm{Id})\mathbf{T}'$ and Π_3 of $\textsc{lift}(\mathbf{T}, n)$ in Algorithm 2 satisfy*

$$\left\| \mathbf{T}' - \sum_i \|a_i\|^{-2} (W(a_i \otimes a_i))^{\otimes 3} \right\|_F \leqslant 126\, \varepsilon\, \sigma_n^{-1/2} \sqrt{n}$$

and

$$\left\| \Pi_3 - \Pi_{\mathrm{Span}\left(W a_i^{\otimes 2} \otimes a_i\right)} \right\| \leqslant 63\, \varepsilon\,.$$

Proof By Lemma 13, the Π_{S_3} computed in step 2 as the projector to the top-n eigenspace of $\Pi_{S\otimes\mathbb{R}^d} \Pi_{\mathrm{sym}} \Pi_{S\otimes\mathbb{R}^d}$ satisfies $\|\Pi_{S_3} - \Pi_{\mathrm{Span}(a_i^{\otimes 3})}\| \leqslant 18\, \varepsilon \sigma_n \mu^{-1}$, and subsequently, the Π_3 computed in steps 3 and 4 as the projector to $(W \otimes \mathrm{Id})S_3$ satisfies $\|\Pi_3 - \Pi_{\mathrm{Span}(W a_i^{\otimes 2} \otimes a_i)}\| \leqslant 63\, \varepsilon$.

Since the rank of the error is at most $2n$, the Frobenius norm error is at most $126\, \varepsilon \sqrt{n}$, and since $\{\|a_i\|^{-1} W a_i^{\otimes 2} \otimes a_i\}$ is an orthonormal set of vectors, the projector to $\mathrm{Span}(W a_i^{\otimes 2} \otimes a_i)$ is just the sum of the self-outer-products of vectors in that set, so

$$\left\| \Pi_3 - \sum \|a_i\|^{-2} (W a_i^{\otimes 2} \otimes a_i)(W a_i^{\otimes 2} \otimes a_i)^{\mathsf{T}} \right\|_F \leqslant 126\, \varepsilon \sqrt{n}\,.$$

Reshaping the $d^3 \times d^3$ matrix Π_3 into a tensor in $(\mathbb{R}^{d^2})^{\otimes 3}$ does not change the Frobenius norm error, and finally, multiplying in the last factor of W may contribute a factor of $\|W\| = \sigma_n^{-1/2}$, so that in the end, $\|\mathbf{T}' - \sum_i \|a_i\|^{-2} (W(a_i \otimes a_i))^{\otimes 3}\|_F \leqslant 126\, \varepsilon\, \sigma_n^{-1/2} \sqrt{n}$. ∎

The next lemma states that the running time is $\tilde{O}(n^2 d^3)$ multiplied by some condition numbers. We assume that asympotically faster matrix multiplications and pseudo-inversions are not used, so that, for instance, squaring a $d \times d$ matrix takes time $\Theta(d^3)$.

Lemma 10 (Running time of LIFT) *Let $a_1, \dots, a_n \in \mathbb{R}^d$ and suppose that $\mathbf{T} = \sum_{i\in[n]} a_i^{\otimes 4} + \mathbf{E}$ satisfies the conditions stated in Lemma 9. Let σ_n be the nth eigenvalue of $\sum_{i\in[n]} a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}}$ and κ the condition number from Lemma 11. Then $\textsc{lift}(\mathbf{T}, n)$ in Algorithm 2 runs in time $\tilde{O}(n d^4 \sigma_n^{-1/2} + n^2 d^3 \kappa^{-1})$, and the efficient implementation steps are correct.*

Proof Step 1 of LIFT invokes Lemma 7 on a $d^2 \times d^2$ matrix T, recovering n dimensions with a spectral gap of $\delta = \sigma_n$. This requires time $\tilde{O}((nd^4 + n^2d^2)\sigma_n^{-1/2})$.

Step 2 again invokes Lemma 7, this time on a $d^3 \times d^3$ matrix $\Pi_{S\otimes\mathbb{R}^d}\Pi_{\text{sym}}\Pi_{S\otimes\mathbb{R}^d}$, recovering n dimensions with a spectral gap of at least $\kappa^2 - 2\varepsilon \in \Omega(\kappa^2)$. Multiplying by $\Pi_{S\otimes\mathbb{R}^d}$ may be done in time $O(nd^3)$ due to its expression as $(\Pi_{S\otimes\mathbb{R}^d}v)_{(\cdot,\cdot,i)} = QQ^\mathsf{T}v_{(\cdot,\cdot,i)}$, since the third mode of $(\mathbb{R}^d)^{\otimes 3}$ is unaffected by $\Pi_{S\otimes\mathbb{R}^d} = QQ^\mathsf{T} \otimes \mathrm{Id}$, and this is a concatenation of d different matrix-vector multiplies that take $O(nd^2)$ time each. Multiplying by Π_{sym} takes $O(d^3)$ time, since the (i,j,k)th row of Π_{sym} has at most 6 nonzero entries corresponding to the different permutations of (i,j,k). Thus the overall time to multiply a vector by $\Pi_{S\otimes\mathbb{R}^d}\Pi_{\text{sym}}\Pi_{S\otimes\mathbb{R}^d}$ is $O(nd^3)$, so that Lemma 7 gives a runtime of $\tilde{O}((n^2d^3 + n^2d^3)\kappa^{-1})$ for this step.

Step 3 is a concatenation of d different matrix products, each of which involves multiplying a $d^2 \times n$ matrix $R_{(\cdot,\cdot,i);\cdot}$ by a $n \times d^2$ matrix $\Lambda^{1/2}Q^\mathsf{T}$ and then multiplying the resulting $n \times n$ matrix by a $d^2 \times n$ matrix Q. Each product thus takes $O(n^2d^2)$ time, and since there are d of them the entire step takes $O(n^2d^3)$ time. The result is equal to $(Q\Lambda^{1/2}Q^\mathsf{T} \otimes \mathrm{Id}_d)R = (W \otimes \mathrm{Id})R$, whose columns form a basis for the columnspace of $M_3 = (W \otimes \mathrm{Id})R\Sigma R^\mathsf{T}(W \otimes \mathrm{Id})$.

Step 4 applies Gram-Schmidt orthonormalization on n vectors in $\mathbb{R}^{d^3}$, taking $\tilde{O}(n^2d^3)$ time. And step 5 takes constant time. Therefore, LIFT takes time $\tilde{O}(nd^4\sigma_n^{-1/2} + n^2d^3\kappa^{-1})$. ■

C.1. Algebraic identifiability argument

The main lemma in this section gives a more careful analysis of the algebraic identifiability argument from Lemma 4, in order to obtain a quantitative condition number bound.

Lemma 11 (Main Identifiability Lemma) *Let $a_1, \ldots, a_n \in \mathbb{R}^d$ with $n \leqslant d^2$. Let S denote* $\mathrm{Span}(\{a_i^{\otimes 2}\})$ *and let S_3 denote* $\mathrm{Span}(\{a_i^{\otimes 3}\})$ *and assume both have dimension n. For each i, let $\{b_{i,j}\}_{j\in[d-1]}$ be an arbitrary orthonormal basis for vectors in $\mathbb{R}^d$ orthogonal to a_i, and let*

$$H^\mathsf{T} := \begin{bmatrix} a_1 \otimes a_1 \otimes b_{1,1} \\ \vdots \\ a_i \otimes a_i \otimes b_{i,j} \\ \vdots \\ a_n \otimes a_n \otimes b_{n,d-1} \end{bmatrix}.$$

Let $R = (HH^\mathsf{T})^{-1/2}H$ be a column-wise orthonormalization of H, and let $K = \frac{1}{2}(\mathrm{Id} - P_{2,3})R$, where $P_{2,3}$ is the permutation matrix that exchanges the 2nd and 3rd modes of $(\mathbb{R}^d)^{\otimes 3}$. Then if $\kappa = \sigma_{\min}(K)$ is non-zero (so that K is full rank),

$$(S \otimes \mathbb{R}^d) \cap \mathrm{sym} = S_3\,,$$

and furthermore,

$$\|\Pi_{S\otimes\mathbb{R}^d}\Pi_{\text{sym}}\Pi_{S\otimes\mathbb{R}^d} - \Pi_{S_3}\| \leqslant 1 - \kappa^2\,.$$

Proof Let $W = (\sum_i a_i^{\otimes 2}a_i^{\otimes 2\mathsf{T}})^{-1/2}$ and let T denote the columnspace of $(W^2 \otimes \mathrm{Id})H$. The columns of W^2H form a basis for the subspace of $S \otimes \mathbb{R}^d$ orthogonal to S_3 since each column of W^2H is orthogonal to every $a_i^{\otimes 3}$. Therefore,

$$\Pi_{S\otimes\mathbb{R}^d} = \Pi_{S_3} + \Pi_T\,.$$

Multiplying this with Π_{sym} and itself and then applying the identities $\Pi_{\text{sym}}\Pi_{S_3} = \Pi_{S_3}\Pi_{\text{sym}} = \Pi_{S_3}$ and $\Pi_{S_3}\Pi_T = \Pi_T\Pi_{S_3} = 0$,

$$\Pi_{S\otimes\mathbb{R}^d}\Pi_{\text{sym}}\Pi_{S\otimes\mathbb{R}^d} = \Pi_{S_3} + \Pi_T\Pi_{\text{sym}}\Pi_T\,.$$

Therefore,

$$\|\Pi_{S\otimes\mathbb{R}^d}\Pi_{\text{sym}}\Pi_{S\otimes\mathbb{R}^d} - \Pi_{S_3}\| \leqslant \|\Pi_{\text{sym}}\Pi_T\|^2\,.$$

We would thus like to show that $\|\Pi_{\text{sym}}\Pi_T\|^2 \leqslant 1-\kappa^2$.

Since $\|\Pi_{\text{sym}}\Pi_T\|^2 = \max_{y'\in T}\|\Pi_{\text{sym}}y'\|^2/\|y'\|^2 = 1 - \min_{y'\in T}\|(\mathrm{Id}-\Pi_{\text{sym}})y'\|^2/\|y'\|^2$, it is enough to show that $\min_{y'\in T}\|(\mathrm{Id}-\Pi_{\text{sym}})y'\|/\|y'\| \geqslant \kappa$. By Lemma 12, that is implied by $\|(\mathrm{Id}-\Pi_{\text{sym}})y\|/\|y\| \geqslant \kappa$ for $y \in \mathrm{img}(H)$.

Since $\Pi_{\text{sym}} \preceq \Pi_{2,3}$ where $\Pi_{2,3}$ is the projector to the space invariant under interchange of the second and third modes of $(\mathbb{R}^d)^{\otimes 3}$ and $\Pi_{2,3} = \frac{1}{2}(\mathrm{Id} + P_{2,3})$, we see that $\|(\mathrm{Id} - \Pi_{\text{sym}})y\|/\|y\| \geqslant \|\frac{1}{2}(\mathrm{Id} - P_{2,3})y\|/\|y\|$ for $y \in \mathrm{img}(H)$. Since the columns of R are an orthonormal basis for $\mathrm{img}(H)$, for $x = R^{-1}y$ spanning all of $\mathbb{R}^n$ we have

$$\frac{\|(\mathrm{Id}-P_{2,3})y\|}{2\|y\|} = \frac{\|(\mathrm{Id}-P_{2,3})Rx\|}{2\|Rx\|} = \frac{\|(\mathrm{Id}-P_{2,3})Rx\|}{2\|x\|} = \frac{\|Kx\|}{\|x\|}\,.$$

The expression on the right is the definition of κ. Therefore, $\|(\mathrm{Id}-\Pi_{\text{sym}})y\|/\|y\| \geqslant \|\frac{1}{2}(\mathrm{Id} - P_{2,3})y\|/\|y\| = \kappa$. ∎

Lemma 12 *For each i, let $\{b_{i,j}\}_{j\in[d-1]}$ be an arbitrary orthonormal basis for vectors in $\mathbb{R}^d$ orthogonal to a_i, and let*

$$H^{\mathsf{T}} := \begin{bmatrix} a_1\otimes a_1\otimes b_{1,1} \\ \vdots \\ a_i\otimes a_i\otimes b_{i,j} \\ \vdots \\ a_n\otimes a_n\otimes b_{n,d-1} \end{bmatrix}.$$

Let $H' = (W^2\otimes \mathrm{Id})H$. If $\|(\mathrm{Id}-\Pi_{\text{sym}})y\| \geqslant t\|y\|$ for all $y \in \mathrm{img}(H)$, then $\|(\mathrm{Id}-\Pi_{\text{sym}})y'\| \geqslant t\|y'\|$ for all $y' \in \mathrm{img}(H')$.

Proof

Let $S = \mathrm{Span}(a_i^{\otimes 2})$ and $S_3 = \mathrm{Span}(a_i^{\otimes 3}) \subseteq \mathrm{sym}$. Observe that $\mathrm{img}(H') \subseteq S\otimes\mathbb{R}^d = \mathrm{img}(H) + S_3$. Therefore, for every $y' \in \mathrm{img}(H')$ there will be some $y \in \mathrm{img}(H)$ and some $z \in S_3$ such that $y' = y + z$. Also, since $S_3 \perp \mathrm{img}(H')$, we have $z \perp y'$, and therefore $\|y\| = \|y' - z\| \geqslant \|y'\|$.

So if the premise of the lemma holds and $\|(\mathrm{Id}-\Pi_{\text{sym}})y\| \geqslant t\|y\|$ for all $y \in \mathrm{img}(H)$, it will also be the case that $\|(\mathrm{Id}-\Pi_{\text{sym}})y'\| = \|(\mathrm{Id}-\Pi_{\text{sym}})(y+z)\| \geqslant t\|y\| \geqslant t\|y'\|$. ∎

C.2. Robustness arguments

The main lemma of this section gives all of the spectral eigenspace perturbation arguments needed to argue the correctness and robustness of Algorithm 2. Here we essentially repeatedly apply Davis and Kahan's sin-Θ theorem (Theorem 6) through a sequence of linear algebraic transformations, along with triangle inequality and some adding-and-subtracting, to argue that the desired top eigenspace remains stable against the spectral-norm errors melded in at each step.

Lemma 13 (Subspace perturbation for LIFT) *Let $T = \sum_{i\in[n]} a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}}$ and let $\widetilde{T}$ be a matrix with $\|T - \widetilde{T}\| \leqslant \varepsilon\, \sigma_n^2 \mu^{-1} \kappa^2$ for some $\varepsilon < 1/63$, where σ_n is the nth eigenvalue of T and μ is the operator norm of $\sum \|a_i\|^{-2} a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}$ and κ is the condition number from Lemma 11. Let $S = \mathrm{Span}(\{a_i^{\otimes 2}\}) = \mathrm{img}(T)$ and let $\widetilde{S} = \mathrm{img}(\widetilde{T})$. Also let $S_3 = \mathrm{Span}(\{a_i^{\otimes 3}\})$. Then*

$$\|\operatorname{top}_n \left(\Pi_{\widetilde{S}\otimes\mathbb{R}^d} \Pi_{\mathrm{sym}} \Pi_{\widetilde{S}\otimes\mathbb{R}^d}\right) - \Pi_{S_3}\| \leqslant 18\, \varepsilon\, \sigma_n \mu^{-1}\,,$$

where top_n denotes the top-n eigenspace. Furthermore, letting $\widetilde{S}_3 = \operatorname{top}_n(\Pi_{\widetilde{S}\otimes\mathbb{R}^d} \Pi_{\mathrm{sym}} \Pi_{\widetilde{S}\otimes\mathbb{R}^d})$ and $W = T^{-1/2}$ and $\widetilde{W} = \widetilde{T}^{-1/2}$, we have

$$\|\Pi_{(\widetilde{W}\otimes\mathrm{Id})\widetilde{S}_3} - \Pi_{(W\otimes\mathrm{Id})S_3}\| \leqslant 63\,\varepsilon\,.$$

Proof For brevity, let $\Pi = \Pi_{S\otimes\mathbb{R}^d}$ and let $\widetilde{\Pi} = \Pi_{\widetilde{S}\otimes\mathbb{R}^d}$. We write

$$\widetilde{\Pi}\,\Pi_{\mathrm{sym}}\,\widetilde{\Pi} \;-\; \Pi\,\Pi_{\mathrm{sym}}\,\Pi \;\;=\;\; \left(\widetilde{\Pi} - \Pi\right)\Pi_{\mathrm{sym}}\,\widetilde{\Pi} \;+\; \Pi\,\Pi_{\mathrm{sym}}\left(\widetilde{\Pi} - \Pi\right)\,.$$

Since $\|T - \widetilde{T}\| \leqslant \varepsilon\,\sigma_n^2\mu^{-1}\kappa^2$, by Theorem 6, $\|\widetilde{\Pi} - \Pi\| = \|\Pi_{\widetilde{S}} - \Pi_S\| \leqslant 3\varepsilon\sigma_n\mu^{-1}\kappa^2$. Since projectors don't increase spectral norm, we conclude

$$\|\widetilde{\Pi}\,\Pi_{\mathrm{sym}}\,\widetilde{\Pi} - \Pi\,\Pi_{\mathrm{sym}}\,\Pi\| \leqslant 6\varepsilon\sigma_n\mu^{-1}\kappa^2\,.$$

Furthermore, by Lemma 11, $\Pi\,\Pi_{\mathrm{sym}}\,\Pi = \Pi_{S_3} + Z$, where Z is a symmetric matrix with $\|Z\| \leqslant 1 - \kappa^2$ whose columnspace is orthogonal to S_3 since $\Pi_{S_3}\Pi_{S\otimes\mathbb{R}^d}\Pi_{\mathrm{sym}}\Pi_{S\otimes\mathbb{R}^d} = \Pi_{S_3}$. Therefore,

$$\|\widetilde{\Pi}\Pi_{\mathrm{sym}}\widetilde{\Pi} - (\Pi_{S_3} + Z)\| \leqslant 6\varepsilon\sigma_n\mu^{-1}\kappa^2\,.$$

The top-n eigenspace of $(\Pi_{S_3} + Z)$ is S_3 and the nth and $(n+1)$th eigenvalues of $(\Pi_{S_3} + Z)$ differ by at least κ^2. So by Theorem 6,

$$\|\operatorname{top}_n(\widetilde{\Pi}\Pi_{\mathrm{sym}}\widetilde{\Pi}) - \Pi_{S_3}\| \leqslant 18\varepsilon\sigma_n\mu^{-1}\,.$$

Multiplying by W multiplies this error by at most a factor of $\|W\|^2 = \sigma_n^{-1}$, so that

$$\|(W\otimes\mathrm{Id})\Pi_{\widetilde{S}_3}(W\otimes\mathrm{Id}) - (W\otimes\mathrm{Id})\Pi_{S_3}(W\otimes\mathrm{Id})\| \leqslant 18\,\varepsilon\mu^{-1}\,.$$

And $\|(\widetilde{W}\otimes\mathrm{Id})\Pi_{\widetilde{S}_3}(\widetilde{W}\otimes\mathrm{Id}) - (W\otimes\mathrm{Id})\Pi_{\widetilde{S}_3}(W\otimes\mathrm{Id})\| \leqslant 3\varepsilon$ since $\Pi_{\widetilde{S}_3}(W\otimes\mathrm{Id})$ has a spectral norm at most $\sigma_n^{-1/2}$, so that

$$\|(\widetilde{W}\otimes\mathrm{Id})\Pi_{\widetilde{S}_3}(\widetilde{W}\otimes\mathrm{Id}) - (W\otimes\mathrm{Id})\Pi_{S_3}(W\otimes\mathrm{Id})\| \leqslant 21\,\varepsilon\mu^{-1}\,.$$

By Lemma 14, the smallest eigenvalue of $(W \otimes \mathrm{Id})\Pi_{S_3}(W \otimes \mathrm{Id})$ is at least μ^{-1}. Therefore, by Theorem 6, $\|\Pi_{(W\otimes\mathrm{Id})\widetilde{S}_3} - \Pi_{(W\otimes\mathrm{Id})S_3}\| \leqslant 63\,\varepsilon$. ∎

The following utility lemma is used to reduce the impact of condition numbers on the algorithm. It shows that when multiplying a third-order tensor in the span of $a_i^{\otimes 3}$ by the second-order whitener $W = T^{-1/2} = (\sum_i a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}})^{-1/2}$, the penalty to the error may be expressed in terms of a sixth-order condition number – the spectral norm of $U = \sum \|a_i\|^{-2} a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}$ – instead of the fourth-order one given by T.

The reason this is important is that $\sum_i a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}}$ suffers from spurious directions: directions $v \in \mathbb{R}^{\otimes 2}$ in which Tv may be very large, but v is not close to any of the $a_i \otimes a_i$, or in fact any rank-1 2-tensor at all. For example, for n random Gaussian vectors, the spurious direction is given by $\Phi = \mathbb{E}_{g\sim\mathbb{N}(0,1)}\, g \otimes g$, which will have $\|T\Phi\| \approx n/d$.

The sixth-order object $U = \sum \|a_i\|^{-2} a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}$ does not suffer with this problem for n up to $\tilde{O}(n^2)$, due to cancellation with the odd number of modes. For instance, $U\,\mathbb{E}_{g\sim\mathbb{N}(0,1)}\, g^{\otimes 3} = 0$ and $\|U(\Phi \otimes u)\| \approx n/d^2$ for all unit $u \in \mathbb{R}^d$ and U generated from random Gaussian vectors.

Lemma 14 (Sixth-order condition numbers) *Let $a_1, \ldots, a_n \in \mathbb{R}^d$ with $n \leqslant d^2$. Let $W = (\sum_i a_i^{\otimes 2} a_i^{\otimes 2\mathsf{T}})^{-1/2}$ have rank n. Let U be the matrix $\sum \|a_i\|^{-2} a_i^{\otimes 3} a_i^{\otimes 3\mathsf{T}}$. Then for a vector $v \in \mathrm{Span}(a_i^{\otimes 3})$, the following hold:*

$$\|(W \otimes \mathrm{Id})v\| \leqslant \|U^{-1}\|^{1/2}\|v\|\,,$$

$$\|(W \otimes \mathrm{Id})v\| \geqslant \|U\|^{-1/2}\|v\|\,.$$

Proof Let $v = \sum \mu_i \|a_i\|^{-1} a_i^{\otimes 3}$. Then

$$\|(W \otimes \mathrm{Id})v\|^2 = \sum \mu_i\mu_j \|a_i\|^{-1}\|a_j\|^{-1}\langle W(a_j \otimes a_j), W(a_i \otimes a_i)\rangle\langle a_j, a_i\rangle = \sum \mu_i^2\,,$$

using the fact that $\{W(a_i \otimes a_i)\}_i$ is an orthonormal set of vectors. ∎

Appendix D. Rounding

In this section, we show how to "round" the lifted tensor to extract the components. That is, assuming we are given the tensor

$$T = \sum_{i\in[n]} (W a_i^{\otimes 2})^{\otimes 3} + E$$

where E is a tensor of Frobenius norm at most $\varepsilon\sqrt{n}$, we show how to find the components a_i.

Lemma 15 *Suppose $a_1, \ldots, a_n \in \mathbb{R}^d$ are unit vectors satisfying the* identifiability assumption *from Lemma 9, and suppose we are given an implicit rank-n representation of the tensor $T = \sum_i (W a_i^{\otimes 2})^{\otimes 3} + E \in (\mathbb{R}^{d^2})^{\otimes 3}$, where $\|E\|_F \leqslant \varepsilon\sqrt{n}$, and an implicit rank-$n$ representation of a matrix Π_3 such that $\|\Pi_3 - \sum_i ((W a_i^{\otimes 2}) \otimes a_i)((W a_i^{\otimes 2}) \otimes a_i)^\top\| \leqslant \varepsilon < \frac{1}{2}$.*

Then for any $\beta, \delta \in (0,1)$ so that $\beta\delta = \Omega(\varepsilon)$ and $\delta = \Omega(\varepsilon)$, there is a randomized algorithm that with high probability in time $O(\frac{1}{\beta} n^{1+O(\beta)} d^3)$ with $\tilde{O}(n^2 d^3)$ preprocessing time recovers a unit vector u such that for some $i \in [n]$,

$$\langle a_i, u\rangle^2 \geqslant 1 - \|W\| \cdot O\left(\frac{\varepsilon}{\beta}\right)^{1/8},$$

so long as $\|W\| \left(\frac{\varepsilon}{\beta}\right)^{1/8} < C$ for a universal constant C.

Further, there is an integer $m \geqslant (1-\delta)n$ so that repeating the above algorithm $\tilde{O}(n)$ times recovers unit vectors $u_1, \ldots, u_m$ so that $\langle u_i, a_i\rangle^2 \geqslant 1 - \|W\| \cdot O\left(\frac{\varepsilon}{\delta\beta}\right)^{1/8}$ for all $i \in [m]$ (up to re-indexing), again so long as $\|W\| \left(\frac{\varepsilon}{\delta\beta}\right)^{1/8} < C$, and with a total runtime of $\tilde{O}(\frac{1}{\beta} n^{2+O(\beta)} d^3)$.

We will prove this theorem in four steps. First, in Appendix D.1 we will show how to recover vectors that are (with reasonable probability) correlated with the whitened Kronecker squares of the components, $Wa_i^{\otimes 2}$. In Appendix D.2, we'll give an algorithm that given a vector close to the whitened square $Wa_i^{\otimes 2}$, recovers a vector close to the component a_i. In Appendix D.3, we give an algorithm that tests if a vector $a \in \mathbb{R}^d$ is close to one of the components $\{a_i\}_{i\in[n]}$. In these first three sections, we omit runtime details; in Appendix D.4 we put the arguments together and address runtime details as well.

D.1. Recovering candidate whitened and squared components

Here, we give an algorithm for recovering components that have constant correlation with the $Wa_i^{\otimes 2}$. In this subsection, our result applies in generality to arbitrary orthonormal vectors $b_1, \ldots, b_n \in \mathbb{R}^{d^2}$. The algorithm and its analysis follow almost directly from Schramm and Steurer (2017); for completeness we re-state the important lemmas here, and detail what little adaptation is necessary.

Lemma 16 *Suppose that $b_1, \ldots, b_n \in \mathbb{R}^{d^2}$ are orthonormal. Then if $T = \sum_{i\in[n]} b_i^{\otimes 3} + E$ for a tensor E with $\|E\|_F \leqslant \varepsilon\sqrt{n}$ and $\delta = \Omega(\varepsilon)$, $\Omega(\frac{\varepsilon}{\delta}) \leqslant \beta < 1$, repeating steps 2 & 3 of Algorithm 3 $\tilde{O}(n^{O(\beta)})$ times will with high probability recover a unit vector u such that $\langle u, b_i\rangle^2 \geqslant 1 - \frac{\varepsilon}{\delta\beta}$ for some $i \in [n]$. Furthermore, repeating steps 2 & 3 of Algorithm 3 $\tilde{O}(n^{1+O(\beta)})$ times will with high probability recover $m \geqslant (1-\delta)\cdot n$ unit vectors $u_1, \ldots, u_m$ such that for each u_i there exists $j \in [n]$ so that $\langle u_i, b_j\rangle^2 \geqslant 1 - \frac{\varepsilon}{\delta\beta}$.*[9]

The proof follows from two lemmas:

Lemma 17 *The tensor $T^{\leqslant 1}$ computed in step 1 of Algorithm 3 remains close to $S = \sum_i b_i^{\otimes 3}$ in Frobenius norm, $\|T^{\leqslant 1} - S\|_F \leqslant \varepsilon\sqrt{n}$, and furthermore*

$$\|T^{\leqslant 1}_{\{1,2\}\{3\}}\| \leqslant 1 \qquad \text{and} \qquad \|T^{\leqslant 1}_{\{1,3\}\{2\}}\| \leqslant 1.$$

9. In particular, if we choose $\delta = \log\log n \cdot \varepsilon$, we will will recover all but $\varepsilon \cdot n \log\log n$ of the b_i in $\tilde{O}(n)$ repetitions.

Algorithm 3 Rounding to a whitened component

Function ROUND(T, β, ε):
Input: a tensor $T \in (\mathbb{R}^{d^2})^{\otimes 3}$, a spectral gap bound β, and an error tolerance ε.

1. Decrease the spectral norm of the error term in rectangular reshapings:
   (a) compute T', the projection of $T_{\{1,2\}\{3\}}$ to O, the set of $d^4 \times d^2$ matrices with spectral norm at most 1
   (b) compute $T^{\leqslant 1}$, the projection of $T'_{\{1,3\}\{2\}}$ to O (may be done up to $\varepsilon\sqrt{n}$ Frobenius norm error).
2. Compute a random flattening of $T^{\leqslant 1}$ along the $\{1\}$ mode: for $g \sim \mathcal{N}(0, \mathrm{Id}_{d^2})$, compute
$$T(g) = \sum_{i \in [d^2]} g_i \cdot T(i, \cdot, \cdot).$$
3. Recover candidate component vectors: compute $u_L(g)$ and $u_R(g)$, the top left- and right-singular vectors of $T(g)$ using $O(\frac{1}{\beta}\log d)$ steps of power iteration.

Output: the candidate components $u_L(g)$ and $u_R(g)$.

The proof of Lemma 17 is identical to the proof of (Schramm and Steurer, 2017, Lemma 4.5), and uses the fact that distances decrease under projection to convex sets to control the error, and the fact that the truncation operation is equivalent to multiplication by a contractive matrix to argue that $T^{\leqslant 1}$ has bounded norm in both reshapings.

Lemma 18 *Suppose that in spectral norm $\|T_{\{1,2\}\{3\}}\|, \|T_{\{1,3\}\{2\}}\| \leqslant 1$, and also that $\|T - \sum_i b_i^{\otimes 3}\|_F \leqslant \varepsilon\sqrt{n}$. Let $T(g)$ be the random flattening of T produced in step 2 of Algorithm 3, and let $u_L(g)$ and $u_R(g)$ be the top left- and right-signular vectors of $T(g)$ respectively. Then there is a universal constant C such that for any $\delta > C \cdot \varepsilon$ and $\Omega(\frac{\varepsilon}{\delta}) \leqslant \beta < 1$, for a $1 - \delta$ fraction of $j \in [n]$,*

$$\mathop{\mathbb{P}}_{g \sim N(0,\mathrm{Id})}\left(\langle u_L(g), b_j\rangle^2 \geqslant 1 - \frac{\varepsilon}{\delta\beta} \quad \text{or} \quad \langle u_R(g), b_j\rangle^2 \geqslant 1 - \frac{\varepsilon}{\delta\beta}\right) \geqslant \tilde{\Omega}\left(n^{-1-O(\beta)}\right),$$

and further when this event occurs the ratio of the first and second singular values of $T(g)$ is lower bounded by β, $\frac{\sigma_1(T(g))}{\sigma_2(T(g))} \geqslant 1 + \beta$.

Proof By assumption, $T^{\leqslant 1} = S + E$ for $S = \sum_{i \in [n]} b_i^{\otimes 3}$, and E is a tensor of Frobenius norm at most $\varepsilon\sqrt{n}$ and spectral norms $\|E_{\{1,2\}\{3\}}\| \leqslant 1$ and $\|E_{\{1,3\}\{2\}}\| \leqslant 1$. For $g \sim \mathcal{N}(0, \Sigma^{-1})$, we have

$$T(g) = S(g) + E(g) = \left(\sum_{k \in [n]} \langle g, b_k\rangle \cdot b_k b_k^\top\right) + \left(\sum_{i \in [d^2]} g_i \cdot E_i\right),$$

where we use $E_i = E(i, \cdot, \cdot)$ to refer to the $d^2 \times d^2$ matrix given by taking the $\{2\}, \{3\}$ flattening of E restricted to coordinate i in mode 1.

The proof of the lemma is now identical to that of (Schramm and Steurer, 2017, Lemma 4.6 and Lemma 4.7). There are two primary differences: the first is that in Schramm and Steurer (2017) the tensor has four modes, and our tensor effectively has 3 modes. This difference is negligible, since in Schramm and Steurer (2017), two of the four modes are always identified anyway.

The second difference is that we choose parameters differently. We take the parameter β appearing in (Schramm and Steurer, 2017, Lemma 4.6) so that $\beta = \Omega(\frac{\varepsilon}{\delta})$[10]; this is to emphasize that for small $\varepsilon \ll \frac{1}{n}$, one can recover all $m = n$ of the components. Because the proof is otherwise the same, we merely sketch an overview here.

The first term in isolation is a random flattening of an orthogonal tensor, and so with probability 1 the eigenvectors of the first term are precisely the b_k. The second term, which is the flattening of the noise term, introduces complications; however, the combination of the spectral norm bound and the Frobenius norm bound on E is enough to argue (using a matrix Bernstein inequality, Markov's inequality and the orthogonality of the b_i) that the random flattening of E cannot have spectral norm larger than ε/δ in more than $1 - \delta$ of the b_k's directions.

To finish the proof, we perform a large deviation analysis on the coefficients $\langle g, b_k\rangle$, lower bounding the probability that for the $1 - \delta$ fraction of the b_k that are not too aligned with the spectrum of E, there is a sufficiently large gap between $\langle g, b_k\rangle$ and the $\langle g, b_i\rangle$ for $i \neq k$ so that b_k is correlated with the top singular vectors of $T(g)$.[11] The bound on the ratio of the singular values comes from (Schramm and Steurer, 2017, Lemma 4.7) as well. ■

Proof [Proof of Lemma 16] The proof simply follows by applying Lemma 17, then Lemma 18. ■

D.2. Extracting components from the whitened squares

We now present the following simple algorithm which recovers a vector close to a_i, given a vector close to $W(a_i^{\otimes 2})$. For convenience we will again work with generic orthonormal vectors b_i in place of the $W(a_i^{\otimes 2})$, and we will assume we have access to the matrix Π_3 (the approximate projector to $\mathrm{Span}\{W(a_i^{\otimes 2}) \otimes a_i\}$) computed in Algorithm 2.

Lemma 19 *Suppose $b_1, \ldots, b_n \in \mathbb{R}^{d^2}$ are orthonormal vectors and $a_1, \ldots, a_n \in \mathbb{R}^d$, and $\Pi_3 \in \mathbb{R}^{d^3 \times d^3}$ is such that $\|\Pi_3 - \sum_i b_i b_i^\top \otimes a_i a_i^\top\| \leqslant \varepsilon$. Then if $u \in \mathbb{R}^{d^2}$ is a unit vector with $\langle u, b_i\rangle^2 \geqslant 1 - \theta$ for $\theta < \frac{1}{10}$, then the output $a \in \mathbb{R}^d$ of Algorithm 4 on u has the property that $|\langle a, a_i\rangle| \geqslant 1 - 4\theta^{1/4} - 4\sqrt{\varepsilon}$.*

Proof Let $P_3 = \sum_i b_i b_i^\top \otimes a_i a_i^\top$. By assmption we can write the approximate projector $\Pi_3 = P_3 + E$, for a matrix E of spectral norm $\|E\| \leqslant \varepsilon$. Based on these expressions we can

10. We comment that the parameter c appearing in the statement of (Schramm and Steurer, 2017, Lemma 4.6) is larger than $\sqrt{2}$; this is necessary for the application of (Schramm and Steurer, 2017, Lemma 4.7), and is not clear from the lemma statement but is implicit in the proof.
11. We note that to obtain correlation $1 - \frac{\varepsilon}{\delta\beta}$, one must directly use the proof of (Schramm and Steurer, 2017, Lemma 4.7), rather than the statement of the lemma (which has assumed that $\frac{2\varepsilon(1+\beta)}{\beta\delta} \leqslant 0.01$, and replaced the expression $1 - \frac{2\varepsilon(1+\beta)}{\beta\delta}$) with the lower bound 0.99).

Algorithm 4 Extracting the component from the whitened square

Function EXTRACT(u, Π_3):
Input: a unit vector $u \in (\mathbb{R}^d)^{\otimes 2}$ such that $\langle u, b_i\rangle^2 \geqslant 1-\theta$ for some $i \in [n]$, and a projector $\Pi_3 \in \mathbb{R}^{d^3 \times d^3}$ such that $\|\Pi_3 - \sum_i b_i b_i^\top \otimes a_i a_i^\top\| \leqslant \varepsilon$.

1. Compute the matrix $M = \Pi_3(uu^\top \otimes \mathrm{Id})$.
2. Compute the top-left singular vector v of M.
3. Taking the reshaping $V = v_{\{3\}\{1,2\}}$, let $a = Vu$.

Output: the vector $a \in \mathbb{R}^d$

re-express the product,

$$M = \Pi_3(uu^\top \otimes \mathrm{Id}) = P_3(uu^\top \otimes \mathrm{Id}) + E(uu^\top \otimes \mathrm{Id}).$$

By assumption, the second term is a matrix of spectral norm at most $\|E\| \leqslant \varepsilon$.

We now consider the first term. If $u = c \cdot b_i + w$, then for the first term we have

$$P_3(uu^\top \otimes \mathrm{Id}) = P_3\left(c^2 \cdot b_i b_i^\top \otimes \mathrm{Id}\right) + P_3\left((c \cdot b_i w^\top + c \cdot w b_i^\top + w w^\top) \otimes \mathrm{Id}\right)$$

The second term is again a matrix of spectral norm at most $3c \cdot \|w\| = 3c \cdot \sqrt{1-c^2}$. The first term can be further simplified as

$$P_3(c^2 \cdot b_i b_i^\top \otimes \mathrm{Id}) = c^2 \cdot \sum_i \langle b_i, b_i\rangle b_i b_i^\top \otimes a_i a_i^\top = c^2 \cdot b_i b_i^\top \otimes a_i a_i^\top,$$

by the orthogonality of the b_i. This is a rank-1 matrix with singular value c^2. Therefore, $M = c^2(b_i \otimes a_i)(b_i \otimes a_i)^\top + \tilde{E}$ where $\|\tilde{E}\| \leqslant \varepsilon + 3c\sqrt{1-c^2}$. It follows from Lemma 20 that if v is the top unit left-singular vector of M, then $\langle v, b_i \otimes a_i\rangle^2 \geqslant 1 - \frac{2}{c^2}\|\tilde{E}\|$.

Now, in step 3 when we re-shape v to a $d \times d^2$ matrix V of Frobenius norm 1, because v is a unit vector we have that $V = a_i b_i^\top + \tilde{V}$ for $\tilde{V}$ of spectral norm $\|\tilde{V}\| \leqslant \|\tilde{V}\|_F \leqslant \sqrt{\frac{2}{c^2}\|\tilde{E}\|}$. Therefore,

$$Vu = (a_i b_i^\top)(c \cdot b_i + w) + \tilde{V}u = c(1 - \langle w, b_i\rangle) \cdot a_i + \tilde{V}u,$$

and the latter vector has norm at most $\|\tilde{V}\|$, and $\langle w, b_i\rangle \leqslant \|w\| \leqslant \sqrt{1-c^2}$. Finally, substituting $c = \sqrt{1-\theta}$ and using our bound on $\|\tilde{E}\|$ and $\|\tilde{V}\|$ and some algebraic simplifications, the conclusion follows. ∎

Lemma 20 *Suppose that $M = uv^\top + E$ for $u \in \mathbb{R}^d, v \in \mathbb{R}^k$ unit vectors and $E \in \mathbb{R}^{d\times k}$ a matrix of spectral norm $\|E\| \leqslant \varepsilon$. Then if x, y are the top left- and right-singular vectors of M, $|\langle x, u\rangle|, |\langle y, v\rangle| \geqslant 1 - 2\varepsilon$.*

Proof Let $M = \sum_i \sigma_i x_i y_i^\top$ be the singular value decomposition of M, with $\sigma_1 \geqslant \cdots \geqslant \sigma_d$. We have that

$$1 - \varepsilon \leqslant u^\top M v \leqslant \sigma_1.$$

On the other hand, if with $\langle x_1, u\rangle = \alpha \leqslant 1$ and $\langle y_1, v\rangle = \beta \leqslant 1$,

$$\sigma_1 = x_1^\top M y_1 \leqslant \alpha\beta + \varepsilon.$$

Therefore,

$$|\alpha|, |\beta| \geqslant \alpha\beta ge 1 - 2\varepsilon,$$

and thus $\min\{|\alpha|, |\beta|\} \geqslant 1 - 2\varepsilon$. ■

D.3. Testing candidate components

The following algorithm allows us to test whether a candidate component u is close to some component a_i.

Algorithm 5 Testing component membership

Function $\textsc{test}(u, \theta, \Pi_{S_3})$:
Input: A unit vector $\hat{u}$, and the correlation parameter θ. Also, Π_3, an approximate projector to $\mathrm{Span}\{(Wa_i^{\otimes 2}) \otimes a_i\}$.

1. Compute $\rho = ((W\hat{u}^{\otimes 2}) \otimes \hat{u})$.
2. If $\|\Pi_3\rho\|_2^2 < (1-\theta)\|\rho\|_2^2$, return FALSE. Otherwise, return TRUE.

Lemma 21 *Let P_3 be the projector to $\mathrm{Span}\{(Wa_i^{\otimes 2}) \otimes a_i\}$, and suppose that we have Π_3 such that $\|\Pi_3 - P_3\| \leqslant \varepsilon < \frac{1}{2}$. Then if Algorithm 5 is run on a vector $\hat{u}$ such that $\langle \hat{u}, a_i\rangle^2 \leqslant 1 - \theta - 2\varepsilon$ for all $i \in [n]$, then Algorithm 5 returns* FALSE.

Converseley, if Algorithm 5 is run on a vector $\hat{u}$ with $\langle \hat{u}, a_i\rangle^2 \geqslant 1 - \left(\frac{\theta-\varepsilon}{10\|W\|}\right)^2 \geqslant 1 - \frac{1}{10}$ for some $i \in [n]$, then when run on a unit vector $\hat{u}$, Algorithm 5 returns TRUE.

Proof By assumption, we can write $\Pi_3 = P_3 + E$ for P_3 the projector to $\mathrm{Span}\{(Wa_i^{\otimes 2}) \otimes a_i\}$ and E a matrix of spectral norm at most ε. From this, we have

$$\Pi_3(W\hat{u}^{\otimes 2}) \otimes \hat{u} = P_3(W\hat{u}^{\otimes 2}) \otimes \hat{u} + E(W\hat{u}^{\otimes 2}) \otimes \hat{u}, \tag{D.1}$$

and $\|E(W\hat{u}^{\otimes 2}) \otimes \hat{u}\| \leqslant \varepsilon \|W\hat{u}^{\otimes 2}\|$. Now, we can write $W\hat{u}^{\otimes 2} = \sum_i c_i W(a_i \otimes a_i) + e$, where e is orthogonal to $\mathrm{Span}\{Wa_i^{\otimes 2}\}$, and we can further write

$$(W\hat{u}^{\otimes 2}) \otimes \hat{u} = \sum_{i \neq j} c_i\gamma_j \cdot (Wa_i^{\otimes 2}) \otimes b_j^{(i)} + \sum_i c_i\gamma_i \cdot (Wa_i^{\otimes 2}) \otimes a_i + e \otimes \hat{u},$$

where $\{b_j^{(i)}\}_{j \neq i}$ is an orthogonal basis for the orthogonal complement of a_i in $\mathbb{R}^d$. By definition, $P_3(Wa_i^{\otimes 2}) \otimes b_j^{(i)} = 0$, as this is orthogonal to every vector in $\mathrm{Span}\{(Wa_i^{\otimes 2}) \otimes a_i\}$. Therefore,

$$P_3(W\hat{u}^{\otimes 2}) \otimes \hat{u} = \sum_i c_i\gamma_i \cdot (Wa_i^{\otimes 2}) \otimes a_i.$$

Now, if $\langle \hat{u}, a_i \rangle^2 \leqslant \tau$ for all $i \in [n]$, then $\gamma_i^2 \leqslant \tau$ for all $i \in [n]$. It thus follows that $\|P_3(W\hat{u}^{\otimes 2}) \otimes \hat{u}\|^2 \leqslant \max_i \gamma_i^2 \cdot \sum_j c_j^2 \leqslant \tau \cdot \|W\hat{u}^{\otimes 2}\|_2^2$. Combining this with Eq. (D.1), we have that

$$\|\Pi_3(W\hat{u}^{\otimes 2}) \otimes \hat{u}\|_2^2 \leqslant (\tau + \varepsilon\sqrt{\tau} + \varepsilon^2) \cdot \|W\hat{u}^{\otimes 2}\|_2^2 \leqslant (\tau + 2\varepsilon)\|W\hat{u}^{\otimes 2}\|_2^2,$$

for $\varepsilon < \frac{1}{2}$. It follows that if $\langle \hat{u}, a_i \rangle^2 = \tau < 1 - \theta - 2\varepsilon$ for all $i \in [n]$, then the algorithm returns FALSE.

Converseley, if without loss of generality $\hat{u} = \zeta \cdot a_1 + \hat{e}$ for $\hat{e} \in \mathbb{R}^d$ orthogonal to a_1, then $W\hat{u}^{\otimes 2} = \zeta^2 \cdot W a_1^{\otimes 2} + We'$ with $\|We'\|_2^2 \leqslant (1 - \zeta^2) \cdot \|W\|^2$. Measuring the correlation of $W\hat{u}^{\otimes 2}$ with $Wa_1^{\otimes 2}$, we have that $c_1 \geqslant \zeta^2$. Also $\gamma_1 \geqslant \zeta$, which implies

$$P_3(W\hat{u}^{\otimes 2}) \otimes \hat{u} = \zeta^3 \cdot (Wa_1^{\otimes 2}) \otimes a_1 + \tilde{e}.$$

where $\tilde{e}$ is a leftover term with $\|\tilde{e}\| \leqslant \sqrt{1-\zeta^2} \cdot \|W\hat{u}^{\otimes 2}\| + \zeta^2\sqrt{1-\zeta^2}$ (where we have used the PSDness of W). Combining this with Eq. (D.1),

$$\Pi_3(W\hat{u}^{\otimes 2}) \otimes \hat{u} = \zeta^3(Wa_1^{\otimes 2}) \otimes a_1 + \tilde{e} + E(W\hat{u}^{\otimes 2}) \otimes \hat{u}.$$

For convenience let $\hat{\rho} = \tilde{e} + E(W\hat{u}^{\otimes 2}) \otimes \hat{u}$; from our previous observations, we have $\|\hat{\rho}\| \leqslant (\varepsilon + \sqrt{1-\zeta^2})\|W\hat{u}^{\otimes 2}\| + \zeta^2\sqrt{1-\zeta^2}$.

Now, if $\langle \hat{u}, a_1 \rangle^2 = \zeta^2 \geqslant 1 - \eta$, we have that

$$(1-\eta) - \sqrt{\eta}\|W\| \leqslant \|W\hat{u}^{\otimes 2}\| \leqslant (1-\eta) + \sqrt{\eta}\|W\|.$$

From this,

$$\begin{aligned}
\|\Pi_3(W\hat{u}^{\otimes 2})\| \geqslant \zeta^3 - \|\hat{\rho}\| &\geqslant (1-\eta)^{3/2} - (\varepsilon + \sqrt{\eta})\|W\hat{u}^{\otimes 2}\| - (1-\eta)\sqrt{\eta} \\
&\geqslant \sqrt{1-\eta}(\|W\hat{u}^{\otimes 2}\| - \sqrt{\eta}\|W\|) - (\varepsilon + \sqrt{\eta})\|W\hat{u}^{\otimes 2}\| - (1-\eta)\sqrt{\eta} \\
&\geqslant (1 - \varepsilon - 2\sqrt{\eta})\|W\hat{u}^{\otimes 2}\| - 2\sqrt{\eta}\|W\|, \\
&\geqslant (1 - \varepsilon - 5\sqrt{\eta}\|W\|)\|W\hat{u}^{\otimes 2}\|.
\end{aligned}$$

where we have used that $\eta < \frac{1}{10}$. Thus, if $\eta < \left(\frac{\theta - \varepsilon}{10\|W\|}\right)^2$, Algorithm 5 does not return FALSE. ■

D.4. Putting things together

Finally, we prove Lemma 15.

Proof [Proof of Lemma 15] By the assumptions of the theorem, we have access to an implicit rank-n representation of $T = \sum_{i\in[n]}(W(a_i^{\otimes 2}))^{\otimes 3} + E \in (\mathbb{R}^{d^2})^{\otimes 3}$, where $W = \left(\sum_{i\in[n]}(a_i^{\otimes 2})(a_i^{\otimes 2})^\top\right)^{-1/2}$, and with $\|T - E\|_F \leqslant \varepsilon\sqrt{n}$. For convenience we denote $b_i = W(a_i^{\otimes 2})$. Note that the b_i are orthonormal vectors in $\mathbb{R}^{d^2}$. We also have implicit access to a rank-n representation of Π_3, where $\|\Pi_3 - \sum_i (b_i \otimes a_i)(b_i \otimes a_i)^\top\| \leqslant \varepsilon$.

We first run step 1 of Algorithm 3 to produce the tensor which we will round. Then, for $\ell = \tilde{O}(n^{1+O(\beta)})$ independent iterations, we run steps 2 & 3 of Algorithm 3 to produce

candidate whitened squares $u_1, \dots, u_\ell$, then run Algorithm 4 on the u_i to produce candidate components $\hat{u}_i$, and finally run Algorithm 5 to check if $\hat{u}_i$ is close to a_j for some $j \in [n]$.

We show that step 1 of Algorithm 3 takes time $\tilde{O}(n^2 d^3)$. Since T is at most $\varepsilon\sqrt{n}$ in Frobenius norm away from a tensor that is a rank-n projector in both rectangular reshapings $T_{\{1,2\},\{3\}}$ and $T_{\{2,3\},\{1\}}$, the $(2n)$th singular values in either reshaping must be at most ε: otherwise the error term would have over n singular values more than ε and therefore Frobenius norm more than $\varepsilon\sqrt{n}$. Also $\|T\|_F = \sqrt{n}$ because it is a rank-n projector in its square matrix reshaping. Therefore, by Lemma 8, step 1 requires time $\tilde{O}(n^2 d^3 + n(nd^3 + nd^2))$ to return an $\varepsilon\sqrt{n}$-approximation in Frobenius norm to the projected matrix.[12] Note that this step only needs to be carried out once regardless of how many times the algorithm is invoked for a specific input T, so the $\tilde{O}(n^2 d^3)$ runtime is incurred as a preprocessing cost.

Then, again by Lemma 8, steps 2 & 3 require time $\tilde{O}(\frac{1}{\beta} n d^3)$, since the ratio of the first and second singular values of the the matrix is $1 + \Omega(\beta)$, and since $O(\frac{1}{\beta}\log d)$ steps of power iteration with $T(g)$ can be implemented by choosing the random direction $g \sim \mathcal{N}(0, \mathrm{Id}_{d^2})$, the starting direction $v_1 \in \mathbb{R}^{d^2}$, and then computing $v_{t+1} = (\mathrm{Id} \otimes g^\top \otimes v_t)T^{\leqslant 1}$ where $T^{\leqslant 1}$ is the truncated tensor.

Thus, if we choose β, δ satisfying the requirements of Lemma 16, after $\tilde{O}(n^{O(\beta)})$ iterations of steps 2 & 3 we will recover a vector $u \in \mathbb{R}^{d^2}$ such that $\langle u, b_i\rangle^2 \geqslant 1 - 3\frac{\varepsilon}{\beta}$, and after $\tilde{O}(n^{1+O(\beta)})$ iterations of steps 2 & 3 we will recover vectors $u_{t_1}, \dots, u_{t_m}$ so that $\langle u_{t_i}, b_i\rangle^2 \geqslant 1 - 3\frac{\varepsilon}{\beta\delta}$ for $m \geqslant (1-\delta)n$ of the $i \in [n]$.

Next, applying Lemma 19 to each of the good candidate vectors obtained in Algorithm 3, Algorithm 4 will give us candidate components $\hat{u}_{t_1}, \dots, \hat{u}_{t_m}$ so that $\langle \hat{u_{t_i}}, a_i\rangle^2 \geqslant 1 - 4\sqrt{\varepsilon} - 4\left(\frac{3\varepsilon}{\delta\beta}\right)^{1/4}$. Since Π_3 has rank n, we write it as UU^T for $U \in \mathbb{R}^{d^3 \times n}$. Then we may reshape $(uu^\mathsf{T} \otimes \mathrm{Id})\Pi_3$ as $uu^\mathsf{T}U'(U^\mathsf{T} \otimes \mathrm{Id})$, where U' is the $d^2 \times nd$ reshaping of U. Multiplying u^T through takes $O(nd^3)$ time and then reshaping the result back results in $(uu^\mathsf{T} \otimes \mathrm{Id})\Pi_3$. Therefore, by Lemma 7, each invocation of Algorithm 4 requires $\tilde{O}(nd^3)$ operations.

Finally, from Lemma 21, we know that if we run Algorithm 5 with $\theta = 10\|W\| \cdot \left(2\varepsilon^{1/4} + 2\left(\frac{3\varepsilon}{\delta\beta}\right)^{1/8}\right) + 2\varepsilon$, we will reject any $\hat{u}$ such that $\langle \hat{u}, a_i\rangle^2 \leqslant 1 - \theta - 2\varepsilon$ for all $i \in [n]$, and will keep all of the good outputs of Algorithm 4. Each iteration of Algorithm 5 requires time $O(d^4 + nd^3 + d^3)$, since we form the vector $(W\hat{u}^{\otimes 2}) \otimes \hat{u}$, then multiply with the rank-n matrix Π_3, and ultimately compute a norm.

This completes the proof. ■

D.5. Cleaning

Lemma 22 *Suppose there is a set of indices J with $|J| = m$ and let $\mathcal{A} \subseteq \{a_i \mid i \in J\}$. Let $S_{\mathcal{A}^k} = \mathrm{Span}(a^{\otimes k} \mid a \in \mathcal{A})$ for any k and let $\Pi_{\mathcal{A}^k}$ be the projector to $S_{\mathcal{A}^k}$. Let $S = \mathrm{Span}(a_i^{\otimes 2})$ and $S_3 = \mathrm{Span}(a_i^{\otimes 3})$.*

12. Some of the lemmas we apply, out of concerns for compatibility with Schramm and Steurer (2017), assume that the maximum singular value of $T^{\leqslant 1}$ is at most 1. Though one could re-do the previous analysis with minimal consequences under the assumption that the spectral norm is at most $1 + \varepsilon$, for brevity we note that we may instead multiply the whole tensor by $\frac{1}{1-\varepsilon}$, and because the tensor has Frobenius norm at most $(1 + 3\varepsilon)\sqrt{n}$, this costs at most $4\varepsilon\sqrt{n}$ additional Frobenius norm error.

Suppose Π *is a projector such that* $\|\Pi - \Pi_{\mathcal{A}^2}\| \leqslant \delta$ *and* $\|\Pi(\mathrm{Id} - \Pi_S)\| \leqslant \varepsilon_2$. *Suppose* Π_3 *is a projector such that* $\|\Pi_3 - \Pi_{S_3}\| \leqslant \varepsilon_3$. *Let* $\Pi' = \mathrm{top}_m[(\Pi \otimes \mathrm{Id})\Pi_3(\Pi \otimes \mathrm{Id})]$. *Then* $\|\Pi' - \Pi_{\mathcal{A}^3}\| \leqslant ?$.

As a consequence, if we have access to unit vectors u_i *such that* $\langle u_i, \widetilde{W}(a_i^{\otimes 2})\rangle \geqslant 1 - \gamma$, *we obtain* v_i *such that* $\langle v_i, W(a_i^{\otimes 2})\rangle \geqslant 1 - ?$ *and furthermore,* $\|\sum v_i v_i^{\mathsf{T}} - \Pi_{\mathcal{A}^2}\| \leqslant ?$.

Proof Since $\{a_i \otimes a_i\}$ is linearly independent, take $b_i = \sum_j \alpha_{ij} a_j \otimes a_j$ so that

$$\Pi - \Pi_{\mathcal{A}^2} = E + \sum \beta_i b_i b_i^{\mathsf{T}} = E + \sum_i \beta_i \sum_{j,k \in J} \alpha_{ij}\alpha_{ik}(a_j \otimes a_j)(a_k \otimes a_k)^{\mathsf{T}},$$

where E has rank up to $2m$ and $\|E\| \leqslant 2\varepsilon^2$.

■

Appendix E. Combining LIFT and ROUND for final algorithm

In this section we describe and analyze our final tensor decomposition algorithm, proving our main theorem.

Algorithm 6 Main algorithm for overcomplete 4-tensor decomposition

Function DECOMPOSE(T):
Input: a tensor $T \in (\mathbb{R}^d)^{\otimes 4}$, numbers $\beta, \delta, \varepsilon \in (0,1)$, numbers $\sigma, \kappa_0 \in \mathbb{R}_{\geqslant 0}$, and $n \leqslant d^2$.

1. Run LIFT(T, n) from Algorithm 2 to obtain an implicit tensor T' and an implicit matrix Π_3, using σ, κ_0 as upper bounds on condition numbers σ_n, κ.
2. Run the algorithm specified by Lemma 15 on input $(T', \Pi_3, \varepsilon, \beta, \delta)$ with independent randomness $t = \tilde{O}(n)$ times, to obtain vectors $u_1, \ldots, u_t$.

Output: $u_1, \ldots, u_t$

Definition 23 (Signed Hausdorff distance) *For sets of vectors* $a_1, \ldots, a_n \in \mathbb{R}^d$ *and* $b_1, \ldots, b_m \in \mathbb{R}^d$, *we define the signed Hausdorff distance to be the maximum of the following two quantities. (1)* $\max_{i\in[n]} \min_{j\in[m],\sigma\in\pm 1} \|a_i - \sigma b_j\|$ *and (2)* $\max_{i\in[m]} \min_{j\in[n],\sigma\in\pm 1} \|b_i - \sigma a_j\|$.

Definition 24 (Condition number of $a_1, \ldots, a_n$) *Let* $a_1, \ldots, a_n \in \mathbb{R}^d$. *Let* $\{b_{ij}\}_{j\in[d-1]}$ *be an arbitrary orthonormal basis for the orthogonal complement of* a_i *in* $\mathbb{R}^d$. *Let*

$$H^{\mathsf{T}} := \begin{bmatrix} a_1 \otimes a_1 \otimes b_{1,1} \\ \vdots \\ a_i \otimes a_i \otimes b_{i,j} \\ \vdots \\ a_n \otimes a_n \otimes b_{n,d-1} \end{bmatrix}.$$

Let $R = (HH^\mathsf{T})^{-1/2}H$ be a column-wise orthonormalization of H, and let $K = \frac{1}{2}(\mathrm{Id} - P_{2,3})R$, where $P_{2,3}$ is the permutation matrix that exchanges the 2nd and 3rd modes of $(\mathbb{R}^d)^{\otimes 3}$. The condition number κ of $a_1, \ldots, a_n$ is the minimum singular value of K.

Theorem 25 *For every $d, n \in \mathbb{N}$ and $\varepsilon, \beta, \delta \in (0,1)$ and $\sigma, \kappa_0 \in \mathbb{R}_{\geqslant 0}$ there is a randomized algorithm $\textsc{decompose}_{d,n,\varepsilon,\beta,\delta,\sigma,\kappa_0}(T)$ with the following guarantees. For every set of unit vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ and every $E \in (\mathbb{R}^d)^{\otimes 4}$ such that*

1. *the operator norm of the square matrix flattening of E satisfies $\frac{\|E_{12;34}\|}{\sigma_n^7 \mu^{-1} \kappa^2} \leqslant \varepsilon$,*
2. *$\kappa = \kappa(a_1, \ldots, a_n) \geqslant \kappa_0$*
3. *$\sigma_n \geqslant \sigma$*

where

1. *σ_n is the n-th singular value of the matrix $\sum_{i \leqslant n} (a_i^{\otimes 2})(a_i^{\otimes 2})^\mathsf{T}$,*
2. *μ is the operator norm of $\sum_{i \leqslant n} (a_i^{\otimes 3})(a_i^{\otimes 3})^\mathsf{T}$, and*
3. *κ is the condition number of $a_1, \ldots, a_n$ as in Definition 24.*

there is a subset $S \subseteq \{a_1, \ldots, a_n\}$ of size $|S| \geqslant (1-\delta)n$ such that given input $T = \sum_{i \leqslant n} a_i^{\otimes 4} + E$ the algorithm produces a set $B = \{b_1, \ldots, b_t\}$ of $t = \tilde{O}(n)$ vectors which with probability at least 0.99 over the randomness in the algorithm has

$$\textsc{signed-Hausdorff-distance}(S, B) \leqslant O\left(\frac{\varepsilon}{\delta\beta}\right)^{1/16}.$$

Furthermore, the algorithm $\textsc{decompose}_{d,n,\varepsilon,\beta,\delta,\sigma,\kappa_0}$ runs in time

$$\tilde{O}\left(\frac{nd^4}{\sqrt{\sigma}} + \frac{n^2 d^3}{\kappa_0} + \frac{n^{2+O(\beta)} d^3}{\beta}\right).$$

We record some intuitive explanations of the parameters in Theorem 25.

- σ, κ_0 are bounds on the minimum singular values of matrices associated to $a_1, \ldots, a_n$, used to determine the necessary precision of linear-algebraic manipulations performed by the algorithm. Decreasing σ, κ_0 yields an algorithm tolerating less well-conditioned tensors, at the expense of running time and/or accuracy guarantees.
- δ determines what fraction of the vectors $a_1, \ldots, a_n$ the algorithm is allowed to fail to return. By decreasing δ the algorithm recovers a larger fraction of $a_1, \ldots, a_n$, at the cost of increasing running time and/or decreasing per-vector accuracy.
- β determines the per-vector accuracy of the algorithm. Increasing β improves the accuracy of the algorithm, but with exponential cost in the running time.
- ε governs the magnitude of allowable noise E. Increasing ε yields a more noise-tolerant algorithm, at the expense of the accuracy of recovered vectors.

We record the following corollary, which follows from Theorem 25 by choosing parameters appropriately.

Corollary 26 *For every $n, d \in \mathbb{N}$ and $\sigma > 0$ (independent of n, d) there is an algorithm with the following guarantees. The algorithm takes input $T = \sum_{i \leqslant n} a_i^{\otimes 4} + E$, and so long as*

1. $\kappa(a_1, \ldots, a_n) \geqslant \sigma$
2. *the minimum nonzero eigenvalue of $\sum_{i \leqslant n} (a_i^{\otimes 2})(a_i^{\otimes 2})^\top$ is at least σ*
3. $\| \sum_{i \leqslant n} (a_i^{\otimes 3})(a_i^{\otimes 3})^\top \| \leqslant 1/\sigma$, *and*
4. $\|E_{12;34}\| \leqslant \mathrm{poly}(\sigma)/(\log n)^{O(1)}$,

with high probability the algorithm recovers $\tilde{O}(n)$ vectors $b_1, \ldots, b_t$ such that there is a set $S \subseteq \{a_1, \ldots, a_n\}$ with $|S| \geqslant (1 - o(1))n$ such that the signed Hausdorff distance from S to $\{b_1, \ldots, b_t\}$ is $o(1)$, in time $\tilde{O}(n^2 d^3 / \mathrm{poly}(\sigma))$.

Furthermore, hypotheses (2),(3) hold for random unit vectors $a_1, \ldots, a_n$ with $\sigma = 0.1$ so long as $n \leqslant d^2/(\log n)^{O(1)}$, and experiments in Appendix H strongly suggest that (1) does as well.

Proof [Proof of Theorem 25] Let $W = (\sum_{i \leqslant n} a_i^{\otimes 2}(a_i^{\otimes 2})^\top)^{-1/2}$. By Lemma 9, the implicit tensor T' and matrix Π_3 returned by LIFT satisfy

$$\left\| T' - \sum_{i \leqslant n} (W(a_i \otimes a_i))^{\otimes 3} \right\|_F \leqslant O(\varepsilon \sigma_n^{9/2} \sqrt{n})$$

and

$$\left\| \Pi_3 - \Pi_{\mathrm{Span}(W a_i^{\otimes 2} \otimes a_i)} \right\| \leqslant O(\varepsilon \sigma_n^4)\,.$$

So, by Lemma 15, with high probability there is a subset $S \subseteq \{a_1, \ldots, a_n\}$ of size $m \geqslant (1-\delta)n$ such for each $a_i \in S$ there is u_j among the vectors $u_1, \ldots, u_t$ returned by the rounding algorithm with

$$\langle a_i, u_j \rangle^2 \geqslant 1 - O\left(\frac{1}{\delta^{1/8}} \cdot \frac{1}{\beta^{1/8}} \cdot \varepsilon^{1/8} \cdot \|W\| \cdot \sqrt{\sigma_n} \right) = 1 - O\left(\frac{\varepsilon}{\delta \beta} \right)^{1/8}$$

where the equality follows because $\|W\| = \sigma_n^{-1/2}$. Furthermore, each of the vectors $u_1, \ldots, u_t$ is similarly close to some $a_i \in S$. This proves the claimed upper bound on the Hausdorff distance.

The running time follows from putting together Lemma 10 and the running time bounds of Lemma 15. ∎

Appendix F. Tools for analysis and implementation

Lemma *[Restatement of Lemma 8] For a tensor $\mathbf{T} \in (\mathbb{R}^{[d]^2})^{\otimes 3}$, suppose that the matrix $T \in \mathbb{R}^{[d]^3 \times [d]^3}$ given by $T_{(i,i',j),(k,k',j')} = \mathbf{T}_{(i,i'),(j,j'),(k,k')}$ has a rank-n decomposition $T = UV^\mathsf{T}$ with $U, V \in \mathbb{R}^{d^3 \times n}$ and $n \leqslant d^2$. Such a rank decomposition provides an implicit representation of the tensor $\mathbf{T}$. This implicit representation supports:*

Tensor contraction: *For vectors $x, y \in \mathbb{R}^{[d]^2}$, the computation of $(x^\mathsf{T} \otimes y^\mathsf{T} \otimes \mathrm{Id})\mathbf{T}$ or $(x^\mathsf{T} \otimes \mathrm{Id} \otimes y^\mathsf{T})\mathbf{T}$ or $(\mathrm{Id} \otimes x^\mathsf{T} \otimes y^\mathsf{T})\mathbf{T}$ in time $O(nd^3)$ to obtain an output vector in $\mathbb{R}^{d^2}$.*

Spectral truncation: *For $R \in \mathbb{R}^{d^2 \times d^4}$ equal to one of the two matrix reshapings $T_{\{1,2\}\{3\}}$ or $T_{\{2,3\}\{1\}}$ of $\mathbf{T}$, an approximation to the tensor $\mathbf{T}^{\leqslant 1}$, defined as $\mathbf{T}$ after all larger-than-1 singular values in its reshaping R are truncated down to 1. Specifically, letting ρ_k be the kth largest singular value of R for $k \leqslant O(n)$, this returns an implicit representation of a tensor $\mathbf{T}'$ such that $\|\mathbf{T}' - \mathbf{T}^{\leqslant 1}\|_F \leqslant (1+\delta)\rho_k \|\mathbf{T}\|_F$ and the reshaping of $\mathbf{T}'$ corresponding to R has largest singular value no more than $1 + (1+\delta)\rho_k$. The representation of $\mathbf{T}'$ also supports the tensor contraction, spectral truncation, and implicit matrix multiplication operations, with no more than a constant factor increase in runtime. This takes time $\tilde{O}(n^2 d^3 + k(nd^3 + kd^2)\delta^{-1/2})$.*

Implicit matrix multiplication: *For a matrix $R \in \mathbb{R}^{[d]^2 \times [d]^2}$ with rank at most $O(n)$, an implicit representation of the tensor $(R^\mathsf{T} \otimes \mathrm{Id} \otimes \mathrm{Id})\mathbf{T}$ or $(\mathrm{Id} \otimes \mathrm{Id} \otimes R^\mathsf{T})\mathbf{T}$, in time $O(nd^4)$. This output also supports the tensor contraction, spectral truncation, and implicit matrix multiplication operations, with no more than a constant factor increase in runtime. Multiplication into the second mode $(\mathrm{Id} \otimes R^\mathsf{T} \otimes \mathrm{Id})\mathbf{T}$ may also be implicitly represented, but without support for the spectral truncation operation.*

Proof

Tensor contraction We start with multiplication of two vectors $x, y \in \mathbb{R}^{d^2}$ into two of the modes of $\mathbf{T}$. Without loss of generality (by interchange of U and V), there are two cases: we want either to compute the vector flattening of $(x \otimes \mathrm{Id}_d)^\mathsf{T} UV^\mathsf{T}(y \otimes \mathrm{Id}_d)$, or, expressing $x = \sum_{i=0}^d r_i \otimes s_i$, we want $\sum_i (\mathrm{Id}_d \otimes \mathrm{Id}_d \otimes r_i)^\mathsf{T} UV^\mathsf{T}(y \otimes s_i)$. For both these cases, we first compute $V^\mathsf{T}(y \otimes \mathrm{Id}_d)$.

We compute $V^\mathsf{T}(y \otimes \mathrm{Id}_d)$ as $[V^\mathsf{T}(y \otimes \mathrm{Id}_d)]_{\cdot;i} = V^\mathsf{T}{}_{\cdot;(\cdot,\cdot,i)} y$. This is a concatenation of d different matrix-vector multiplications using $n \times d^2$ matrices, and so it takes $O(nd^3)$ time.

Then to find $(x^\mathsf{T} \otimes \mathrm{Id}_d)UV^\mathsf{T}(y \otimes \mathrm{Id}_d)$, we simply repeat the above procedure to find $(x^\mathsf{T} \otimes \mathrm{Id}_d)U$ and then multiply the $d \times n$ and $n \times d$ matrices together in $O(nd^2)$ time.

To find $\sum_i (\mathrm{Id}_d \otimes \mathrm{Id}_d \otimes r_i)^\mathsf{T} UV^\mathsf{T}(y \otimes s_i)$ after finding the rank decomposition $x = \sum_{i=0}^d r_i \otimes s_i$ which takes $O(d^3)$ time by SVD, we multiply each s_i into our computed value of $V^\mathsf{T}(y \otimes \mathrm{Id}_d)$ to obtain d different n-dimensional vectors $t_i = V^\mathsf{T}(y \otimes s_i)$. Since there are d of these vectors and each is a matrix-vector multiplication with an $n \times d$ matrix, this takes $O(nd^2)$ time. Then $\sum_i (\mathrm{Id}_d \otimes \mathrm{Id}_d \otimes r_i)^\mathsf{T} U t_i$ can be reshaped as a multiplication of a $d^2 \times nd$ reshaping of U with the vector $\sum_i t_i \otimes r_i$. It takes $O(nd^3)$ time to perform the matrix-vector multiplication, and $O(nd^2)$ time to sum up $\sum_i t_i \otimes r_i$.

Spectral truncation Next, we truncate the larger-than-1 singular values of the $(\{1\}, \{2,3\})$ and $(\{3\}, \{1,2\})$ matrix reshapings $R \in \mathbb{R}^{d^2 \times d^4}$ of $\mathbf{T}$. Without loss of generality, suppose we

are in the $(\{3\},\{1,2\})$ case. In this case, we would like to find the right-singular vectors and singular values of the operator that takes $y \in \mathbb{R}^{d^2}$ to the vector flattening of the $d \times d^3$ matrix $UV^\mathsf{T}(y \otimes \mathrm{Id}_d)$. Letting Z be the $nd \times d^2$ reshaping of V, this is the same as $(U \otimes \mathrm{Id}_d)Zy$, which shares its right-singular vectors with $M := Z^\mathsf{T}(U^\mathsf{T}U \otimes \mathrm{Id}_d)Z$.

We claim that matrix-vector multiplication by M can be implemented in $O(nd^3)$ time, with $O(n^2d^3)$ preprocessing time for computing the product $U^\mathsf{T}U$. The matrix-vector multiplications by Z and Z^T take time $O(nd^3)$, and then multiplying Zy by $U^\mathsf{T}U \otimes \mathrm{Id}_d$ is reshaping-equivalent to multiplying $U^\mathsf{T}U$ into the $n \times d$ matrix reshaping of Zy, which takes $O(n^2d)$ time with the precomputed $n \times n$ matrix $U^\mathsf{T}U$. Therefore, LazySVD (Allen-Zhu and Li, 2016, Corollary 4.4) takes time $\tilde{O}(n^2d^3\delta^{-1/2})$ to yield a rank-k eigendecomposition $P\Lambda P^\mathsf{T}$ such that $\|M^{1/2} - P\Lambda^{1/2}P^\mathsf{T}\| \leqslant (1+\delta)\rho_k$.

To obtain the output $\mathbf{T}'$ of this procedure, let $(P\Lambda^{1/2}P^\mathsf{T} - \mathrm{Id})^{>0}$ be $P\Lambda^{1/2}P^\mathsf{T} - \mathrm{Id}$ with all of its nonpositive eigenvalues removed: this may be implemented by removing nonpositive entries from $\Lambda^{1/2} - \mathrm{Id}$. Then implicitly multiply $(\mathrm{Id} - (P\Lambda^{1/2}P^\mathsf{T} - \mathrm{Id})^{>0})$ into the third mode of $\mathbf{T}$ (although this matrix has rank larger than n, we may implement it by implicitly subtracting $(\mathrm{Id}\otimes\mathrm{Id}\otimes(P\Lambda^{1/2}P^\mathsf{T} - \mathrm{Id})^{>0})\mathbf{T}$ from $\mathbf{T}$). We are trying to approximate multiplying $(\mathrm{Id} - (M^{1/2} - \mathrm{Id})^{>0})$ into the third mode of $\mathbf{T}$, so let $\Delta = (M^{1/2} - \mathrm{Id})^{>0} - (P\Lambda^{1/2}P^\mathsf{T} - \mathrm{Id})^{>0}$ be the difference. Then $\|\Delta\| \leqslant \|M^{1/2} - P\Lambda^{1/2}P^\mathsf{T}\| \leqslant (1+\delta)\rho_k$, so that we suffer an additive error of at most $(1+\delta)\rho_k$ in spectral norm. And the final error in the low-rank representation is $(\Delta \otimes \mathrm{Id})UV^\mathsf{T}$. Since $\|\Delta \otimes \mathrm{Id}\| \leqslant (1+\delta)\rho_k$ and UV^T has Frobenius norm $\|\mathbf{T}\|_F$, we find a final error of $(1+\delta)\rho_k\|\mathbf{T}\|_F$ in Frobenius norm.

Implicit matrix multiplication Finally, to implicitly multiply a $d^2 \times d^2$ rank-n matrix R into a mode of $\mathbf{T}$, simply store the singular value decomposition $R = P\Sigma Q^\mathsf{T}$. Whenever a vector needs to be multiplied into that mode in the future, multiply that vector by R before carrying out the implicit tensor operation as previously specified, and if a vector needs to be output from that mode, multiply it by R^T before outputting. This incurs a time cost of $O(nd^2)$ per operation.

A special case arises in the spectral truncation operation, where we do not allow implicit multiplication to have been done in the second mode. Suppose then without loss of generality that R was multiplied into the first mode of $\mathbf{T}$ and we truncate the $(\{3\},\{1,2\})$ matrix reshaping. Then we will have to compute $U^\mathsf{T}(RR^\mathsf{T}\otimes\mathrm{Id}_d)U$ instead of $U^\mathsf{T}U$ in the preprocessing step. This can be done by multiplying $R^\mathsf{T} = Q\Sigma P^\mathsf{T}$ with the $d^2 \times nd$ reshaping of U, which takes $O(n^2d^3)$ time per future spectral truncation operation. ■

Appendix G. Notes on Table 1

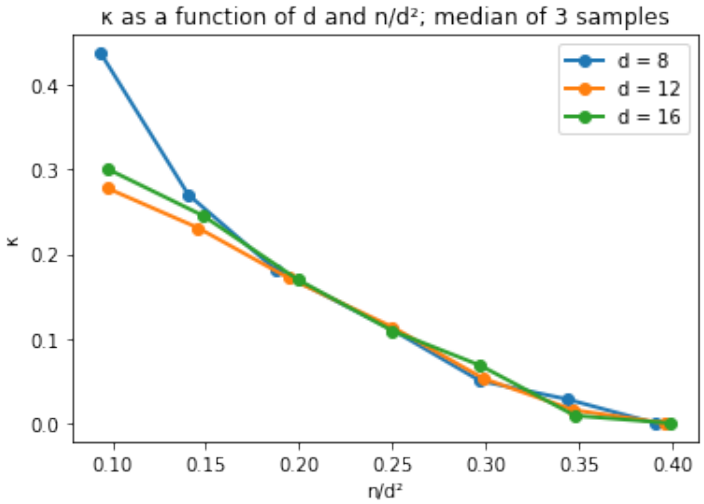
We record a few notes on parameter regimes used to compare various algorithms for tensor decomposition in Table 1.

- Robust algorithms with algebraic assumptions often require $\|E\| \leqslant \sigma(a_1,\dots,a_n)$, where $\sigma(a_1,\dots,a_n)$ is some measure of well-conditioned-ness of $a_1,\dots,a_n$, the details of which may vary from algorithm to algorithm. In this table we report results for the setting that $\sigma(a_1,\dots,a_n) \geqslant \Omega(1)$; such values of σ (for all the notions of well-conditioned-ness represented) are achieved by random $a_1,\dots,a_n$.

- The algorithm of Anandkumar et al. (2017) is phrased for 3-tensors rathern than 4-tensors; this is the origin of the rank bound $n \leqslant d^{1.5}$ rather than $d \leqslant n^2$ achieved by algorithms for 4-tensors. In general for k-tensors one expects efficient algorithms to tolerate overcompleteness $n \leqslant d^{k/2}$ (despite tensor rank factorizations remaining unique for much larger n), so the overcompleteness guarantee of Anandkumar et al. (2017) is comparable to the other algorithms.
- We have estimated the running time of the SoS algorithm of Ma et al. (2016) by assuming that the semidefinite programs involved are solved using standard black-box techniques (e.g the ellipsiod method).

Appendix H. Simulations for condition number of random tensors

In this section we report on computer simulations which strongly suggest that if $a_1, \ldots, a_n$ are i.i.d. random unit vectors with $n \ll d^2$ then with high probability $\kappa(a_1, \ldots, a_n) \geqslant \Omega(1)$. We computed κ for several values of n, d on with $a_1, \ldots, a_n$ taken to be i.i.d. uniformly random unit vectors. Our results are consistent with the hypothesis that $\kappa(a_1, \ldots, a_n) \geqslant c - \tilde{O}(n/d^2)$ for some absolute constant $c \approx 1/2$:



We expect the values of d, n employed here – $d \approx 10, n \approx 100$, so that κ is the condition number of a certain random matrix of dimensions about $10^3 \times 10^3$ – to be predictive of the asymptotic behavior of $\kappa(a_1, \ldots, a_n)$, because the spectra of random matrices display strong concentration even in relatively small dimensions.

We also note that the hypothesis that $\kappa > c - \tilde{O}(n/d^2)$ is well supported by the fact that relatively standard techniques from random matrix theory yield the same bound for a closely related random matrix to $K(a_1, \ldots, a_n)$ from Definition 3. In particular, the following may be proved by a long but standard calculation, using Matrix Bernstein and decoupling inequalities:

Lemma 27 (Condition number of basic swap matrix) *Let $a_1, \ldots, a_n$ be independent random d-dimensional unit vectors. Let $B_i \in \mathbb{R}^{(d-1)\times d}$ be a random basis for the orthogonal complement of a_i in $\mathbb{R}^d$. Let $P \in \mathbb{R}^{d^3 \times d^3}$ be the permutation matrix which swaps second and*

third modes of $(\mathbb{R}^d)^{\otimes 3}$. *Let*

$$A = \mathop{\mathbb{E}}_{a}(a \otimes a \otimes \mathrm{Id})(a \otimes a \otimes \mathrm{Id})^\top .$$

Let $R \in \mathbb{R}^{d^3 \times n(d-1)}$ *have* n *blocks of dimensions* $d^3 \times (d-1)$, *where the* i*-th block is*

$$R_i = A^{-1/2}(a_i \otimes a_i \otimes B_i) - PA^{-1/2}(a_i \otimes a_i \otimes B_i)$$

where we abuse notation and denote the PSD square root of the pseudoinverse of A *by* $A^{-1/2}$. *Then there is a function* $d'(d) = \Theta(d^2)$ *such that* $\mathbb{E}\,\|R^\top R - d'(d) \cdot \mathrm{Id}\| \leqslant O(\log d)^2 \cdot \max(d\sqrt{n}, n, d^{3/2})$. *In particular, if* $d \ll n \ll d^2$,

$$\mathbb{E}\,\|\tfrac{1}{d'(d)} R^\top R - \mathrm{Id}\| \leqslant O(n(\log d)^2/d^2)\,.$$

The matrix R from this lemma differs from K only in the use of $A^{-1/2}$ in place of $(H_1^\top H_1)^{-1/2}, (H_2^\top H_2)^{-1/2}$. While we expect $A^{-1/2}$ (a non-random matrix) to be close to both $(H_1^\top H_1)^{-1/2}, (H_2^\top H_2)^{-1/2}$ (at least in subspaces close to $Im(H_1)$ and $Im(H_2)$, respectively) establishing this is a challenging task in random matrix theory – in particular, both inverses of random matrices and spectra of random matrices with dependent entries are notoriously difficult to analyze. We leave this challenge to future work.

In this section we report on computer simulations which strongly suggest that if the components $a_1, \ldots, a_n$ are $n \ll d^2$ random unit vectors from a variety of ensembles, then with high probability $\kappa(a_1, \ldots, a_n) \geqslant \Omega(1)$. The ensembes include:

1. *Spherical measure:* $a_1, \ldots, a_n \in \mathbb{R}^d$ are i.i.d. random unit vectors (see Fig. 1).

2. *Sparse:* $a_1, \ldots, a_n \in \mathbb{R}^d$ are sampled i.i.d. by choosing $\frac{1}{4}d$ coordinates in $[d]$ uniformly at random, sampling each of those coordinates from $\mathcal{N}(0,1)$, and setting the rest to 0 (see Fig. 2).

3. *Hypercube:* $a_1, \ldots, a_n \in \mathbb{R}^d$ are i.i.d. samples from $\{0,1\}^d$ (see Fig. 3).

4. *Spiked covariance:* $a_1, \ldots, a_n \in \mathbb{R}^d$ are sampled from $\mathcal{N}(0, \mathrm{Id} + \lambda \cdot uu^\top)$ for a random unit vector u and $\lambda > 0$. We note that in this case, though the covariance matrix of $a_1, \ldots, a_n$ has condition number $O(\frac{1}{\lambda})$, our experimental results support the hypothesis that $\kappa(a_1, \ldots, a_n) = \Omega(1)$ for λ as large as $\lambda = \frac{1}{2}d$ (see Fig. 4).

These ensembles are designed to capture a number of characteristics of real data which we would like the condition number to be robust to: sparsity, discrete values, and correlations (of relatively extreme magnitude).

In each of these cases, we computed κ for several values of n, d on with $a_1, \ldots, a_n$ taken to be i.i.d. uniformly random unit vectors. Our results are consistent with the hypothesis that (with high probability) $\kappa(a_1, \ldots, a_n) \geqslant c - \tilde{O}(n/d^2)$ for some absolute constant $c \approx \frac{1}{2}$.

We expect the values of d, n employed here – $d \approx 10, n \approx 100$, so that κ is the condition number of a certain random matrix of dimensions about $10^3 \times 10^3$ – to be predictive of the asymptotic behavior of $\kappa(a_1, \ldots, a_n)$, because the spectra of random matrices display strong concentration even in relatively small dimensions.

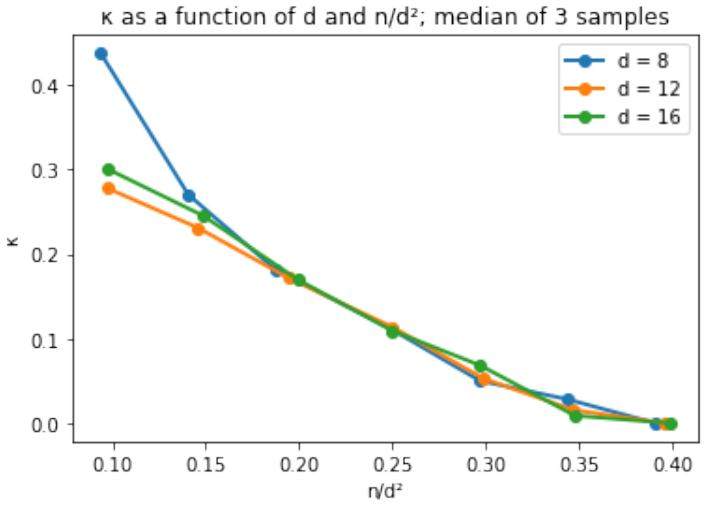


Figure 1: Condition number as a function of dimension d and lifted overcompleteness n/d^2 for vectors sampled from the spherical measure.

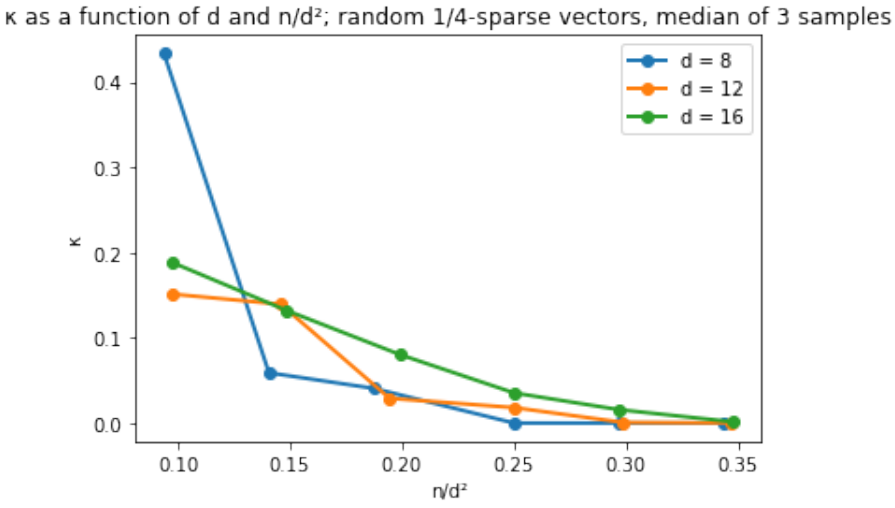


Figure 2: Condition number as a function of dimension d and lifted overcompleteness n/d^2 for random sparse vectors.

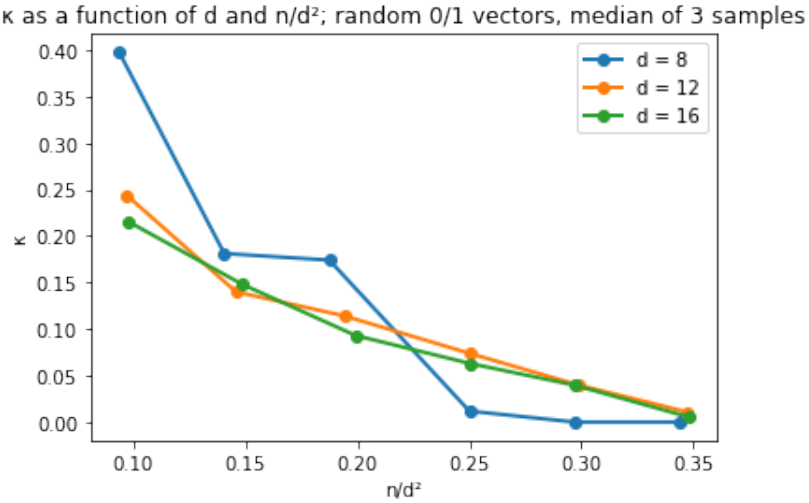


Figure 3: Condition number as a function of dimension d and lifted overcompleteness n/d^2 for vectors sampled from the Boolean hypercube.

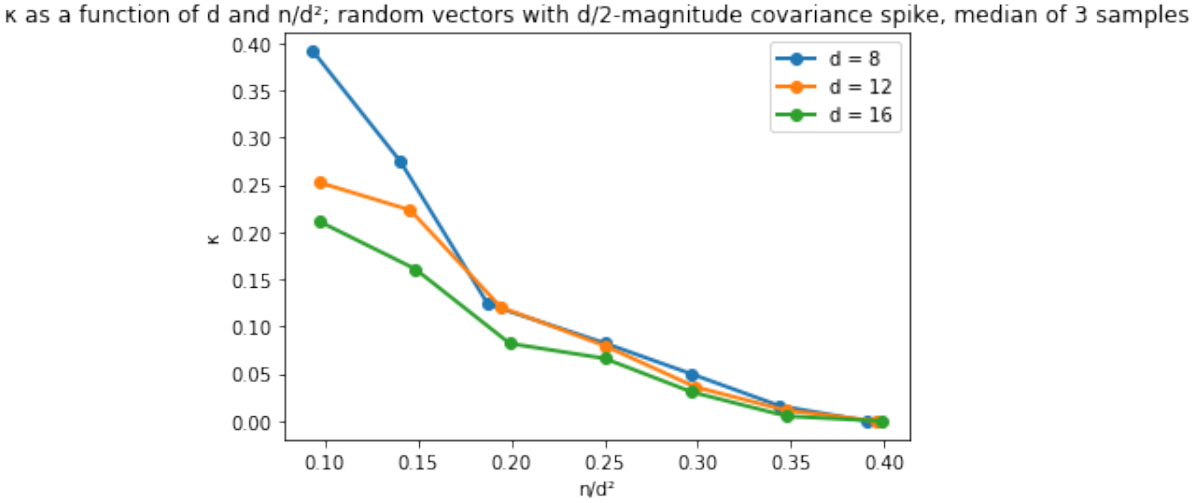


Figure 4: Condition number as a function of dimension d and lifted overcompleteness n/d^2 for vectors sampled from $\mathcal{N}(0, \mathrm{Id} + \frac{1}{2}d \cdot uu^\top)$ for a random unit vector u.

We also note that the hypothesis that $\kappa > c - \tilde{O}(n/d^2)$ is well supported by the fact that relatively standard techniques from random matrix theory yield the same bound for a closely related random matrix to $K(a_1, \ldots, a_n)$ from Definition 3. In particular, the following may be proved by a long but standard calculation, using Matrix Bernstein and decoupling inequalities:

Lemma 28 (Condition number of basic swap matrix) *Let $a_1, \ldots, a_n$ be independent random d-dimensional unit vectors. Let $B_i \in \mathbb{R}^{(d-1)\times d}$ be a random basis for the orthogonal complement of a_i in $\mathbb{R}^d$. Let $P \in \mathbb{R}^{d^3 \times d^3}$ be the permutation matrix which swaps second and third modes of $(\mathbb{R}^d)^{\otimes 3}$. Let*

$$A = \mathop{\mathbb{E}}_{a}(a \otimes a \otimes \mathrm{Id})(a \otimes a \otimes \mathrm{Id})^\top .$$

Let $R \in \mathbb{R}^{d^3 \times n(d-1)}$ have n blocks of dimensions $d^3 \times (d-1)$, where the i-th block is

$$R_i = A^{-1/2}(a_i \otimes a_i \otimes B_i) - PA^{-1/2}(a_i \otimes a_i \otimes B_i)$$

where we abuse notation and denote the PSD square root of the pseudoinverse of A by $A^{-1/2}$. Then there is a function $d'(d) = \Theta(d^2)$ such that $\mathbb{E}\,\|R^\top R - d'(d) \cdot \mathrm{Id}\| \leqslant O(\log d)^2 \cdot \max(d\sqrt{n}, n, d^{3/2})$. In particular, if $d \ll n \ll d^2$,

$$\mathbb{E}\,\|\tfrac{1}{d'(d)} R^\top R - \mathrm{Id}\| \leqslant O(n(\log d)^2/d^2)\,.$$

The matrix R from this lemma differs from K only in the use of $A^{-1/2}$ in place of $(H_1^\top H_1)^{-1/2}, (H_2^\top H_2)^{-1/2}$. While we expect $A^{-1/2}$ (a non-random matrix) to be close to both $(H_1^\top H_1)^{-1/2}, (H_2^\top H_2)^{-1/2}$ (at least in subspaces close to $Im(H_1)$ and $Im(H_2)$, respectively) establishing this is a challenging task in random matrix theory – in particular, both inverses of random matrices and spectra of random matrices with dependent entries are notoriously difficult to analyze. We leave this challenge to future work.