Arnold Mathematical Journal
https://doi.org/10.1007/s40598-023-00235-4

RESEARCH EXPOSITION

Approximate Real Symmetric Tensor Rank

Alperen A. Ergür[1] · **Jesus Rebollo Bueno**[3] · **Petros Valettas**[2]

Received: 9 January 2023 / Revised: 16 June 2023 / Accepted: 30 July 2023

Abstract
We investigate the effect of an ε-room of perturbation tolerance on symmetric tensor decomposition. To be more precise, suppose a real symmetric d-tensor f, a norm $\|\cdot\|$ on the space of symmetric d-tensors, and $\varepsilon > 0$ are given. What is the smallest symmetric tensor rank in the ε-neighborhood of f? In other words, what is the symmetric tensor rank of f after a clever ε-perturbation? We prove two theorems and develop three corresponding algorithms that give constructive upper bounds for this question. With expository goals in mind, we present probabilistic and convex geometric ideas behind our results, reproduce some known results, and point out open problems.

Keywords Symmetric Tensor Rank · Energy Increment Method · Maurey's Empricial Method · Approximate Sparsification · Polynomial Optimization

1 Introduction

Tensors encode fundamental questions in mathematics and complexity theory, such as finding lower bounds on the matrix multiplication exponent, and have been extensively studied from this perspective [13, 18, 30, 40]. In computational mathematics, tensor decomposition-based methods gained prominence in the 1990s [11], and became a common tool for learning latent (hidden) variable models after [3]. Tensor decomposition-based methods are now broadly used in application domains ranging from phylogenetics to community detection in networks. We suggest [32] as an excellent survey for clarifying basic concepts and for many examples of tensor computations. Tensor decomposition-based methods are also used for a large range of

✉ Alperen A. Ergür
alperen.ergur@utsa.edu

1 Mathematics and Computer Science, The University of Texas at San Antonio, San Antonio, Texas, USA

2 Mathematics & Electrical Enginering and Computer Science, University of Missouri, Columbia, Missouri, USA

3 Mathematics, The University of Texas at San Antonio, San Antonio, Texas, USA

Published online: 22 August 2023

tasks in machine learning such as training shallow and deep neural nets [21, 37], ubiquitous applications of the moments method [23, 34], computer vision applications [38], and much more: see [1, 43] for surveys of results available as of 2008 and 2017, respectively.

As opposed to using arbitrary tensors without any structure, the usage of symmetric tensors appears as a common thread in wide-ranging applications of tensor-decomposition-based methods. This is the main focus of our paper: the real symmetric decomposition of real symmetric tensors. Let us be more precise:

Definition 1.1 (*Symmetric Tensor Rank*) Let f be an n-variate real symmetric d-tensor and $S^{n-1} := \{u \in \mathbb{R}^n : \|u\|_2 = 1\}$. The smallest $m \in \mathbb{N}$ for which there exist $c_1, \ldots, c_m \in \mathbb{R}$ and $v_1, \ldots, v_m \in S^{n-1}$ so that

$$f = \sum_{i=1}^{m} c_i \underbrace{v_i \otimes v_i \otimes \cdots \otimes v_i}_{\text{d times}}$$

is called the symmetric tensor rank and we denote this rank by srank(f).

Symmetric tensor rank is sometimes called CAND in signal processing, and can be named real-Waring rank after identification with homogeneous polynomials [8]. Analogous definitions for asymmetric tensors are called CANDECOMP, PARAFAC, or CP as a short for these two names [12]. We emphasize that in our definition, real symmetric tensors are decomposed into rank-1 real tensors, whereas in basic references such as [8, 12], the main focus is on the decomposition of real symmetric tensors into complex rank-one tensors. One reason for using complex decomposition is to be able to employ tools from algebraic geometry which work better on algebraically closed fields, see, e.g., the delightful paper [35]. Our aim in this paper is to use convex geometric tools to take advantage of the beauties of real geometry: being an ordered field makes the geometry over the reals (and rank notions) intrinsically different than the complex ones.

It is known that the tensor rank on reals is not stable under perturbation: It is typical/expected for designers of tensor decomposition algorithms to exercise caution not to let noise obscure a low-rank input tensor as a high rank one. In a similar spirit to the smoothed analysis [44], we suggest viewing the inherent existence of error in real number computations as an advantage rather than an obstacle. More formally, we propose to relax the srank notion with an ε-room of tolerance.

Definition 1.2 (*Approximate Symmetric Tensor Rank*) Let $\|\cdot\|$ denote a norm on the space of n-variate real symmetric d-tensors. Given a symmetric d-tensor f, we define the ε-approximate rank of f with respect to $\|\cdot\|$ as follows:

$$\text{srank}_{\|\cdot\|,\varepsilon}(f) := \min\{\text{srank}(h) : \|h - f\| \leq \varepsilon\}.$$

Our main results, Theorems 3.1, 3.3, 4.1, 4.7, and Corollary 5.2 show that $\text{srank}_{\|\cdot\|,\varepsilon}$ behave significantly different than its algebraic counterpart srank(f).

From an operational perspective, one might prefer to use an "efficient" family of norms instead of using an arbitrary norm as in Definition 1.2. Although some of our

theorems hold for arbitrary norms, our main focus is on perturbation with respect to L_p-norms. This is due to the existence of efficient quadrature rules to compute L_p-norms of symmetric tensors [14, 25].

The rest of the paper is organized as follows: in Sect. 2, we introduce the vocabulary and basic concepts; in Sects. 3, 4, and 5, we present three constructive estimates, based on three different ideas, for approximate rank. Section 3.4 presents implementation details of the energy increment algorithm (Algorithm 1). Finally, in Sect. 6, we consider an application to optimization.

2 Mathematical Concepts

In this section, for the sake of clarity, we explicitly introduce all the mathematical notions that are used in this paper.

2.1 Basic Terminology and Monomial Index

Let $T^d(\mathbb{R}^n) := \mathbb{R}^n \otimes \mathbb{R}^n \otimes \cdots \otimes \mathbb{R}^n$ be the set of all d-tensors. Then, we consider the action of the symmetric group on the set $\{1, 2, 3, \ldots, d\}$, $\mathcal{S}_d$, on $T^d(\mathbb{R}^n)$ as follows: for $\sigma \in \mathcal{S}_d$ and $u^{(1)} \otimes u^{(2)} \otimes \cdots \otimes u^{(d)} \in T^d(\mathbb{R}^n)$, we have

$$\sigma(u^{(1)} \otimes u^{(2)} \otimes \cdots \otimes u^{(d)}) = u^{(\sigma(1))} \otimes u^{(\sigma(2))} \otimes \cdots \otimes u^{(\sigma(d))}.$$

The action of $\mathcal{S}_d$ extends linearly to the entire space $T^d(\mathbb{R}^n)$. A tensor $A \in T^d(\mathbb{R}^n)$ is called a symmetric tensor if $\sigma(A) = A$ for all $\sigma \in \mathcal{S}_d$. We denote the vector space of symmetric d-tensors on $\mathbb{R}^n$ by $P_{n,d}$. Equivalently, one can think about this space as the span of self-outer products of vectors $v \in \mathbb{R}^n$, that is,

$$P_{n,d} := \operatorname{span}\{\underbrace{v \otimes v \otimes v \otimes \cdots \otimes v}_{\text{d times}} \mid v \in \mathbb{R}^n\}.$$

Now, we pose the following question: Given a rank-1 symmetric tensor $v \otimes v \otimes v \in P_{n,3}$, what is the difference between $[v \otimes v \otimes v]_{1,2,1}$ and $[v \otimes v \otimes v]_{1,1,2}$? Due to symmetry, these two entries are equal. Likewise, for any element A in $P_{n,d}$, two entries $a_{i_1,i_2,\ldots,i_d}$ and $a_{j_1,j_2,\ldots,j_d}$ are identical whenever $\{i_1, i_2, \ldots, i_d\}$ and $\{j_1, j_2, \ldots, j_d\}$ are equal as supersets. This allows the use of monomial index: A superset $\{i_1, i_2, \ldots, i_d\}$ is identified with a monomial $x^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$, where $\alpha = (\alpha_1, \ldots, \alpha_n)$, α_j is the number of j's in $\{i_1, i_2, \ldots, i_d\}$, and $d = \alpha_1 + \alpha_2 + \cdots + \alpha_n$ is the degree of the monomial. In the monomial index, instead of listing $\binom{d}{\alpha}$ equal entries for all the supersets identified with x^α, we only list the sum of these entries once.

2.2 Euclidean and Functional Norms

For $f \in P_{n,d}$ and $x \in S^{n-1}$, when we write $f(x)$ we mean f applied to $[x, x, \ldots, x]$ as a symmetric multilinear form. For $r \in [2, \infty)$, the L_r functional norms on $P_{n,d}$ are

defined as

$$\|f\|_r := \left(\int_{S^{n-1}} |f(x)|^r \, \sigma(x) \right)^{1/r}, \quad f \in P_{n,d},$$

where σ is the uniform probability measure on the sphere S^{n-1}. The L_∞-norm on $P_{n,d}$ is defined by

$$\|f\|_\infty := \max_{v \in S^{n-1}} |f(v)|.$$

For all L_r-norms, we use B_r to denote the unit ball of the space $(P_{n,d}, \|\cdot\|_r)$. That is,

$$B_r := \{p \in P_{n,d} : \|p\|_r \leq 1\}.$$

We recall an important fact about L_r-norms of symmetric tensors established in [6].

Lemma 2.1 (Barvinok) *Let $g \in P_{n,d}$, then we have*

$$\|g\|_{2k} \leq \|g\|_\infty \leq \binom{kd+n-1}{kd}^{\frac{1}{2k}} \|g\|_{2k}.$$

In particular, for $k \geq n \log(ed)$, we have

$$\|g\|_{2k} \leq \|g\|_\infty \leq c \, \|g\|_{2k}$$

for some constant c.

Definition 2.2 (*Hilbert–Schmidt in the monomial index*) Let $p, q \in P_{n,d}$ indexed using the monomial notation, that is $p = [b_\alpha]_\alpha$ and $q = [c_\alpha]_\alpha$ where $\alpha \in \mathbb{Z}^n_{\geq 0}$ satisfies $|\alpha| := \alpha_1 + \cdots + \alpha_n = d$. Then, the Hilbert–Schmidt inner product of p and q is given by

$$\langle p, q \rangle_{\mathrm{HS}} := \sum_{|\alpha|=d} \frac{b_\alpha c_\alpha}{\binom{d}{\alpha}}.$$

Note that in algebraic geometry literature this norm is named as Bombieri–Weyl norm. Now, for simplicity, we define $q_v := \underbrace{v \otimes v \otimes v \otimes \cdots \otimes v}_{\text{d times}}$ for a $v \in S^{n-1}$, then we have the following identity:

$$\max_{v \in S^{n-1}} |\langle g, q_v \rangle_{\mathrm{HS}}| = \max_{v \in S^{n-1}} |g(v)| = \|g\|_\infty. \tag{1}$$

2.3 Nuclear Norm and Veronese Body

We start this section by recalling the connection between norms and the geometry of the corresponding unit balls. Every centrally symmetric convex body $K \subset \mathbb{R}^n$ induces a unique norm, that is, for $x \in \mathbb{R}^n$

$$\|x\|_K := \min\{\lambda > 0 : x \in \lambda K\}. \tag{2}$$

For every $v \in S^{n-1}$, we have two associated symmetric tensors: $p_v = v \otimes v \otimes v \otimes \cdots \otimes v$ and $-p_v$. Using the terminology established in [41], we define the Veronese body, $V_{n,d}$, as follows:

$$V_{n,d} := \text{conv}\{\pm p_v : v \in S^{n-1}\}. \tag{3}$$

The norm introduced by the convex body $V_{n,d}$, $\|.\|_{V_{n,d}}$, is called the nuclear norm and it is usually denoted in the literature by $\|.\|_*$. It follows from (2) that for every $q \in P_{n,d}$, we have

$$\|q\|_* = \min\left\{\sum_{i=1}^{m} |\lambda_i| : q = \sum_{i=1}^{m} \lambda_i p_{v_i},\ v_i \in S^{n-1}\right\};$$

for background material on these facts see Section 3 of the survey [24]. Considering (1), one may notice that for every $q \in P_{n,d}$

$$\|q\|_\infty = \max_{f \in V_{n,d}} \langle q, f \rangle_{\text{HS}},$$

meaning that the norm introduced by $V_{n,d}$ on $P_{n,d}$ is dual to the L_∞-norm. Then, by the duality of the norms $\|.\|_\infty$ and $\|.\|_*$, for every $g \in P_{n,d}$, we have

$$\|g\|_* = \max_{q \in B_\infty} \langle g, q \rangle_{\text{HS}}. \tag{4}$$

Formulation (4) suggests a semi-definite programming approach for computing $\|.\|_*$ by approximating B_∞ with the sum of squares hierarchy. Note that this approach would yield *lower bounds* for the nuclear norm that improve as the degree of sum of squares hierarchy is increased. Luckily for us, this increasing lower bounds via sum of squares idea is already made rigorous and can be implemented using any semi-definite programming software [36].

2.4 Type-2 Constant of a Norm

The type-2 constant allows us to create a sparse randomly constructed approximation to a given vector with controlled error; the definition of the type-2 constant carries an essential idea to control the trade-off between error and sparsity. We will give more

details and intuition on this matter in Sect. 4. To define the type-2 constant, we first need to recall that a Rademacher random variable ξ is defined by

$$\mathbb{P}(\xi = -1) = \mathbb{P}(\xi = 1) = 1/2.$$

Definition 2.3 (*type-2 constant*) Let $\|\cdot\|$ be a norm on $\mathbb{R}^n$. The type-2 constant of $X = (\mathbb{R}^n, \|\cdot\|)$, denoted by $T_2(X)$, is the smallest possible $T > 0$ such that for any $m \in \mathbb{N}$ and any collection of vectors $x_1, \ldots, x_m \in \mathbb{R}^n$ one has

$$\mathbb{E}_{\xi_1,\ldots,\xi_m} \left\| \sum_{i=1}^{m} \xi_i x_i \right\|^2 \leq T^2 \sum_{i=1}^{m} \|x_i\|^2, \tag{5}$$

where ξ_i, $i = 1, 2, \ldots, m$ are independent Rademacher random variables.

Lemma 2.4 (Properties of Type-2 Constant [31, 46])

(1) *Let A be an invertible linear map.*
If $\|x\|_D := \|A^{-1}x\|_K$ for all $x \in X$, then $T_2(X, \|.\|_D) = T_2(X, \|.\|_K)$
(2) *Every Euclidean norm has type-2 constant 1.*
(3) *If Y is a subspace of X, then $T_2(Y) \leq T_2(X)$.*
(4) *If X is n-dimensional, then $T_2(X) \leq \sqrt{n}$, and ℓ_1-norm has type-2 constant $\sqrt{n}$.*
(5) *Let $2 \leq p < \infty$. Then, $T_2(\ell_p^n) \lesssim \sqrt{\min\{p, \log n\}}$, where $\ell_p^n = (\mathbb{R}^n, \|\cdot\|_p)$.*

3 Approximate Rank Estimate via Energy Increment

Energy increment is a general strategy in additive combinatorics to set up a greedy approximation to an a priori unknown object, see [45]. Our theorems and algorithms in this section are inspired by the energy increment method as we explain below. We begin by presenting an approximate rank estimate for L_r-norms.

Theorem 3.1 *For $r \in [2, \infty]$, $\|.\|_r$ denotes the L_r-norm on $P_{n,d}$. Then, for any $f \in P_{n,d}$ and $\varepsilon > 0$, we have*

$$\operatorname{srank}_{\|\cdot\|_r,\varepsilon}(f) \leq \frac{\|f\|_{\mathrm{HS}}^2}{\varepsilon^2},$$

where $\|\cdot\|_{\mathrm{HS}}$ denotes the Hilbert–Schmidt norm.

One may wonder why this result is interesting for all L_r-norms when it takes the strongest form for $r = \infty$. The reason is, of course, the computational complexity. Symmetric tensors that are close to each other in terms of L_∞-distance behave almost identical as homogeneous functions on S^{n-1}, but it is NP-Hard to compute L_∞-distance for $d \geq 4$. For $r > n \log(ed)$ the norms L_r and L_∞ on $P_{n,d}$ are equivalent, see Lemma 2.1. Therefore, we only hope to be able to compute approximate decomposition for L_r where r is not proportional to n. Algorithm 1 and Theorem 3.3 below delineate

the trade-off between the tightness of the estimate depending on r and the cost of computation.

Now, we present our energy increment algorithm. In Algorithm 1, Π_W denotes the orthogonal projection on the subspace W with respect to the Hilbert–Schmidt norm, and $q_v := \underbrace{v \otimes v \otimes \cdots \otimes v}_{\text{d times}}$.

Algorithm 1 Approximate Rank via Energy Increment

1: **Input** $f \in P_{n,d}$, $\|.\|_r$ for $2 \le r \le \infty$, and $\varepsilon > 0$.
2: Initialize $\tilde{f} = 0$, $e = \infty$, $W = \{0\}$.
3: **repeat**
4: Find a $v \in S^{n-1}$ such that $\frac{1}{2}\left\|f - \tilde{f}\right\|_r \le \left|(f - \tilde{f})(v)\right|$.
5: $W = \operatorname{span}(W \cup \{q_v\})$
$\tilde{f} = \Pi_W(f)$, $e = \left\|f - \tilde{f}\right\|_r$
6: **until** $e < \varepsilon$
7: **Output** $\tilde{f}$
8: **Post-condition** $\|f - \tilde{f}\|_r < \varepsilon$, and $\operatorname{srank}(\tilde{f}) \le \varepsilon^{-2}\|f\|_{\mathrm{HS}}^2$

Details on the implementation of steps in Algorithm 1 are explained in Sect. 3.4 alongside some experimental results. Our next theorem gives a sampling approach for the search step (4).

Theorem 3.2 *Let $n, d \ge 1$ and $2 \le r \le n\log(ed)$. Let $p \in P_{n,d}$ and suppose $v_1, v_2, \ldots, v_N$ are vectors that are sampled independently from the uniform probability measure on the sphere S^{n-1}. Then, we have*

$$\mathbb{P}\left(\max_{i \le N} |p(v_i)| \ge \frac{1}{2}\|p\|_r\right) \ge 1 - \exp\left(-N/[\alpha(n,d,r)]^{2r}\right),$$

where $\alpha(n,d,r) := \min\{(c_1 r)^{d/2}, \binom{rd+n-1}{rd}^{\frac{1}{2r}}\}$ for a constant c_1. In particular, if $N \ge t[\alpha(n,d,r)]^{2r}$, we have

$$\mathbb{P}\left(\max_{i \le N} |p(v_i)| \ge \frac{1}{2}\|p\|_r\right) \ge 1 - e^{-t}.$$

The proof of Theorem 3.2 is included in Sect. 3.2. As a consequence of Theorem 3.2 and the bounds obtained in the proof of Theorem 3.1, we have the following result on Algorithm 1.

Theorem 3.3 *For a given $f \in P_{n,d}$ and $r \in [2, \infty]$,*

- *Algorithm 1 takes at most $\frac{\|f\|_{\mathrm{HS}}^2}{\varepsilon^2}$ many loops before terminating;*
- *for step (4) in Algorithm 1: searching over a uniform sample on S^{n-1} with size $N \ge t[\alpha(n,d,r)]^{2r}$, where $\alpha(n,d,r)$ as in Theorem 3.2, yields a point $v \in S^{n-1}$ such that $\frac{1}{2}\|f\|_r \le |f(v)|$ with probability at least $1 - e^{-t}$;*

- *the output $\tilde{f}$ of Algorithm 1 satisfies the following properties:*

$$\left\| f - \tilde{f} \right\|_r \leq \varepsilon \ , \ \operatorname{srank}(\tilde{f}) \leq \#\{ \textit{ loops before termination of Algorithm 1 } \} \leq \frac{\|f\|_{\mathrm{HS}}^2}{\varepsilon^2}.$$

3.1 Upper Bound for the Number of Steps in Algorithm 1

The energy increment method gives a general strategy to set up a greedy procedure to decompose a given object into "structured", "pseudorandom", and "error" parts [33, 45]. In what follows, we apply this strategy to obtain a low-rank approximation for a symmetric tensor.

Lemma 3.4 (Greedy Approximation) *Let $(H, \langle \cdot, \cdot \rangle)$ be an inner product space, $\tau : H \to [0, \infty)$ a cost function, and suppose $S \subset B_H = \{z \in H : \|z\|_H^2 = \langle z, z \rangle = 1\}$ separates points in H with respect to τ, that is,*

$$\tau(f) \leq \sup_{w \in S} |\langle f, w \rangle|, \quad f \in H.$$

Then, given $f \in H$ and $\varepsilon > 0$ there exist m points $w_1, \ldots, w_m \in S$ with $m \leq \lfloor \|f\|_H^2 / \varepsilon^2 \rfloor$ and scalars $\lambda_1, \ldots, \lambda_m$ such that

$$\tau\left(f - \sum_{i=1}^{m} \lambda_i w_i \right) \leq \varepsilon.$$

Proof of Lemma 3.4 To begin with, we assume that for the given $f \in H$ and $\varepsilon > 0$, we have $\tau(f) > \varepsilon$. Then, by the separation property, there exists $w_1 \in S$ so that $|\langle f, w_1 \rangle| > \varepsilon$. Now, let $W_1 := \operatorname{span}\{w_1\}$, $p_1 := P_{W_1}(f)$ be the orthogonal projection of f onto W_1, and note that

$$\varepsilon < |\langle w_1, f \rangle| = |\langle w_1, p_1 \rangle| \leq \|p_1\|_H.$$

If $\tau(f - p_1) \leq \varepsilon$ the process stops. If $\tau(f - p_1) > \varepsilon$, then by the separation property again, there exists $w_2 \in S$ so that

$$\varepsilon < |\langle f - p_1, w_2 \rangle| = |\langle p_2 - p_1, w_2 \rangle| \leq \|p_2 - p_1\|_H,$$

where $p_2 := P_{W_2}(f)$ and $W_2 := \operatorname{span}\{w_1, w_2\}$. If $\tau(f - p_2) \leq \varepsilon$, the process stops. If $\tau(f - p_2) > \varepsilon$, we repeat. After m steps, we have extracted $w_1, \ldots w_m \in S$, built the flag of finite-dimensional subspaces

$$\{\mathbf{0}\} = W_0 \subset W_1 \subset \cdots \subset W_m$$
$$W_s = \operatorname{span}\{w_1, \ldots, w_s\}, \ s = 1, \ldots, m,$$

and the lattice of their corresponding orthogonal projections P_{W_s}, $s = 1, \ldots, m$ with $\|p_s - p_{s-1}\|_H > \varepsilon$, where $p_s = P_{W_s}(f)$ for $s = 1, \ldots, m$ (here $p_0 = P_{W_0} = \mathbf{0}$).

Claim. This process terminates after at most m steps where $m < \|f\|_H^2/\varepsilon^2$, that is $\tau(f - p_m) \leq \varepsilon$.
Proof of Claim. Indeed, we may write

$$\|f\|_H^2 \geq \|p_m\|_H^2 = \left\| \sum_{s=1}^{m} (p_s - p_{s-1}) \right\|_H^2 = \sum_{s=1}^{m} \|p_s - p_{s-1}\|_H^2,$$

where we have used that $\langle p_k - p_{k-1}, p_\ell - p_{\ell-1} \rangle = 0$ for $k < \ell$. Since $\|p_s - p_{s-1}\|_H > \varepsilon$, the claim is proved. To complete the proof of the lemma notice that $p_m \in W_m$, hence $p_m = \sum_{i=1}^{m} \lambda_i w_i$ for some scalars $\lambda_1, \ldots, \lambda_m$. □

The intuition suggested by the lemma is easy to express: As long as one uses a cost function τ that is upper bounded by $\sup_{w \in S} |\langle f, w \rangle|$, Lemma 3.4 gives a greedy approximation to input object f with controlled distance in terms of the cost τ.

Proof of Theorem 3.1 We use the set $S := \{\underbrace{v \otimes v \otimes v \otimes \cdots \otimes v}_{\text{d times}} : v \in S^{n-1}\}$, the inner product $\langle ., . \rangle_{\text{HS}}$, and the cost function $\|.\|_r$ to set up the greedy approximation outlined in Lemma 3.4. The proof relies on the following observations:

(1) $\|g\|_r \leq \|g\|_\infty = \sup_{q \in S} |\langle g, q \rangle_{\text{HS}}|$ for all $g \in P_{n,d}$ and all $2 \leq r \leq \infty$,
(2) if one follows the proof of Lemma 3.4 applied to our specific case, one observes that $w_i = v_i \otimes v_i \cdots \otimes v_i$ for some $v_i \in S^{n-1}$.

Therefore, $\text{srank}(\sum_{i=1}^{m} \lambda_i w_i) \leq m \leq \frac{\|f\|_{\text{HS}}^2}{\varepsilon^2}$. □

3.2 Bounds on the Sample Size for Executing the Step (4) in Algorithm 1

This section is to prove Theorem 3.2. We start with proving a reverse Hölder inequality for symmetric tensors.

Lemma 3.5 *Let $p \in P_{n,d}$, then for $n \geq 2d$ and $k \in [2, n/d]$, we have*

$$\|p\|_k \leq (Ck)^{d/2} \|p\|_2,$$

where $C > 0$ is an absolute constant.

Proof of Lemma 3.5 Let $Z \sim N(\mathbf{0}, I_n)$ be a standard Gaussian vector in $\mathbb{R}^n$. We will make use of the following facts:

Fact 3.6 $Z/\|Z\|_2$ is uniformly distributed on S^{n-1} and $\|Z\|_2$ is independent of $Z/\|Z\|_2$. Thereby, for $r > 0$, it follows that

$$\mathbb{E}|p(Z)|^r = \mathbb{E}\|Z\|_2^{rd} \cdot \|p\|_{L_r}^r.$$

For a proof, the reader is referred to [42]. The next fact is a consequence of the Gaussian hypercontractivity, see, e.g., [4, Proposition 5.48.].

Fact 3.7 For any tensor Q of degree at most d and for every $r \geq 2$, one has

$$\left(\mathbb{E}|Q(Z)|^r\right)^{1/r} \leq (r-1)^{d/2}\left(\mathbb{E}|Q(Z)|^2\right)^{1/2}.$$

Finally, we need the asymptotic behavior of high-moments of $\|Z\|_2$.

Fact 3.8 For $r > 0$, we have $\mathbb{E}\|Z\|_2^r = 2^{r/2}\Gamma(\frac{n+r}{2})/\Gamma(\frac{n}{2})$. This follows by switching to polar coordinates. Therefore, for $r > 0$, Stirling's approximation yields

$$\left(\mathbb{E}\|Z\|_2^r\right)^{1/r} \asymp \sqrt{n+r}.$$

Finally, taking into account the above facts, we may write

$$\|p\|_k^k = \frac{\mathbb{E}|p(Z)|^k}{\mathbb{E}\|Z\|_2^{kd}} \leq \frac{(k-1)^{\frac{kd}{2}}(\mathbb{E}|p(Z)|^2)^{k/2}}{\mathbb{E}\|Z\|_2^{kd}} \leq (k-1)^{\frac{dk}{2}}\|p\|_2^k \frac{\left(\mathbb{E}\|Z\|_2^{2d}\right)^{k/2}}{\mathbb{E}\|Z\|_2^{kd}}.$$

Using the estimate for the moments of $\|Z\|_2$, we obtain

$$\|p\|_k \leq (Ck)^{d/2}\left(\frac{n+2d}{n+kd}\right)^{d/2}\|p\|_2,$$

and the result follows. □

Proof of Theorem 3.2 First, note that we may write

$$\begin{aligned}\mathbb{P}\left(\max_{i\leq N}|p(X_i)| < \frac{1}{2}\|p\|_r\right) &= \left[\mathbb{P}\left(|p(X_1)| < \frac{1}{2}\|p\|_r\right)\right]^N \\ &= \left[1-\mathbb{P}\left(|p(X_1)| \geq \frac{1}{2}\|p\|_r\right)\right]^N \\ &\leq \exp\left(-N\mathbb{P}\left(|p(X_1)| \geq \frac{1}{2}\|p\|_r\right)\right).\end{aligned}$$

Second, we provide a lower bound for the probability $\mathbb{P}\left(|p(X_1)| \geq \frac{1}{2}\|p\|_r\right)$. By the Paley–Zygmund inequality, we obtain

$$\mathbb{P}\left(|p(X_1)| \geq \frac{1}{2}\|p\|_r\right) \geq (1-2^{-r})^2\frac{\|p\|_r^{2r}}{\|p\|_{2r}^{2r}}.$$

To bound the ratio $\|p\|_{2r}/\|p\|_r$, we employ Lemmas 2.1 and 3.5 as follows:

$$\|p\|_{2r} \leq (C_1 r)^{d/2}\|p\|_2 \leq (C_1 r)^{d/2}\|p\|_r$$

so

$$\|p\|_{2r} \leq \|p\|_\infty \leq \binom{rd+n-1}{rd}^{\frac{1}{2r}} \|p\|_r .$$

Therefore,

$$\frac{\|p\|_{2r}}{\|p\|_r} \leq \min\{(c_1 r)^{d/2}, \binom{rd+n-1}{rd}^{\frac{1}{2r}}\},$$

which completes the proof. □

3.3 Comparison with Earlier Results and Open Questions

First, let us write a consequence of Theorem 3.1 for an easier interpretation.

Corollary 3.9 *For $r \in [2, \infty]$, $\|.\|_r$ denotes the L_r-norm on $P_{n,d}$. Then, for any $f \in P_{n,d}$ and for any $0 < \delta < 1$, there exists a $q \in P_{n,d}$ with $\|f - q\|_r \leq \frac{\|f\|_{\mathrm{HS}}}{(1-\delta)\sqrt{n}}$ and* $\operatorname{srank}(q) \leq n(1-\delta)^2$.

To bring this result to its most simple form: for the case of symmetric matrices and operator norm, i.e., $d = 2$ and $r = \infty$, this result says that the closest singular matrix w.r.t. to the operator norm is at most $\frac{\|f\|_{\mathrm{HS}}}{\sqrt{n}}$ away. Therefore, in this very special case, the result seems to be tight; one can consider the case where all singular values of f are equal and use the Eckart–Young theorem. However, for general tensor spaces equipped with L_r-norms, for moderately small r, the result does not seem to be tight. The following problem remains open:

Open Problem 3.10 Obtain sharp estimates on the approximate symmetric rank with respect to all L_r-norms for $r \in [2, \infty)$ and for all $P_{n,d}$.

The main of result of [7] combined with the celebrated Alexander–Hirschowitz Theorem, see, e.g., [9], provides a bound for the srank of real symmetric tensors. In particular, the srank is typically between $\frac{1}{n}\binom{n+d-1}{d}$ and $\frac{2}{n}\binom{n+d-1}{d}$ for $d > 2$ except for the cases $(n, d) \in \{(3, 4), (4, 4), (5, 4), (5, 3)\}$. This beautiful result coming from algebraic geometry is exact, static and it universally holds for any symmetric d-tensor. Our estimate in Theorem 3.1, and later in Theorem 4.1, are approximate, dynamic, and give a different estimate depending on the norm of the input. This basically shows that the symmetric rank and approximate symmetric rank are different in nature. Note that we fix $\varepsilon > 0$, that is, the approximate rank notion is also different from the rank notions that require taking limits.

If one is still interested in strict comparison, Theorem 3.1 improves upon the algebraic geometry estimate for

$$\ln(\frac{1}{\varepsilon}) < \frac{1}{2}\ln\binom{n+d-1}{d} - \ln\|f\|_{\mathrm{HS}} - \frac{1}{2}\ln n \leq \frac{d-1}{2}\ln n - \ln\|f\|_{\mathrm{HS}}$$

Therefore, for Theorem 3.1 to be useful for small ε, we need $\ln \|f\|_{\mathrm{HS}}$ to be smaller compared to $\frac{d-1}{2} \ln n$. As a rule of thumb, we need $\ln \|f\|_{\mathrm{HS}} \leq \frac{d}{4} \ln n$, and the smaller is the better. To see if this is meaningful for applications, we looked at input models for symmetric tensors that are considered in recent literature. As an example, in [28], the input model for tensors is the following: one samples $a_1, a_2, \ldots, a_K \in S^{n-1}$ where $K = O(n^{\frac{d}{2}})$ in a way makes the collection of rank-one tensors $a_i \otimes a_i \otimes \cdots \otimes a_i$ to have "restricted isometry property". Then, one considers the tensor $p := \sum_{i=1}^{K} a_i \otimes a_i \otimes \cdots \otimes a_i$ and adds a small perturbations to it. That is, we consider $f := p + h$ where h has very small norm, e.g., $\|h\|_{\mathrm{HS}} = O(\frac{1}{n})$. Due to the "restricted isometry property", one has

$$\|p\|_{\mathrm{HS}}^2 \sim \sum_{i=1}^{k} \|a_i \otimes a_i \otimes \cdots a_i\|_{\mathrm{HS}}^2 = K.$$

In the end, the input tensor f has $\|f\|_{\mathrm{HS}} = O(n^{\frac{d}{4}})$, and f is $O(\frac{1}{n})$ close to a tensor p with rank $O(n^{\frac{d}{2}})$. A main result in [28, Theorem 16] is to show that the proposed algorithm (with high probability) removes the "noise" in f, and recovers the decomposition with rank $O(n^{\frac{d}{2}})$. Here, we will consider a much more flexible input model and still obtain a similar result: let $q \in P_{n,d}$ be a symmetric tensor with $\|q\|_{\mathrm{HS}} = O(n^{\frac{d}{4}})$. We impose no further assumptions on q. For instance, if q is a typical input, then it has symmetric rank between $\frac{1}{n}\binom{n+d-1}{d}$ and $\frac{2}{n}\binom{n+d-1}{d}$, that is, for a typical q, we have $\operatorname{srank}(q) = \Omega(n^{d-1})$. For this input tensor q, Theorem 3.1 yields the following: for any $\varepsilon > 0$,

$$\operatorname{srank}_{\|\cdot\|_r,\varepsilon}(q) \leq \frac{O(n^{\frac{d}{2}})}{\varepsilon^2}.$$

The meaning of this is that for a fixed small ε, say $\frac{1}{\varepsilon} = \ln n$, Theorem 3.1 (and Algorithm 1) finds an ε-close symmetric tensor that has rank $O(n^{\frac{d}{2}} \ln^2 n)$.

The usage of random tensors as a testing ground also brings the following problem, which remains open to the best of our knowledge.

Open Problem 3.11 Let f be an isotropic Gaussian, with respect to the inner product $\langle ., . \rangle_{\mathrm{HS}}$, random element of the vector space $P_{n,d}$, and let $\varepsilon > 0$ be fixed. Prove upper and lower bounds, that holds with high probability, for the quantity $\operatorname{srank}_{\|\cdot\|_r,\varepsilon}(f)$ in the range $r \in [2, \infty)$.

The development in this paper is entirely self-contained. Our search to locate earlier appearance of a similar results in the literature yielded only the following. The main result of [16] used for the specific case of symmetric tensors corresponds to our Theorem 3.1 for $r = \infty$: computing with L_∞ is generally intractable, but this nice result was sufficient for the theoretical purposes the authors considered. Our contribution is to prove algorithmic results that hold for all L_r-norms: Algorithm 1 and Theorem 3.3

delineate the trade-off between the computational complexity (the sample size) and the tightness of approximation for the entire range $r \in [2, \infty]$.

There is a vast literature on tensor decomposition algorithms. We do not intend to survey this vast and interesting literature due to following reason: our Algorithm 1 and existing tensor decomposition algorithms in literature have different goals. The goal of Algorithm 1 is to show that the approximation in Theorem 3.1 is efficiently computable as long as the used L_r-norm is efficiently computable, i.e., r is small and independent of n. Existing algorithms on symmetric tensor decomposition aims to solve a much harder problem, that is, to find an optimal low-rank approximation for a given symmetric tensor. This requires finding the "latent" rank-one tensors, and is known to be a hard problem [12, 26]. We do not aim to solve this NP-Hard problem: our algorithm only gives an upper bound for the approximate rank. Practically, Algorithm 1 can be used to pre-process a given tensor before deploying a more expensive tensor decomposition algorithm: most tensor decomposition algorithms require a guess on the rank of the input tensor, for which the guaranteed rank upper bound from Algorithm 1 can be used.

3.4 Implementation of Algorithm 1

We note from the outset that our current implementation is in a preliminary form. Our main goal is to show that the approximate rank estimate in Theorem 3.1 is constructive: a decomposition that realizes the estimate is effectively computable. We do not claim to have a scalable implementation.

We used a Windows 11 PC, with a Intel Core i7 2.3 GHz processor and 32.0 GB installed ram to experiment with the implementation. The code is available on first authors' personal webpage.

(1) We computed L_r-norms by (re)implementing (with Cristancho and Velasco's kind permission) the quadrature rules from [14] in Python. The quadrature rule for computing the L_r-norms is by far the most expensive step of the algorithm.
(2) Theorem 3.2 provides a bound on the sample size for step (4). In practice, as long as one finds a vector that satisfies the requirement in step 4 of Algorithm 1 the computation is correct. For experiments, we fixed a sample size of 100, 000 and loop in case a vector with such characteristics is not found. We observe that even with this fixed sample size a vector with the correct characteristics was always found.
(3) A practical improvement for Algorithm 1 came from the following observation: in the implementation, we put the extra constraint of the new vectors for step 4 should have an angle bigger than $\arccos(0.8)$ with the older ones. This practical trick observably improved the performance. In future work, this idea needs to be improved and analyzed.
(4) For the experiment, we consider a randomly generated n-variate $2d$-tensors of the type

$$f = \sum_{i=1}^{m} c_i \underbrace{v_i \otimes v_i \otimes \cdots \otimes v_i}_{\text{2d times}} + \frac{\varepsilon}{2} \sum_{i_1, i_2, \ldots, i_d} e_{i_1} \otimes e_{i_1} \otimes e_{i_2} \otimes e_{i_2} \otimes \cdots \otimes e_{i_d} \otimes e_{i_d}$$

where $c_1, \ldots, c_m \in \mathbb{R}$ uniformly distributed according to a standard Gaussian, and $v_1, \ldots, v_m$ are uniformly distributed on the n-dimensional sphere. Basically, the input f is a very high-rank symmetric tensor that is $\frac{\varepsilon}{2}$-close to a rank m tensor. We get the following results for different values of m, n, d, r, ε in the experiment:

- For $m = 10, n = 4, 2d = 4, r = 4, \varepsilon = 0.3$, the dimension of the space is 35, and the algorithm found an $\tilde{f}$ of rank 3 for which $\left\| f - \tilde{f} \right\|_r < 0.29$ in 3.43 s.
- For $m = 10, n = 4, 2d = 24, r = 4$, and $\varepsilon = 0.3$, the dimension of the space is 2925, and the algorithm found an $\tilde{f}$ of rank 2 for which $\left\| f - \tilde{f} \right\|_r < 0.21$ in about 4.8 s.
- For $m = 10, n = 6, 2d = 18, r = 4$, and $\varepsilon = 0.3$, the dimension of the space is 33649, and the algorithm an $\tilde{f}$ of rank 1 for which $\left\| f - \tilde{f} \right\|_r < 0.22$ in about 2 min 48 s.
- For $m = 10, n = 8, 2d = 8, r = 8$, and $\varepsilon = 0.3$, the dimension of the search space is 6435, and the algorithm found an $\tilde{f}$ of rank 4 for which $\left\| f - \tilde{f} \right\|_r < 0.29$ in about 6 min 57 s.
- For $m = 14, n = 12, 2d = 10, r = 8$, and $\varepsilon = 0.3$, we were not able to run the algorithm due to quadrature rule taking too much space in memory.

Our experiment enforced our belief that Algorithm 1 is as efficient as the quadrature rule to compute the L_r-norm; the rest of the steps do not create much computational overhead. This is evident from the sensitivity of the computing time to the number of variables rather than the degree of the tensor: the size of the quadrature nodes grows moderately with respect to degree but drastically with respect to variables. A more optimized implementation of the quadrature rule, or a parallelized version, would greatly improve the performance and allow computations with more variables.

4 Approximate Rank Estimate via Sparsification

Algorithm 2 Approximate Rank via Sparsification

1: **Input** $p(x) = \sum_{i=1}^{N} c_i v_i \otimes v_i \cdots \otimes v_i$, $T = T_2(P_{n,d}, \|.\|)$, and $\varepsilon > 0$.
2: μ is the measure supported on set $\{1, 2, \ldots, N\}$ with $\mu(i) = \frac{|c_i|}{\sum_i |c_i|}$
3: Sample $k := \lceil 4\varepsilon^{-2} T^2 (\sum_{i=1}^{N} |c_i|)^2 \rceil$ many elements $\lambda_1, \lambda_2, \ldots, \lambda_k$ from μ.
4: Set $q_k := \frac{1}{k} \sum_{i=1}^{k} \operatorname{sign}(c_{\lambda_i}) v_i \otimes v_i \cdots \otimes v_i$
5: **if** $\|p - q_k\| > \varepsilon$ **then**
6: Return to step 3
7: **else**
8: Set $q = q_k$
9: **end if**
10: **Output** q
11: **Post-condition** $\|p - q\| \leq \varepsilon$, and $\operatorname{srank}(q) \leq 4\varepsilon^{-2} T^2 (\sum |c_i|)^2$

Algorithms and theorems in this section rely on Maurey's empirical method from geometric functional analysis which was presented in the 1980s paper by [39]. Special cases of this lemma have been (re)discovered many times in recent literature, e.g., [5, 27] where further algorithmic results were also obtained. We reproduce Maurey's idea in Sect. 4.1 for expository purposes. Note that the type-2 constant, T_2, was defined in Sect. 2.3.

Theorem 4.1 *Let $\|\cdot\|$ be a norm on $P_{n,d}$ such that $\|v \otimes v \otimes \cdots \otimes v\| \leq 1$ for all $v \in S^{n-1}$. Let T denote type-2 constant of $P_{n,d}$, $\|.\|$, let $\|\cdot\|_*$ denote the nuclear norm. Then, for any $f \in P_{n,d}$ and $\varepsilon > 0$, we have*

$$\operatorname{srank}_{\|\cdot\|,\varepsilon}(f) \leq \frac{4T^2 \|f\|_*^2}{\varepsilon^2}.$$

Algorithm 2 admits any decomposition as an input and gives a low-rank approximation via sparsification. In the specific case of the input being a nuclear decomposition, the algorithm finds an approximation that is a realization of Theorem 4.1.

Theorem 4.2 *Algorithm 2 terminates in ℓ steps with a probability of at least $1 - 2^{-2\ell}$.*

Theorems 4.1 and 4.2 are proved in Sect. 4.1.

4.1 Sparsification via Maurey's Empirical Method

Lemma 4.3 (Empirical Approximation) *Let $(X, \|\cdot\|)$ be a normed space and a set $S \subset B_X := \{x \in X : \|x\| \leq 1\}$. For any $x \in \operatorname{conv} S$ and $m \in \mathbb{N}$, there exist $z_1, \ldots, z_m$ in S (not necessarily distinct) such that*

$$\left\| x - \frac{1}{m} \sum_{j=1}^{m} z_j \right\| \leq \frac{2T_2(X)}{\sqrt{m}}.$$

Proof Since $x \in \operatorname{conv} S$, there exist $v_1, \ldots, v_\ell \in S$ and $\lambda_1, \ldots, \lambda_\ell \in [0, 1]$ with $\lambda_1 + \cdots + \lambda_\ell = 1$ and $x = \lambda_1 v_1 + \cdots + \lambda_\ell v_\ell$. We introduce the random vector Z taking values on $\{v_1, \ldots, v_\ell\}$ with probability distribution $\mathbb{P}$ where $\mathbb{P}(Z = v_i) = \lambda_i$ for $i = 1, 2, \ldots, \ell$. Clearly, $\mathbb{E}[Z] = x$. Now, we apply an empirical approximation of $\mathbb{E}[Z]$ in the norm $\|\cdot\|$. To this end, let $Z_1, \ldots, Z_m$ be a sample, that is, Z_i are independent copies of Z. We set $Y_m := \frac{1}{m} \sum_{j=1}^{m} Z_j$ and note that $\mathbb{E}[Y_m] = \mathbb{E}[Z] = x$. Now, we use a symmetrization argument: introduce Z_i' independent copies of Z_i, whence $\mathbb{E}[Y_m'] = \mathbb{E}[\frac{1}{m} \sum_{i=1}^{m} Z_i'] = x$. Thus, by Jensen's inequality, we readily get

$$\mathbb{E}\|Y_m - x\|^2 = \mathbb{E}\|Y_m - \mathbb{E}\, Y_m'\|^2 \leq \mathbb{E}\|Y_m - Y_m'\|^2 = \frac{1}{m^2} \mathbb{E} \left\| \sum_{j=1}^{m} (Z_j - Z_j') \right\|^2.$$

Next, $Z_i - Z_i'$ are symmetric, whence, if (ε_i) are independent Rademacher random variables, and independent from both Z_i, Z_i', then the joint distribution of $\varepsilon_i (Z_i - Z_i')$

is the same with $(Z_i - Z_i')$. Thereby, we may write

$$\frac{1}{m^2}\mathbb{E}\left\|\sum_{j=1}^{m}(Z_j - Z_j')\right\|^2 = \frac{1}{m^2}\mathbb{E}\left\|\sum_{j=1}^{m}\varepsilon_j(Z_j - Z_j')\right\|^2 \leq \frac{4}{m^2}\mathbb{E}\left\|\sum_{j=1}^{m}\varepsilon_j Z_j\right\|^2$$

where in the last passage, we have applied the triangle inequality and the numerical inequality $(a+b)^2 \leq 2(a^2+b^2)$. Using the definition of the type-2 constant, we have $\mathbb{E}\left\|\sum_{j=1}^{m}\varepsilon_j Z_j\right\|^2 \leq T^2\sum_{j=1}^{m}\|Z_j\|^2 \leq mT^2$, where we have used the fact that $\|Z_j\| \leq 1$ a.s. The result follows from the first-moment method. □

Proof of Theorem 4.1 Let $p \in P_{n,d}$ with $p \neq 0$ and set $p_1 := p/\|p\|_*$. Since the nuclear norm is induced by the convex body $V_{n,d}$, we have that $p_1 \in V_{n,d}$. Hence, by Lemma 4.3, we infer that there exist $v_i \in S^{n-1}$ for $i = 1, 2, \ldots, m$, $m = \left\lceil \frac{4T^2\|p\|_*^2}{\varepsilon^2} \right\rceil$, and $\xi_i \in \{-1, 1\}$ such that $\left\| p_1 - \frac{1}{m}\sum_{i=1}^{m}\xi_i p_{v_i} \right\| \leq \frac{\varepsilon}{\|p\|_*}$, which completes the proof. □

Proof of Theorem 4.2 Using the proof of Lemma 4.3, it follows that $\mathbb{E}\|p - q_k\| \leq \frac{\varepsilon}{4}$. Moreover, we also observe that by Markov's inequality $\mathbb{P}\{\|p - q_k\| > \varepsilon\} \leq \frac{1}{4}$. Thus, the "if" statement at step 5 returns True at the ℓ-th trial with probability at least $1 - 2^{-2\ell}$. □

Remark 4.4 (1) Aiming for better guarantees, i.e., a higher probability estimate of the desired event, one may work with higher moments and apply Kahane–Khintchine inequality.

(2) We should emphasize that the key parameter in the empirical approximation is the "Radamacher type-2 constant $T_2(S)$ of the set S" rather than the Rademacher type of the ambient space X. This simple but crucial observation will permit us to provide tighter bounds in our context (see Theorem 4.7).

4.2 Type-2 Constant Estimates for Norms on Symmetric Tensors

The results of this section hold for any norm, however, in practice, we use the norms that we can efficiently compute. As mentioned earlier, currently our collection of "efficient norms" includes the L_r norms thanks to efficient quadrature rules [14]. Our estimates for the type-2 constants of L_r-norms on $P_{n,d}$ for $2 \leq r \leq \infty$ is as follows:

Theorem 4.5 *Let $(P_{n,d}, L_r)$ be the space of symmetric d-tensors on $\mathbb{R}^n$ equipped with L_r-norm as defined in Sect. 2.2. Then, for $r \in [2, \infty]$, we have*

$$T_2(P_{n,d}, L_r) \lesssim \sqrt{\min\{r, n\log(ed)\}}.$$

Proof of Theorem 4.5 Although the fact that $T_2(L_r(\Omega, \mu)) \lesssim \sqrt{r}$ is well known, see [2], we provide here a sketch of proof for reader's convenience. The proof makes use

of Khintchine's inequality which reads as follows: let ξ_j be independent Rademacher random variables and α_j be arbitrary real numbers, for $j \in \mathbb{N}$. Then, we have

$$\left(\mathbb{E}\left|\sum_j \alpha_j \xi_j\right|^r\right)^{1/r} \leq B_r \left(\sum_j |\alpha_j|^2\right)^{1/2},$$

for some scalar B_r with $B_r = O(\sqrt{r})$. Let $h_1, \ldots, h_N \in L_r$, then we may write

$$\mathbb{E}\left\|\sum_{j=1}^N \xi_j h_j\right\|_{L_r}^r = \int \mathbb{E}\left|\sum_{j=1}^N \xi_j h_j(\omega)\right|^r \mathrm{d}\mu(\omega) \leq B_r^r \int \left(\sum_{j=1}^N |h_j(\omega)|^2\right)^{r/2} \mathrm{d}\mu(\omega),$$

where we have applied Khintchine's inequality for each fixed ω. Now, we recall the following variational argument: for $0 < p < 1$ and for non-negative numbers $u_1, \ldots, u_N$, one has

$$\left(\sum_{j=1}^n u_j^p\right)^{1/p} = \inf\left\{\sum_{j=1}^N u_j \theta_j : \sum_{j=1}^N \theta_j^q \leq 1,\ \theta_j > 0\right\}, \quad q := \frac{p}{p-1} < 0.$$

Note that, for "$p = 2/r$" and for "$u_j = |h_j(\omega)|^r$", after integration, we have

$$\int \left(\sum_{j=1}^N |h_j(\omega)|^2\right)^{r/2} \mathrm{d}\mu(\omega) \leq \int \sum_{j=1}^N u_j(\omega)\theta_j \,\mathrm{d}\mu(\omega) = \sum_{j=1}^N \theta_j \|h_j\|_{L_r}^r,$$

for any choice of positive scalars θ_j so that $\sum_j \theta_j^q \leq 1$.

For the type-2 constant of $(P_{n,d}, \|\cdot\|_\infty)$, we combine the type-2 estimate for L_r along with the fact, which follows from Lemma 2.1, that $c\|\cdot\|_\infty \leq \|\cdot\|_r \leq \|\cdot\|_\infty$ for $r \geq n\log(ed)$.

□

4.3 An Improvement of the Sparsification Estimate

The definition of the type-2 constant considers all vectors $f_i \in P_{n,d}$ and asks for a constant that satisfies $\mathbb{E}_{\xi_1,\ldots,\xi_m} \left\|\sum_{i=1}^m \xi_i f_i\right\|^2 \leq T^2 \sum_{i=1}^m \|f_i\|^2$. However, for our sparsification purposes, we only work with vectors of the type $f_i = v \otimes v \otimes \cdots \otimes v$ for some $v \in S^{n-1}$. Instead of using type-2 constant definition, which considers the entire space $P_{n,d}$, if we can re-do our proofs only focusing on the vectors $f_i = v \otimes v \otimes \cdots \otimes v$, we can improve the estimates; see Remark 4.4. We obtain such an improvement for the case of L_∞-norm using the following Khintchine type inequality.

Theorem 4.6 (Khintchine inequality for symmetric tensors) *Let $x_1, \ldots, x_m$ be vectors in $\mathbb{R}^n$, let $d \in \mathbb{N}$ and $d \geq 2$, then for any subset $S \subset S^{n-1}$, we have*

$$\mathbb{E}_\varepsilon \sup_{z \in S} \left| \sum_{i=1}^m \varepsilon_i \langle x_i, z \rangle^d \right| \leq 2d \left(\sum_{i=1}^m \|x_i\|_2^{2d} \right)^{1/2}.$$

where ε_i are independent Rademacher random variables.

As a consequence of Theorem 4.6, we have

Theorem 4.7 (Improved sparsification for L_∞-norm) *For $f \in P_{n,d}$ and $\varepsilon > 0$, we have*

$$\operatorname{srank}_{\|.\|_\infty, \varepsilon}(f) \leq \frac{8d^2 \, \|f\|_*^2}{\varepsilon^2}.$$

Observe that if $\tau = v \otimes v \otimes \cdots \otimes v$ for some $v \in \mathbb{R}^n$, then we have $\|v\|_2^d = \|\tau\|_\infty$. Also note that for the set $S = S^{n-1}$, we have $\sup_{z \in S} \left| \sum_{j=1}^m \varepsilon_j \langle x_j, z \rangle^d \right| = \left\| \sum_{j=1}^m \varepsilon_j f_j \right\|_\infty$, where $f_i := x_i \otimes x_i \otimes \cdots \otimes x_i$ for $i = 1, 2, \ldots, m$. Hence, by Theorem 4.6, we have

$$\mathbb{E} \left\| \sum_{i=1}^m \varepsilon_i f_i \right\|_\infty \leq 2d \left(\sum_{i=1}^m \|f_i\|_\infty^2 \right)^{1/2}. \tag{6}$$

Following the proof of Theorem 4.1 line by line, but replacing the type-2 estimate from Theorem 4.5 in the proof with the estimate (6), we obtain Theorem 4.7, provided that $\|f_i\|_\infty = 1$.

Remark 4.8 Theorem 4.7 improves Theorem 4.1 if $d^2 < n$, which is the common situation when one works with tensors. Theorem 4.1 also immediately improves Step (3) in Algorithm 2: one can use $k \asymp \frac{d^2 \, \|f\|_*^2}{\varepsilon^2}$ when working with the L_∞-norm.

Proof of Theorem 4.6 To ease the exposition, we present the argument in two steps:
Step 1: Comparison Principle. Let $T \subset \mathbb{R}^m$ and $\varphi_j : \mathbb{R} \to \mathbb{R}$ be functions that satisfy the Lipschitz condition $|\varphi_j(t) - \varphi_j(s)| \leq L_j |t - s|$ for all $t, s \in \mathbb{R}$ and $\varphi_j(0) = 0$ for $j = 1, 2, \ldots, m$. If $\varepsilon_1, \ldots, \varepsilon_m$ are independent Rademacher variables, then

$$\mathbb{E} \sup_{t \in T} \left| \sum_{j=1}^m \varepsilon_j \varphi_j(t_j) \right| \leq 2 \mathbb{E} \sup_{t \in T} \left| \sum_{j=1}^m \varepsilon_j L_j t_j \right|.$$

This is consequence of a comparison principle due to Talagrand [31, Theorem 4.12]). Indeed, let $S := \{(L_j t_j)_{j \leq m} \mid t \in T\}$ and let $h_j(s) := \varphi_j(s/L_j)$. Note that h_j are

contractions with $h_j(0) = 0$ and

$$\mathbb{E}\sup_{t\in T}\left|\sum_{j=1}^{m}\varepsilon_j\varphi_j(t_j)\right| = \mathbb{E}\sup_{s\in S}\left|\sum_{j=1}\varepsilon_j h_j(s_j)\right|.$$

Hence, a direct application of [31, Theorem 4.12] yields

$$\mathbb{E}\sup_{s\in S}\left|\sum_{j=1}\varepsilon_j h_j(s_j)\right| \leq 2\mathbb{E}\sup_{s\in S}\left|\sum_{j=1}^{m}\varepsilon_j s_j\right| = 2\mathbb{E}\sup_{t\in T}\left|\sum_{j=1}^{m}\varepsilon_j L_j t_j\right|,$$

as desired.

Step 2: Defining Lipschitz maps. In view of the previous fact it suffices to define appropriate Lipschitz contractions which will permit us to further bound the Rademacher average from above by a more computationally tractable average. To this end, we consider the function $\varphi : \mathbb{R} \to \mathbb{R}$ which, for $t \geq 0$, it is defined by

$$\varphi(t) := \begin{cases} t^d, & 0 \leq t \leq 1 \\ d(t-1)+1, & t \geq 1 \end{cases},$$

and we extend to $\mathbb{R}$ via $\varphi(-t) = (-1)^d\varphi(t)$ for all t. Note that f satisfies $\|\varphi\|_{\mathrm{Lip}} = d$. Now, we define $\varphi_j : \mathbb{R} \to \mathbb{R}$ by $\varphi_j(t) := \|x_j\|_2^d\varphi(t)$ and notice that $\|\varphi_j\|_{\mathrm{Lip}} = d\|x_j\|_2^d$. Hence, by the comparison principle (Step 2) for $T = \{(\langle z, \bar{x_j}\rangle)_{j\leq m} \mid z \in S^{n-1}\}$, where $\bar{x_j} = x_j/\|x_j\|_2$, we obtain

$$\mathbb{E}\sup_{z\in S^{n-1}}\left|\sum_j \varepsilon_j\langle z, x_j\rangle^d\right| = \mathbb{E}\sup_{z\in S^{n-1}}\left|\sum_j \varepsilon_j\varphi_j(\langle z, \overline{x_j}\rangle)\right| \leq 2d\mathbb{E}\sup_{z\in S^{n-1}}\left|\sum_j \varepsilon_j\|x_j\|_2^d\langle z, \overline{x_j}\rangle\right|.$$

Lastly, we have

$$\mathbb{E}\sup_{z\in S^{n-1}}\left|\sum_j \varepsilon_j\|x_j\|_2^d\langle z, \overline{x_j}\rangle\right| = \mathbb{E}\left\|\sum_j \varepsilon_j\|x_j\|_2^{d-1}x_j\right\|_2,$$

and the result follows by applying the Cauchy–Schwarz inequality and taking into account the fact that $(\varepsilon_j)_{j\leq m}$ are orthonormal in L_2. □

Remark 4.9 Let us point out that if $d \geq 2$ is even, then we may slightly improve the quantity of the datum $(x_i)_{i\leq m}$ on the right hand-side at the cost of a logarithmic term in dimension. Indeed; let $d = 2k$, $k \in \mathbb{N}$, $k \geq 1$. We apply Step 2 for $T = \{(\langle x_j, \theta\rangle^2)_{j\leq m} \mid \theta \in S^{n-1}\}$ and the even contractions $\varphi_j : \mathbb{R} \to \mathbb{R}$ which, for $s \geq 0$,

are defined by $\varphi_i(s) = \min\{\frac{s^k}{k\|x_i\|_2^{2k-2}}, \frac{\|x_i\|_2^2}{k}\}$. Thus, we obtain

$$\mathbb{E}\left\|\sum_{i=1}^m \varepsilon_i f_i\right\|_\infty \leq d\mathbb{E}\left\|\sum_{i=1}^m \varepsilon_i \|x_i\|_2^{d-2} x_i \otimes x_i\right\|_{\text{op}}.$$

One may proceed in various ways to bound the latter Rademacher average. For example, we may employ the matrix Khintchine inequality [47, Exercise 5.4.13.] to get

$$\mathbb{E}\left\|\sum_{i=1}^m \varepsilon_i \|x_i\|_2^{d-2} x_i \otimes x_i\right\|_{\text{op}} \lesssim \sqrt{\log n}\left\|\sum_{i=1}^m \|x_i\|_2^{2d-2} x_i \otimes x_i\right\|_{\text{op}}^{1/2}.$$

Clearly, $\left\|\sum_{i=1}^m \|x_i\|_2^{2d-2} x_i \otimes x_i\right\|_{\text{op}}^{1/2} \leq \left(\sum_{i=1}^m \|x_i\|_2^{2d}\right)^{1/2}$.

4.4 Comparison with Earlier Results and Open Problems

The quality of approximation provided by Algorithm 2 depends on the constant c with the property that $c \geq \|q\|_*$. It is known that computing the best such c, i.e., the nuclear norm (or the nuclear decomposition), is NP-Hard [22]. As mentioned in Sect. 2.3, one can use sum of squares hierarchy to obtain an increasing sequence of lower bounds for symmetric tensor nuclear norm [36]. Practically, one would like to have a quick decreasing sequence of upper bounds to compare against the increasing sequence of lower bounds coming from sum of squares hierarchy.

Open Problem 4.10 Design an efficient randomized approximation scheme (approximating from above) for the symmetric tensor nuclear norm.

Our search for similar results to Theorem 4.1 in the literature yielded the following: Theorem 5 of [17] used for symmetric tensors would roughly correspond to the special case of Theorem 4.1 for Schatten-p norms. The focus of [17] is to demonstrate that separation between different notions of tensor ranks is not robust under perturbation. We work only with srank and impose no restrictions on the employed norm. We show that the type-2 constant and the nuclear norm universally govern the quality of the empirical approximation in Algorithm 2 for any norm.

5 Approximate Rank Estimates via Frank–Wolfe

This section presents a supplementary result for the specific case of using a Euclidean norm in Theorem 4.1. The theoretical result of this section, Corollary 5.2, is not stronger than what one could obtain using Theorem 4.1. The main difference is that the corresponding algorithm does not require any decomposition of the input tensor, but just needs a guess on the nuclear norm. Another important difference is that the algorithm of this section is the only algorithm in this paper that actually finds

Algorithm 3 Approximate Rank via Frank–Wolfe

Require: $\varepsilon > 0$, a starting point $q \in V_{n,d}$, and stepsize strategy $\gamma_k = \frac{2}{k+1}$.
1: Initialize $p_0 = 0$.
2: **for** $k = 0$ to $T - 1$ **do**
3: **if** $\|p_k - q\|_{\mathrm{HS}} < \varepsilon$ **then**
4: Halt, set $\tilde{q} := p_k$, and output $\tilde{q}$.
5: **else**
6: $h_k = \underset{h \in V_{n,d}}{\arg\min} \langle h, \nabla F(p_k) \rangle_{\mathrm{HS}}$
7: $p_{k+1} = p_k + \gamma_k (h_k - p_k)$
8: **end if**
9: **end for**
10: **Post-condition** $\|q - \tilde{q}\|_{\mathrm{HS}} \leq \varepsilon$ and $\mathrm{srank}(\tilde{q}) \leq 8\varepsilon^{-2}$

the "latent" rank-one tensors, and hence is computationally more expensive. Our main purpose is to obtain an alternative proof of Theorem 4.1 for the case Euclidean norms with an easier argument, and we do not hope for computational efficiency. On the other hand, popular tensor decomposition methods, such as [29], report practical efficiency and at the same time involve similar expensive optimization subroutines as the one used in Algorithm 3. This suggests there might be room for experimentation to see if Algorithm 3 is useful for particular benchmark problems, which we have to leave to the interested readers due to time constraints.

The algorithm is based on optimizing an objective function on the Veronese body that was defined in Sect. 2.3. More precisely, given $q \in V_{n,d}$, we consider the objective function

$$F(p) := \frac{1}{2} \|p - q\|_{\mathrm{HS}}^2 ,$$

and we minimize the objective function on $V_{n,d}$. The algorithm, in return, constructs a low-rank approximation of q, and the number of steps taken by the algorithm controls the rank of its output.

Each recursive step in the algorithm is solved directly over the constraint set $V_{n,d}$: therefore, every linear function involved attains the minimum at some extreme point of $V_{n,d}$ given by $\pm v \otimes \cdots \otimes v$ for some $v \in S^{n-1}$. Therefore, the h_i's produced in step 5 are always rank-1 symmetric tensors. In the end, the number of steps of the algorithm controls the srank of the output p_k.

Lemma 5.1 *Algorithm 3 terminates in at most* $\left\lceil 8/\varepsilon^2 \right\rceil$ *many steps.*

Proof of Lemma 5.1 Recall that $F(p) = \frac{1}{2} \|p - q\|_{\mathrm{HS}}^2$, so we have that $\nabla F(p) = -q + p$ for all p. Therefore, for every g_1 and g_2, we have

$$F(g_2) - F(g_1) = \frac{1}{2} \|g_2 - g_1\|_{\mathrm{HS}}^2 + \langle g_2 - g_1, \nabla F(g_1) \rangle_{\mathrm{HS}}.$$

This gives the following:

$$\begin{aligned} F(p_{k+1}) - F(p_k) &= \langle p_{k+1} - p_k, \nabla F(p_k)\rangle + \frac{1}{2}\|p_{k+1} - p_k\|_{\mathrm{HS}}^2 \\ &= \gamma_k \langle h_k - p_k, \nabla F(p_k)\rangle + \frac{1}{2}\gamma_k^2 \|h_k - p_k\|_{\mathrm{HS}}^2 \\ &\leq \gamma_k \langle h_k - p_k, \nabla F(p_k)\rangle + 2\gamma_k^2 \\ &\leq \gamma_k \langle q - p_k, \nabla F(p_k)\rangle + 2\gamma_k^2 \\ &\leq \gamma_k (F(q) - F(p_k)) + 2\gamma_k^2. \end{aligned}$$

Setting $\delta_k = F(p_k) - F(q)$, the inequality reads

$$\delta_{k+1} - \delta_k \leq -\gamma_k \delta_k + 2\gamma_k^2$$

that is

$$\delta_{k+1} \leq (1 - \gamma_k)\delta_k + 2\gamma_k^2.$$

Using $\gamma_k = \frac{2}{k+1}$, we obtain

$$F(p_{k+1}) - F(q) \leq \frac{8}{k+1}.$$

Hence, given a desired level of accuracy $\varepsilon > 0$ the algorithm terminates in at most $\left\lceil \frac{8}{\varepsilon^2} \right\rceil$ steps. □

Note that for any $f \in P_{n,d}$, we have $\frac{f}{\|f\|_*} \in V_{n,d}$. Thus, as a corollary of Lemma 5.1, using $\frac{\varepsilon}{\|f\|_*}$, we obtain the following rank estimate.

Corollary 5.2 *Let $f \in P_{n,d}$, then we have*

$$\mathrm{srank}_{\|\cdot\|_{\mathrm{HS}},\varepsilon}(f) \leq \frac{8\|f\|_*^2}{\varepsilon^2}.$$

Lemma 5.1 controls the number of steps in the Frank–Wolfe type algorithm. Thus, the remaining piece in complexity analysis is to understand the computational complexity of Step 6. First, we observe that $\nabla F(p_k) = -q + p_k$ and $h_k = \arg\min_{h \in V_{n,d}} \langle h, p_k - q\rangle_{\mathrm{HS}}$. In other words, $h_k = q_{v_k}$ for which we have $(q - p_k)(v_k) = \max_{v \in S^{n-1}} (q - p_k)(v)$. Therefore, finding h_k is equivalent to optimizing $q - p_k$ on the sphere S^{n-1}. This optimization step is indeed expensive (NP-Hard for $d \geq 4$). Here, we content ourselves by providing an estimate on the complexity of Step 6.

Lemma 5.3 *Given $p \in P_{n,d}$, one can find $v \in S^{n-1}$ with*

$$|p(v)| \leq \max_{z \in S^{n-1}} |p(z)| \leq \frac{1}{1-\eta^2}|p(v)|$$

by computing at most $O((3d/\eta)^n)$ many pointwise evaluations of p on S^{n-1}.

This lemma follows from a standard covering argument, see Proposition 4.5 of [15] for an exposition. An alternative approach to polynomial optimization is the sum of squares (SOS) hierarchy: for the case of optimizing a polynomial on the sphere using SOS, the best current result seems to be [20, Theorem 1]. This result shows that SOS produces a constant error approximation to $\|p\|_\infty$ of a degree-d symmetric tensor p with n variables in its (nc_n)-th layer, where c_n is a constant depending on n. In terms of algorithmic complexity, this means SOS is proved to produce a constant error approximation with $O(n^n)$ complexity. Therefore, for the cases $d < n$, the simple lemma above seems stronger than state of the art theorems for the sum of squares approach.

Remark 5.4 The Frank–Wolfe algorithm in this section is quite natural. However, we could not locate any earlier use of this algorithm for symmetric tensor decomposition. We do not know the earliest appearance of this idea in different settings; as far as we are able to locate, the beautiful paper [10] deserves the credit.

6 An Application to Optimization

This section concerns the optimization of symmetric d-tensors for even d when $\|p\|_*$ is small. Suppose one has $p = \sum_i c_i v_i \otimes v_i \otimes \cdots \otimes v_i$ where $\sum_i |c_i| \leq c\|p\|_*$ for some constant c. If we are given a decomposition with this property, then we can approximate $\|p\|_\infty$ in a reasonably fast and accurate way: we first apply Algorithm 2 to p, that is, we compute $q \in P_{n,d}$ such that $\|p-q\|_{HS} \leq \varepsilon$ and

$$q = \frac{1}{m}\sum_{i=1}^{m} v_i \otimes v_i \otimes \cdots \otimes v_i,$$

where $\operatorname{srank}(q) = m \leq \left\lceil \frac{c^2\|p\|_*^2}{\varepsilon^2} \right\rceil$. Also notice that

$$\big|\|p\|_\infty - \|q\|_\infty\big| \leq \|p-q\|_\infty \leq \|p-q\|_{\mathrm{HS}} \leq \varepsilon.$$

The next step is to compute $\|q\|_\infty$ and an approach is offered by Lemma 2.1. First, observe that

$$\|q\|_{2k}^{2k} = \frac{1}{m^{2k}} \sum_{1 \leq i_1, i_2, \ldots, i_k \leq m} \int_{S^{n-1}} \prod_{j=1}^{k} \langle x, v_{i_j} \rangle^d \, \sigma(x)$$

and note that there are $\binom{m+k-1}{k} = O(k^m)$ many summands in the expression of $\|q\|_{2k}^{2k}$. In addition, the values of these summands are given by a Gamma-like function at the vectors $v_1, v_2, \ldots, v_m$. Second, observe that for $k \gtrsim \frac{n}{\varepsilon}\ln(ed/\varepsilon)$, we have $(edk/n)^{\frac{n}{2k}} < 1+\varepsilon$. Therefore, for $k > \frac{cn}{\varepsilon}\ln(\frac{ed}{\varepsilon})$ using Lemma 2.1 and Stirling's estimate, one has

$$\|q\|_{2k} \leq \|q\|_\infty \leq \left(\frac{edk}{n}\right)^{\frac{n}{2k}} \|q\|_{2k} \leq (1+\varepsilon)\,\|q\|_{2k}\,.$$

In return, for $k \asymp \frac{n}{\varepsilon}\ln(\frac{ed}{\varepsilon})$, we can calculate

$$\|q\|_{2k} - \varepsilon \leq \|p\|_\infty \leq (1+\varepsilon)\,\|q\|_{2k} + \varepsilon$$

by computing $O\left((\frac{n\ln(ed)}{\varepsilon^2})^m\right)$ many summands. In principle, this approach gives an algorithm that operates in time $O\left((n\ln(ed))^{\frac{c^2\|p\|_*^2}{\varepsilon^2}}\right)$. However, one must be aware of potential numerical issues due to integration of high degree terms.

In addition to the above, there is an alternative approach coming from [19] with advantages in numerical computations. After we compute $q = \frac{1}{m}\sum_{i=1}^m v_i \otimes v_i \otimes \cdots \otimes v_i$, it is possible to exploit the fact that $q \in W := \operatorname{span}\{v_i \otimes v_i \otimes \cdots \otimes v_i : 1 \leq i \leq m\}$ and $\dim W \leq \left\lceil \frac{c^2\|p\|_*^2}{\varepsilon^2} \right\rceil$. The approach presented in Theorem 1.6 of [19] gives a $1-\frac{1}{n}$ approximation to $\|q\|_\infty$ using $O\left(n^{\frac{c^2\|p\|_*^2}{\varepsilon^2}}\right)$ many pointwise evaluations. Moreover, this approach has the advantage of being simple and using only degree-d tensors. The following theorem summarizes the discussion in this section.

Theorem 6.1 *Let $p = \sum_i c_i v_i \otimes v_i \otimes \cdots \otimes v_i$ where $\sum_i |c_i| \leq c\,\|p\|_*$. Then, using Algorithm 2 and the results of* [19]*:*

- *we compute a $q \in P_{n,d}$ such that $\operatorname{srank}(q) \leq \frac{c^2\|p\|_*^2}{\varepsilon^2}$ and $\left|\|p\|_\infty - \|q\|_\infty\right| \leq \varepsilon$,*
- *we compute a $1-\frac{1}{n}$ approximation of $\|q\|_\infty$, with high probability, using $O(n^{\frac{c^2\|p\|_*^2}{\varepsilon^2}})$ many pointwise evaluations of q on the sphere S^{n-1}.*

Acknowledgements We thank Jiawang Nie for answering our questions on optimization of low-rank symmetric tensors using sum of squares. We thank Sergio Cristancho and Mauricio Velasco for explaining the mathematical underpinning of their quadrature rule in [14], and allowing us to implement it in Python. We thank Carlos Castro Rey for his valuable help and feedback on the Python implementation. A.E. was partially supported by NSF CCF 2110075. P.V. is supported by Simons Foundation grant 638224.

References

1. Acar, E., Yener, B.: Unsupervised multiway data analysis: a literature survey. IEEE Trans. Knowl. Data Eng. **21**(1), 6–20 (2008)

2. Albiac, F., Kalton, N.J.: Topics in Banach Space Theory, Graduate Texts in Mathematics, vol. 233. Springer, New York (2006)
3. Anandkumar, A., Ge, R., Hsu, D., Kakade, S.M., Telgarsky, M.: Tensor decompositions for learning latent variable models. J. Mach. Learn. Res. **15**, 2773–2832 (2014)
4. Aubrun, G., Szarek, S.J.: Alice and Bob meet Banach, Mathematical Surveys and Monographs, vol. 223. American Mathematical Society, Providence (2017). The interface of asymptotic geometric analysis and quantum information theory
5. Barman, S.: Approximating Nash equilibria and dense subgraphs via an approximate version of Carathéodory's theorem. SIAM J. Comput. **47**(3), 960–981 (2018)
6. Barvinok, A.: Estimating L_∞ norms by L_{2k} norms for functions on orbits. Found. Comput. Math. **2**(4), 393–412 (2002)
7. Blekherman, G., Teitler, Z.: On maximum, typical and generic ranks. Math. Ann. **362**(3), 1021–1031 (2015)
8. Brachat, J., Comon, P., Mourrain, B., Tsigaridas, E.: Symmetric tensor decomposition. Linear Algebra Appl. **433**(11–12), 1851–1872 (2010)
9. Brambilla, M.C., Ottaviani, G.: On the Alexander–Hirschowitz theorem. J. Pure Appl. Algebra **212**, 1229–1251 (2008)
10. Combettes, C.W., Pokutta, S.: Revisiting the approximate Caratheodory problem via the Frank–Wolfe algorithm, arXiv preprint. arXiv:1911.04415 (2019)
11. Comon, P.: Independent component analysis, a new concept? Signal Process. **36**(3), 287–314 (1994)
12. Comon, P., Golub, G., Lim, L.-H., Mourrain, B.: Symmetric tensors and symmetric tensor rank. SIAM J. Matrix Anal. Appl. **30**(3), 1254–1279 (2008)
13. Conner, A., Gesmundo, F., Landsberg, J.M., Ventura, E.: Rank and border rank of Kronecker powers of tensors and Strassen's laser method. Comput. Complex. **31**(1), 1–40 (2022)
14. Cristancho, S., Velasco, M.: Harmonic hierarchies for polynomial optimization, arXiv preprint. arXiv:2202.12865 (2022)
15. Cucker, F., Ergür, A.A., Tonelli-Cueto, J.: Functional norms, condition numbers and numerical algorithms in algebraic geometry, arXiv preprint. arXiv:2102.11727 (2021)
16. de la Vega, W.F., Karpinski, M., Kannan, R., Vempala, S.: Tensor decomposition and approximation schemes for constraint satisfaction problems. In: Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing, pp. 747–754 (2005)
17. De las Cuevas, G., Klingler, A., Netzer, T.: Approximate tensor decompositions: disappearance of many separations. J. Math. Phys. **62**(9), 093502 (2021)
18. Derksen, H.: On the nuclear norm and the singular value decomposition of tensors. Found. Comput. Math. **16**(3), 779–811 (2016)
19. Ergür, A.A.: Approximating nonnegative polynomials via spectral sparsification. SIAM J. Optim. **29**(1), 852–873 (2019)
20. Fang, K., Fawzi, H.: The sum-of-squares hierarchy on the sphere and applications in quantum information theory. Math. Program. **190**(1), 331–360 (2021)
21. Fornasier, M., Klock, T., Rauchensteiner, M.: Robust and resource-efficient identification of two hidden layer neural networks. Constr. Approx. **55**(1), 475–536 (2022)
22. Friedland, S., Lim, L.-H.: Nuclear norm of higher-order tensors. Math. Comput. **87**(311), 1255–1281 (2018)
23. Ge, R.: Tensor methods in machine learning. http://www.offconvex.org/2015/12/17/tensor-decompositions/ (2015)
24. Gowers, W.T.: Decompositions, approximate structure, transference, and the Hahn–Banach theorem. Bull. Lond. Math. Soc. **42**(4), 573–606 (2010)
25. Hale, N., Townsend, A.: Fast and accurate computation of Gauss–Legendre and Gauss–Jacobi quadrature nodes and weights. SIAM J. Sci. Comput. **35**(2), A652–A674 (2013)
26. Hillar, C.J, Lim, L.-H.: Most tensor problems are NP-Hard. J. ACM (JACM) **60**(6), 1–39 (2013)
27. Ivanov, G.: Approximate Carathéodory's theorem in uniformly smooth Banach spaces. Discrete Comput. Geom. **66**(1), 273–280 (2021)
28. Kileel, J., Klock, T., Pereira, J.M.: Landscape analysis of an improved power method for tensor decomposition. Adv. Neural Inf. Process. Syst. **34**, 6253–6265 (2021)
29. Kolda, T.G., Mayo, J.R.: An adaptive shifted power method for computing generalized tensor eigenpairs. SIAM J. Matrix Anal. Appl. **35**(4), 1563–1581 (2014)

30. Landsberg, J.M., Teitler, Z.: On the ranks and border ranks of symmetric tensors. Found. Comput. Math. **10**(3), 339–366 (2010)
31. Ledoux, M., Talagrand, M.: Probability in Banach spaces, Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)], vol. 23. Springer, Berlin (1991). Isoperimetry and processes
32. Lim, L.-H.: Tensors in computations. Acta Numer. **30**, 555–764 (2021)
33. Lovász, L., Szegedy, B.: Szemerédi's lemma for the analyst. GAFA Geom. Funct. Anal. **17**(1), 252–270 (2007)
34. Moitra, A.: Algorithmic Aspects of Machine Learning. Cambridge University Press, Cambridge (2018)
35. Nie, J.: Low rank symmetric tensor approximations. SIAM J. Matrix Anal. Appl. **38**(4), 1517–1540 (2017)
36. Nie, J.: Symmetric tensor nuclear norms. SIAM J. Appl. Algebra Geom. **1**(1), 599–625 (2017)
37. Oymak, S., Soltanolkotabi, M.: Learning a deep convolutional neural network via tensor decomposition. Inf. Inference J. IMA **10**(3), 1031–1071 (2021)
38. Panagakis, Y., Kossaifi, J., Chrysos, G.G., Oldfield, J., Nicolaou, M.A., Anandkumar, A.: Tensor methods in computer vision and deep learning. Proc. IEEE **109**(5), 863–890 (2021)
39. Pisier, G.: Remarques sur un résultat non publié de B. Maurey, Séminaire d'Analyse fonctionnelle (dit "Maurey-Schwartz") (1980–1981), talk:5
40. Pratt, K.: Waring rank, parameterized and exact algorithms. In: 2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS). IEEE, pp. 806–823 (2019)
41. Sanyal, R., Sottile, F., Sturmfels, B.: Orbitopes. Mathematika **57**(2), 275–314 (2011)
42. Schechtman, G., Zinn, J.: On the volume of the intersection of two L_p^n balls. Proc. Am. Math. Soc. **110**(1), 217–224 (1990)
43. Sidiropoulos, N.D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E.E., Faloutsos, C.: Tensor decomposition for signal processing and machine learning. IEEE Trans. Signal Process. **65**(13), 3551–3582 (2017)
44. Spielman, D.A.: The smoothed analysis of algorithms. In: International Symposium on Fundamentals of Computation Theory. Springer, pp. 17–18 (2005)
45. Tao, T.: Structure and Randomness. American Mathematical Society, Providence (2008). Pages from year one of a mathematical blog
46. Tomczak-Jaegermann, N.: Banach–Mazur distances and finite-dimensional operator ideals. Pitman Monographs and Surveys in Pure and Applied Mathematics, vol. 38, Pure and Applied Mathematics, vol. 38, 395 (1989)
47. Vershynin, R.: High-dimensional probability, Cambridge Series in Statistical and Probabilistic Mathematics, vol. 47. Cambridge University Press, Cambridge (2018). An introduction with applications in data science, With a foreword by Sara van de Geer