APPROXIMATE LOW-RANK DECOMPOSITION FOR REAL SYMMETRIC TENSORS

ALPEREN A. ERGÜR, JESUS REBOLLO BUENO, PETROS VALETTAS

Abstract. We investigate the effect of an ε-room of perturbation tolerance on symmetric tensor decomposition from an algorithmic perspective. More precisely, we prove theorems and design algorithms for the following problem: Suppose a real symmetric d-tensor f, a norm $\|\cdot\|$ on the space of symmetric d-tensors, and $\varepsilon > 0$ error tolerance with respect to $\|\cdot\|$ 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 provide two different theoretical bounds and three algorithms for approximate symmetric tensor rank estimation. Our first result is a randomized energy increment algorithm for the case of L_p-norms. Our second result is a simple sampling-based algorithm, inspired by some techniques in geometric functional analysis, that works for any norm. We also provide a supplementary algorithm in the case of the Hilbert-Schmidt norm. All our algorithms come with rigorous complexity estimates, which in turn yield our two main theorems on symmetric tensor rank with ε-room of tolerance. We also report on our experiments with a preliminary implementation of the energy increment algorithm.

1. Introduction

Tensor decomposition-based methods were introduced in the context of independent component analysis in the 90s [Com94], and became more widely known for learning latent variable models after [AGH$^+$14]. They are now broadly used in computational science with application domains ranging from phylogenetics to community detection in networks. We suggest [Lim21] as an excellent survey for clarifying basic concepts and for many examples of tensor computations in applied mathematics. Tensor decomposition-based methods are also used for a large range of tasks in machine learning such as training shallow and deep neural nets [FKR22, OS21], ubiquitous applications of the moments method [Moi18, Ge15], computer vision applications [PKC$^+$21], and much more. See [AY08, SDLF$^+$17] for surveys of the vast number 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 in both traditional applied mathematics and in machine learning. This is the main focus of our paper: We study 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 $c_1, \dots, c_m \in \mathbb{R}$ and $v_1, \dots, v_m \in S^{n-1}$ 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).

Note that the symmetric tensor rank is also called CANDECOMP, PARAFAC, CP as a short for these two names, or real-Waring rank after identification with homogeneous polynomials [CGLM08, BCMT10]. We emphasize that in our definition real symmetric tensors are decomposed into rank-1 real tensors, whereas in basic references such as [CGLM08, BCMT10] the main focus is on the decomposition of real symmetric tensors into complex rank-1 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 [Nie17a]. 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 the 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 [Spi05], 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:

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

Our main results, Theorem 3.1, Theorem 3.3, Theorem 3.4, Theorem 3.5, and Theorem 3.8 show that significant theoretical and algorithmic gains become possible by relaxing the symmetric tensor decomposition with an ε-room of perturbation tolerance.

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 on symmetric tensors [CV22, HT13].

The rest of the paper is organized as follows: In section 2 we introduce the vocabulary and basic concepts, in section 3 we state our results and put them in the context of earlier work, and in section 4 we explain the ideas and intuition behind our main results. We provide clean proofs for all of our results and technical tools in Section 5. In section 6 we report on the implementation details of Algorithm 1.

2. Mathematical Concepts

In this section, for the sake of clarity, we explicitly introduce all the mathematical notions used in the statements of the main results in Section 3.

2.1. Basic Terminology and Monomial Index.

We define $T^d(\mathbb{R}^n) := \mathbb{R}^n \otimes \mathbb{R}^n \otimes \cdots \otimes \mathbb{R}^n$ to 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 $u^{(1)} \otimes u^{(2)} \otimes \ldots \otimes u^{(d)} \in T^d(\mathbb{R}^n)$ 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 α_j is the number of j's in $\{i_1, i_2, \ldots, i_d\}$, and $d = \alpha_1 + \alpha_2 + \ldots + \alpha_n$. In the monomial index, instead of listing $\binom{d}{\alpha}$ equal entries for all the supersets identified with x^α, we only list their sum 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 [Bar02].

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 $|\alpha| := \alpha_1 + \ldots + \alpha_n = d$. Then, the Hilbert-Schmidt inner product of p and q is given by

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

For simplicity, we define $p_v := \underbrace{v \otimes v \otimes v \otimes \cdots \otimes v}_{\text{d times}}$ for a $v \in S^{n-1}$, then we have the following identity:

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

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 [SSS11] 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 [Gow10]. Considering (1), one may notice that for every $q \in P_{n,d}$

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

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

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

Formulation (4) suggests a semi-definite programming approach for computing $\|.\|_*$ by approximating B_∞ with the sum of squares hierarchy. Luckily for us, this idea is already made rigorous by an expert in the field and the semidefinite program is ready to use [Nie17b].

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 section 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 $x_1, \dots, x_m \in \mathbb{R}^n$ one has

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

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

Lemma 2.4 (Properties of Type-2 Constant)**.** (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, $T_2(\ell_p^n) \lesssim \sqrt{\min\{p, \log n\}}$.*

3. Main Theorems and Algorithms

3.1. **Approximate Low-Rank Decomposition 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 [Tao08]. Our theorems and algorithms in this section are inspired by the energy increment method as we explain in Section 4.1. 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*

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

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

A proof of this theorem is presented in Section 4.1. 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 isomorphic, see Lemma 2.1. So 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 Greedy Approximation via Energy Increment

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

Details on the implementation of steps in Algorithm 1 are explained in Section 6 alongside some experimental results. Here we present theoretical results on the number of loops before termination and a sampling approach for the search step (4).

Theorem 3.2. *Let $n, d \geq 1$ and $2 \leq r \leq n\log(ed)$. Let $f \in P_{n,d}$ 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*

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

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 $N \geq t[\alpha(n,d,r)]^{2r}$ we have

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

The proof of Theorem 3.2 is included in section 5. 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 $\frac{\|f\|_{\text{HS}}^2}{\varepsilon^2}$ many loops before terminating;*
- *for step (4) in Algorithm 1: searching over a uniform sample on S^{n-1} with size $N \geq t[\alpha(n,d,r)]^{2r}$, where $\alpha(n,d,r)$ as in Theorem 3.2, yields a $v \in S^{n-1}$ such that $\frac{1}{2}\|f\|_r \leq |f(v)|$ with probability at least $1 - e^{-t}$;*
- *the output $\tilde{f}$ of Algorithm 1 satisfies the following properties:*

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

3.2. **Approximate Low-Rank Decomposition via Sparsification.** Algorithms and theorems in this section rely on Maurey's empirical method from geometric functional analysis which was presented in an 80s paper by [Pis81]. We explain this beautiful idea of Maurey in Section 4.2. Note that the type-2 constant, T_2, was defined in Section 2.3.

Theorem 3.4. *Let $\|\cdot\|$ be a norm on $P_{n,d}$ with type-2 constant T, $\|\cdot\|_*$ the nuclear norm, and $f \in P_{n,d}$. Then, for any $\varepsilon > 0$*

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

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 3.4.

Theorem 3.5. *Algorithm 2 terminates in ℓ steps with a probability of at least $1 - 2^{-2\ell}$.*

Proofs of Theorem 3.4 and Theorem 3.5 are in Section 4.2.

Algorithm 2 Sparsification for Low-Rank Tensor Decomposition

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 $k := 4\varepsilon^{-2} T^2 (\sum_{i=1}^{N} |c_i|)^2$ 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 $\operatorname{srank}(q) \leq 4\varepsilon^{-2} T^2 (\sum |c_i|)^2$

3.2.1. *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 [CV22]. Our estimates for the type-2 constants of L_r-norms on $P_{n,d}$ for $2 \leq r \leq \infty$, proved in Section 5, is as follows:

Theorem 3.6. *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 Section 2.2. Then, for $r \in [2, \infty]$ we have*

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

3.3. **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 $\mathbb{E}_{\xi_1,\ldots,\xi_m} \|\sum_{i=1}^{m} \xi_i f_i\|^2 \leq T^2 \sum_{i=1}^{m} \|f_i\|^2$. 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 4.3 following Lemma 4.2. We obtain such an improvement for the case of L_∞-norm using the following Khintchine type inequality.

Theorem 3.7 (Khintchine inequality for symmetric tensors)**.** *Let $x_1, \ldots, x_m$ be vectors in $\mathbb{R}^n$, let $d \in \mathbb{N}, d \geq 2$, and define rank-1 symmetric tensors*

$$f_i := \underbrace{x_i \otimes x_i \otimes \cdots \otimes x_i}_{d \text{ times}}$$

for $i = 1, 2, \ldots, m$. 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.

We prove this theorem in Section 5. As a consequence of Theorem 3.7, we have:

Theorem 3.8 (Improved sparsification for L_∞-norm)**.** *For $f \in P_{n,d}$ and $\varepsilon > 0$, we have*

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

Observe that if $f_i = v \otimes v \otimes \cdots \otimes v$ for some $v \in \mathbb{R}^n$ then we have $\|v\|_2^d = \|f_i\|_\infty$. Also note that for the set $S = S^{n-1}$, we have $\sup_{z\in S}\left|\sum_{j=1}^m \varepsilon_j \langle x_j, z\rangle^d\right| = \|\sum_{j=1}^m \varepsilon_j f_j\|_\infty$. Let $f_i := v_i \otimes v_i \otimes \ldots \otimes v_i$ for $i = 1, 2, \ldots, m$, then by Theorem 3.7 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 3.4 line by line, but replacing the type-2 estimate from Theorem 3.6 in the proof with the estimate (6), yields Theorem 3.8, provided that $\|f_i\|_\infty = 1$.

Remark 3.9. Theorem 3.8 improves Theorem 3.4 if $d^2 < n$, which is the common situation when one works with tensors. Theorem 3.4 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.

3.4. **A Frank-Wolfe Type Algorithm.** This section presents a supplementary result for the specific case of using a Euclidean norm in Theorem 3.1. We should note from the outset that the algorithm in this section is mainly for conceptual purposes, i.e., it yields Corollary 3.11. The algorithm is based on optimizing an objective function on the Veronese body that was defined in Section 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 $V_{n,d}$: So every linear

Algorithm 3 Frank-Wolfe for Approximate Low-Rank Decomposition

Require: $\varepsilon > 0$, a starting point $q \in V_{n,d}$, and step-size strategy $\gamma_k = \frac{2}{k+1}$.
1: **for** $k = 0$ to $T - 1$ **do**
2: **if** $\|p_k - q\|_{HS} < \varepsilon$ **then**
3: halt and output p_k.
4: **else**
5: $h_k = \arg\min_{h\in V_{n,d}} \langle h, \nabla F(p_k)\rangle_{HS}$
6: $p_{k+1} = p_k + \gamma_k(h_k - p_k)$
7: **end if**
8: **end for**

function involved attains the minima at some extreme point of $V_{n,d}$ given by $\pm v \otimes \ldots \otimes v$ 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 3.10. *Algorithm 3 terminates in at most $\lceil 8/\varepsilon^2 \rceil$ many steps.*

The proof of Lemma 3.10 is included in Section 5. Note that for any $f \in P_{n,d}$ we have $\frac{f}{\|f\|_*} \in V_{n,d}$. Thus as a corollary of Lemma 3.10, using $\frac{\varepsilon}{\|f\|_*}$, we obtain the following rank estimate.

Corollary 3.11. *Let $f \in P_{n,d}$, then we have*

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

Remark 3.12. The special case of Theorem 3.4 for Hilbert-Schmidt norm also gives Corollary 3.11. The raison d'etre for Section 3.4 is that we obtain Corollary 3.11 with a deterministic and conceptually different algorithm than the sampling-based algorithm attached to Theorem 3.4.

Remark 3.13. We should note that even though computing $\|f\|_*$ can be costly, computing an upper bound $c \geq \|f\|_*$ can be done quickly via linear algebra. Thus finding a c such that $f/c \in V_{n,d}$ is always computationally tractable, and for any such c Algorithm 3 produces an approximate rank decomposition with rank $\frac{8c^2}{\varepsilon^2}$.

Lemma 3.10 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 5. 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 $d \geq 4$). On the other hand, popular tensor decomposition methods, such as [KM14], report practical efficiency and at the same time involve this same optimization step as a subroutine. This suggests there might be room for experimentation to see if Algorithm 3 is useful for particular benchmark problems. Here we content ourselves by providing an estimate on complexity of Step 5.

Lemma 3.14. *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 [CETC21] 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 [FF21, 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. So, for the cases $d < n$ the simple lemma above seems stronger than state of the art theorems for the sum of squares approach.

Remark 3.15. The Frank-Wolfe algorithm in this section is quite natural and, to the best of our knowledge, was not used before 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 [CP19] deserves the credit.

3.5. **An application to optimization.** This section concerns the optimization of symmetric d-tensors for even d when $\|p\|_*$ is small. Suppose one has $p = \sum_i c_i v_i \otimes v_i \otimes \ldots \otimes v_i$ 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

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

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

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

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 $(edk/n)^{\frac{n}{2k}} < 1+\varepsilon$. So, 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 [Erg19] with advantages in numerical computations. After we compute $q = \frac{1}{m}\sum_{i=1}^m v_i \otimes v_i \otimes \ldots \otimes v_i$, it is possible to exploit the fact that $q \in W := \operatorname{span}\{v_i \otimes v_i \otimes \ldots \otimes v_i : 1 \leq i \leq m\}$ and $\dim W \leq \lceil \frac{c^2\|p\|_*^2}{\varepsilon^2} \rceil$. The approach presented in Theorem 1.6 of [Erg19] 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 3.16. *Let $p = \sum_i c_i v_i \otimes v_i \otimes \ldots \otimes v_i$ where $\sum_i |c_i| \leq c\|p\|_*$. Then using Algorithm 2 and the results of* [Erg19]*:*

- *we compute a $q \in P_{n,d}$ such that* $\operatorname{srank} q \leq \frac{c^2\|p\|_*^2}{\varepsilon^2}$ *and* $|\|p\|_\infty - \|q\|_\infty| \leq \varepsilon$,
- *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}.*

3.6. **Discussion on Prior Works and Our Results.**

- The main of result of [BT15] combined with the celebrated Alexander-Hirschowitz Theorem, see e.g. [BO08], 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 estimate coming from algebraic geometry is exact, static and it universally holds for any symmetric d-tensor. Our estimates in Theorem 3.1, Theorem 3.4, and Theorem 3.8 are approximate, dynamic, and give a different estimate depending on the norm of the input.

Our development is entirely self-contained. Our search to locate earlier appearance of perturbed tensor decomposition idea in the literature yielded only the following two references:

- The main result of [dlVKKV05] 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 $r \in [2,\infty]$.
- Theorem 5 of [DlCKN21] used for symmetric tensors would roughly correspond to Theorem 3.4 for Schatten-p norms. The focus of [DlCKN21] 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.

Now we need to compare Algorithm 1 to earlier symmetric tensor decomposition algorithms such as [KKMP21] or [Nie17a]. The main theoretical difference seems to be the existing methods assume that the input symmetric tensor is low-rank, whereas Algorithm 1 admits an input f as long as $\frac{\|f\|_{HS}^2}{\varepsilon^2}$ is relatively small compared to $\frac{1}{n}\binom{n+d-1}{d}$. The bound $\frac{\|f\|_{\mathrm{HS}}^2}{\varepsilon^2}$ can be computed efficiently while the low-rank assumption on the input tensor cannot be certified. In terms of practical performance, the most expensive part of Algorithm 1 seems to be step (4) where we have a sampling approach based on Theorem 3.2. We elaborate on implementation details and experiments with Algorithm 1 in Section 6. Algorithm 2 is very quick but the quality of the approximation depends on the input decomposition of the tensor. This algorithm can be useful for establishing a quick upper bound for the rank of the input, which in turn allows deploying existing established methods that need low-rank assumption on the input. Algorithm 3 is mostly for theoretical purposes (Corollary 3.11).

4. Technical Ideas and Intuition for Proofs

4.1. **Energy Increment.** 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 [Tao08, LS07]. In what follows we apply this strategy to obtain a low-rank approximation for a symmetric tensor.

Lemma 4.1 (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,*

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

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

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

Proof of this lemma is in Section 5, however, the intuition suggested by the lemma can be readily explained: As long as one uses a cost function τ that is upper bounded by $\sup_{w\in S} |\langle f, w\rangle|$, Lemma 4.1 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 $\langle ., .\rangle_{\mathrm{HS}}$, and the cost function $\|.\|_r$ to set up the greedy approximation outlined in Lemma 4.1. The proof relies on the following observations:

(1) $\|g\|_r \le \|g\|_\infty = \sup_{q\in S} |\langle g, q\rangle_{\mathrm{HS}}|$ for all $g \in P_{n,d}$ and all $2 \le r \le \infty$,
(2) if one follows the proof of Lemma 4.1 applied to our specific case, one observes that $w_i = v_i \otimes v_i \cdots \otimes v_i$ for some $v_i \in S^{n-1}$.

So, $\operatorname{srank}(\sum_{i=1}^m \lambda_i w_i) \le m \le \frac{\|f\|_{\mathrm{HS}}^2}{\varepsilon^2}$. ■

4.2. **Sparsification via Empirical Approximation.**

Lemma 4.2 (Empirical Approximation)**.** *Let $(X, \|\cdot\|)$ be a normed space and a set $S \subset B_X := \{x \in X : \|x\| \le 1\}$. For any $x \in \operatorname{conv} S$ and $m \in \mathbb{N}$, there exist $z_1, \dots, z_m$ in S (not necessarily distinct) such that*

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

The idea behind this lemma is simple and quite powerful. It at least goes back to Maurey; see [Pis81]. Special cases of this lemma have been (re)discovered many times in recent literature, e.g., [Bar18, Iva21] where further algorithmic results were also obtained. We give the proof of Lemma 4.2 to explain the intuition behind our approach, and for expository purposes.

Proof. Since $x \in \mathrm{conv}S$ there exist $v_1, \dots, v_\ell \in S$ and $\lambda_1, \dots, \lambda_\ell \in [0,1]$ 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 $\{v_1, \dots, v_\ell\}$ with probability distribution $\mathbb{P}$ where $\mathbb{P}(Z = v_i) = \lambda_i$ for $i = 1, 2, \dots, \ell$. Clearly, $\mathbb{E}[Z] = x$. Now we apply an empirical approximation of $\mathbb{E}[Z]$ in the norm $\|\cdot\|$. To this end, let $Z_1, \dots, Z_m$ 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

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

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

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

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

Proof of Theorem 3.4 . 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 4.2 we infer that there exist $v_i \in S^{n-1}$ for $i = 1, 2, \dots, m$, $m = \lceil \frac{4T^2\|p\|_*^2}{\varepsilon^2} \rceil$, and $\xi_i \in \{-1, 1\}$ such that $\left\|p_1 - \frac{1}{m}\sum_{i=1}^m \xi_i p_{v_i}\right\| \le \frac{\varepsilon}{\|p\|_*}$, which completes the proof. ■

Proof of Theorem 3.5. Using the proof of Lemma 4.2, it follows that $\mathbb{E}\|p - q_k\| \le \frac{\varepsilon}{4}$. Moreover, we also observe that by Markov's inequality $\mathbb{P}\{\|p - q_k\| > \varepsilon\} \le \frac{1}{4}$. Thus, the "if" statement at step 5 returns True at the ℓ-th trial with probability at least $1 - 2^{-2\ell}$. ■

Remark 4.3.
(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. We leave the details to the interested reader.
(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 3.8).

5. Proofs

5.1. Type-2 constant estimates for L_r-norms.

Proof of Theorem 3.6. Although the fact that $T_2(L_r(\Omega, \mu)) \lesssim \sqrt{r}$ is well-known, see [AK06], 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 $h_1, \dots, h_N \in L_r$, then we may write

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

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 $u_1, \dots, u_N$ 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

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

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 $r \geq n\log(ed)$. ■

5.2. **Proof of Khintchine type inequality for symmetric tensors.**

Proof of Theorem 3.7. 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 $j = 1, 2, \dots, m$. If $\varepsilon_1, \dots, \varepsilon_m$ 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 [LT91, 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 [LT91, 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 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 5.1. 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

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

One may proceed in various ways to bound the latter Rademacher average. For example we may employ the matrix Khintchine inequality [Ver18, Exercise 5.4.13.] to get

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

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

5.3. **Proof of sample size bound in Algorithm 1.** The proof of Theorem 3.2 will make use of the following reverse Hölder inequality for symmetric tensors.

Lemma 5.2. *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 5.2. 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 5.3. $Z/\|Z\|_2$ 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 [SZ90]. The next fact is a consequence of the Gaussian hypercontractivity, see e.g., [AS17, Proposition 5.48.].

Fact 5.4. For any tensor Q of degree at most d and for every $r \geq 2$ one has

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

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

Fact 5.5. 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

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

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

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

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

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

To bound the ratio $\|f\|_{2r}/\|f\|_r$, we employ Lemma 2.1 and Lemma 5.2 as follows:

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

so

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

Therefore,

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

which completes the proof. ■

5.4. **Proof of energy increment lemma.**

Proof of Lemma 4.1. 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 $w_1, \dots w_m \in S$, built the flag of finite-dimensional subspaces

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

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 $\lambda_1, \ldots, \lambda_m$. ■

5.5. **Proof of the step count for Algorithm 3.**

Proof of Lemma 3.10. Recall that $F(p) = \frac{1}{2}\|p - q\|_{HS}^2$, so we have that $\nabla F(p) = -q + p$ for all p. Therefore, for every g_1 and g_2 we have

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

This gives the following:

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

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 $\lceil \frac{8}{\varepsilon^2} \rceil$ steps. ■

6. Implementation Details and Experiments with Algorithm 1

We note from the outset that our current implementation is in a preliminary form to get a sense of Algorithm 1. It was run on a Windows 11 PC, with processor AMD Ryzen 5 3400G with Radeon Vega Graphics 3.70 GHz, and 32.0 GB installed ram. Here are the details of our humble experiment.

(1) We computed L_r-norms by (re)implementing (with authors' kind permission) the quadrature rules from [CV22] in Python. We preferred Python to make it eventually accessible to a wider community. However, we must note that Julia seems to very much outperform Python for computing L_r-norms of symmetric tensors. The cost of Python quadrature implementation limited the scale of the experiment.

(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 characteristics is always found.

(3) The projection step (5) of Algorithm 1 was surprisingly fast.

(4) A practical improvement for Algorithm 1 came from the following observation: If the condition on step 4 held for a vector, it also held for vectors close to them, resulting in an inappreciable improvement of the approximation. 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.

(5) For the experiment, we consider the high rank randomly generated n-variate $2d$-tensors of the type

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

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. We get the following results:

- Considering $m = 10$, $n = 4$, $2d = 4$ and $r = 4$, the algorithm, for which the dimension of the space is 35, finds an $\tilde{f}$ of rank 7 for which $\|f - \tilde{f}\|_r < 0.092$ in 5.91 seconds.
- Considering $m = 10$, $n = 8$, $2d = 6$ and $r = 8$, the algorithm, for which the dimension of the space is 1716, finds an $\tilde{f}$ of rank 16 for which $\|f - \tilde{f}\|_r < 0.099$ in 4 minutes and 46 seconds.
- Considering $m = 10$, $n = 8$, $2d = 8$ and $r = 8$, the algorithm, for which the dimension of the space is 6435, finds an $\tilde{f}$ of rank 18 for which $\|f - \tilde{f}\|_r < 0.0125$ in about 3 hours.

7. Acknowledgments

We thank Jiwang 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 [CV22], and allowing us to implement it in python. A.E. was partially supported by NSF CCF 2110075. P.V. is supported by Simons Foundation grant 638224.

References

[AGH+14] Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M Kakade, and Matus Telgarsky, *Tensor decompositions for learning latent variable models*, Journal of machine learning research **15** (2014), 2773–2832. 1

[AK06] Fernando Albiac and Nigel J. Kalton, *Topics in Banach space theory*, Graduate Texts in Mathematics, vol. 233, Springer, New York, 2006. 11

[AS17] Guillaume Aubrun and Stanisław J. Szarek, *Alice and Bob meet Banach*, Mathematical Surveys and Monographs, vol. 223, American Mathematical Society, Providence, RI, 2017, The interface of asymptotic geometric analysis and quantum information theory. 13

[AY08] Evrim Acar and Bülent Yener, *Unsupervised multiway data analysis: A literature survey*, IEEE transactions on knowledge and data engineering **21** (2008), no. 1, 6–20. 1

[Bar02] Alexander Barvinok, *Estimating L_∞ norms by L_{2k} norms for functions on orbits*, Foundations of Computational Mathematics **2** (2002), no. 4, 393–412. 3

[Bar18] Siddharth Barman, *Approximating Nash equilibria and dense subgraphs via an approximate version of Carathéodory's theorem*, SIAM J. Comput. **47** (2018), no. 3, 960–981. 11

[BCMT10] Jerome Brachat, Pierre Comon, Bernard Mourrain, and Elias Tsigaridas, *Symmetric tensor decomposition*, Linear Algebra and its Applications **433** (2010), no. 11-12, 1851–1872. 1

[BO08] Maria Chiara Brambilla and Giorgio Ottaviani, *On the Alexander–Hirschowitz theorem*, Journal of Pure and Applied Algebra (2008), 1229–1251. 9

[BT15] Grigoriy Blekherman and Zach Teitler, *On maximum, typical and generic ranks*, Mathematische Annalen **362** (2015), no. 3, 1021–1031. 9

[CETC21] Felipe Cucker, Alperen A Ergür, and Josué Tonelli-Cueto, *Functional norms, condition numbers and numerical algorithms in algebraic geometry*, arXiv preprint arXiv:2102.11727 (2021). 8

[CGLM08] Pierre Comon, Gene Golub, Lek-Heng Lim, and Bernard Mourrain, *Symmetric tensors and symmetric tensor rank*, SIAM Journal on Matrix Analysis and Applications **30** (2008), no. 3, 1254–1279. 1

[Com94] Pierre Comon, *Independent component analysis, a new concept?*, Signal processing **36** (1994), no. 3, 287–314. 1

[CP19] Cyrille W Combettes and Sebastian Pokutta, *Revisiting the Approximate Caratheodory Problem via the Frank-Wolfe Algorithm*, arXiv preprint arXiv:1911.04415 (2019). 8

[CV22] Sergio Cristancho and Mauricio Velasco, *Harmonic hierarchies for polynomial optimization*, arXiv preprint arXiv:2202.12865 (2022). 2, 6, 16

[DlCKN21] Gemma De las Cuevas, Andreas Klingler, and Tim Netzer, *Approximate tensor decompositions: disappearance of many separations*, Journal of Mathematical Physics **62** (2021), no. 9, 093502. 9

[dlVKKV05] W Fernandez de la Vega, Marek Karpinski, Ravi Kannan, and Santosh Vempala, *Tensor decomposition and approximation schemes for constraint satisfaction problems*, Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, 2005, pp. 747–754. 9

[Erg19] Alperen A Ergür, *Approximating Nonnegative Polynomials via Spectral Sparsification*, SIAM Journal on Optimization **29** (2019), no. 1, 852–873. 9

[FF21] Kun Fang and Hamza Fawzi, *The sum-of-squares hierarchy on the sphere and applications in quantum information theory*, Mathematical Programming **190** (2021), no. 1, 331–360. 8

[FKR22] Massimo Fornasier, Timo Klock, and Michael Rauchensteiner, *Robust and resource-efficient identification of two hidden layer neural networks*, Constructive Approximation **55** (2022), no. 1, 475–536. 1

[Ge15] Rong Ge, *Tensor Methods in Machine Learning*, `http://www.offconvex.org/2015/12/17/tensor-decompositions/` (2015). 1

[Gow10] W. T. Gowers, *Decompositions, approximate structure, transference, and the Hahn–Banach theorem*, Bulletin of the London Mathematical Society **42** (2010), no. 4, 573–606. 4

[HT13] Nicholas Hale and Alex Townsend, *Fast and accurate computation of gauss–legendre and gauss–jacobi quadrature nodes and weights*, SIAM Journal on Scientific Computing **35** (2013), no. 2, A652–A674. 2

[Iva21] Grigory Ivanov, *Approximate Carathéodory's theorem in uniformly smooth Banach spaces*, Discrete Comput. Geom. **66** (2021), no. 1, 273–280. 11

[KKMP21] Joe Kileel, Timo Klock, and João M Pereira, *Landscape analysis of an improved power method for tensor decomposition*, Advances in Neural Information Processing Systems **34** (2021), 6253–6265. 10

[KM14] Tamara G Kolda and Jackson R Mayo, *An adaptive shifted power method for computing generalized tensor eigenpairs*, SIAM Journal on Matrix Analysis and Applications **35** (2014), no. 4, 1563–1581. 8

[Lim21] Lek-Heng Lim, *Tensors in computations*, Acta Numerica **30** (2021), 555–764. 1

[LS07] László Lovász and Balázs Szegedy, *Szemerédi's lemma for the analyst*, GAFA Geometric And Functional Analysis **17** (2007), no. 1, 252–270. 10

[LT91] Michel Ledoux and Michel Talagrand, *Probability in Banach spaces*, Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)], vol. 23, Springer-Verlag, Berlin, 1991, Isoperimetry and processes. MR 1102015 12

[Moi18] Ankur Moitra, *Algorithmic aspects of machine learning*, Cambridge University Press, 2018. 1

[Nie17a] Jiawang Nie, *Low rank symmetric tensor approximations*, SIAM Journal on Matrix Analysis and Applications **38** (2017), no. 4, 1517–1540. 1, 10

[Nie17b] ———, *Symmetric tensor nuclear norms*, SIAM Journal on Applied Algebra and Geometry **1** (2017), no. 1, 599–625. 4

[OS21] Samet Oymak and Mahdi Soltanolkotabi, *Learning a deep convolutional neural network via tensor decomposition*, Information and Inference: A Journal of the IMA **10** (2021), no. 3, 1031–1071. 1

[Pis81] G. Pisier, *Remarques sur un résultat non publié de B. Maurey*, Séminaire d'Analyse fonctionnelle (dit "Maurey-Schwartz") (1980-1981), talk:5. 5, 11

[PKC+21] Yannis Panagakis, Jean Kossaifi, Grigorios G Chrysos, James Oldfield, Mihalis A Nicolaou, Anima Anandkumar, and Stefanos Zafeiriou, *Tensor methods in computer vision and deep learning*, Proceedings of the IEEE **109** (2021), no. 5, 863–890. 1

[SDLF+17] Nicholas D Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E Papalexakis, and Christos Faloutsos, *Tensor decomposition for signal processing and machine learning*, IEEE Transactions on Signal Processing **65** (2017), no. 13, 3551–3582. 1

[Spi05] Daniel A Spielman, *The smoothed analysis of algorithms*, International Symposium on Fundamentals of Computation Theory, Springer, 2005, pp. 17–18. 2

[SSS11] Raman Sanyal, Frank Sottile, and Bernd Sturmfels, *Orbitopes*, Mathematika **57** (2011), no. 2, 275–314. 3

[SZ90] G. Schechtman and J. Zinn, *On the volume of the intersection of two L_p^n balls*, Proc. Amer. Math. Soc. **110** (1990), no. 1, 217–224. 13

[Tao08] Terence Tao, *Structure and randomness*, American Mathematical Society, Providence, RI, 2008, Pages from year one of a mathematical blog. 4, 10

[Ver18] Roman Vershynin, *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. 13