Sum-of-Squares Lower Bounds for Non-Gaussian Component Analysis

Ilias Diakonikolas
Department of Computer Sciences
University of Wisconsin-Madison
Madison, United States
ilias@cs.wisc.edu

Sushrut Karmalkar
Department of Computer Sciences
University of Wisconsin-Madison
Madison, United States
s.sushrut@gmail.com

Shuo Pang
Department of Computer Science
University of Copenhagen
Copenhagen, Denmark
shpa@di.ku.dk

Aaron Potechin
Department of Computer Science
University of Chicago
Chicago, United States
potechin@uchicago.edu

2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS) | 979-8-3315-1674-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/FOCS61266.2024.00063

***Abstract*—Non-Gaussian Component Analysis (NGCA) is the statistical task of finding a non-Gaussian direction in a high-dimensional dataset. Specifically, given i.i.d. samples from a distribution P_v^A on $\mathbb{R}^n$ that behaves like a known distribution A in a hidden direction v and like a standard Gaussian in the orthogonal complement, the goal is to approximate the hidden direction. The standard formulation posits that the first $k-1$ moments of A match those of the standard Gaussian and the k-th moment differs. Under mild assumptions, this problem has sample complexity $O(n)$. On the other hand, all known efficient algorithms require $\Omega(n^{k/2})$ samples. Prior work developed sharp Statistical Query and low-degree testing lower bounds suggesting an information-computation tradeoff for this problem.**

Here we study the complexity of NGCA in the Sum-of-Squares (SoS) framework. Our main contribution is the first super-constant degree SoS lower bound for NGCA. Specifically, we show that if the non-Gaussian distribution A matches the first $(k-1)$ moments of $\mathcal{N}(0,1)$ and satisfies other mild conditions, then with fewer than $n^{(1-\varepsilon)k/2}$ many samples from the normal distribution, with high probability, degree $(\log n)^{\frac{1}{2}-o_n(1)}$ SoS fails to refute the existence of such a direction v. Our result significantly strengthens prior work by establishing a super-polynomial information-computation tradeoff against a broader family of algorithms. As corollaries, we obtain SoS lower bounds for several problems in robust statistics and the learning of mixture models.

Our SoS lower bound proof introduces a novel technique, that we believe may be of broader interest, and a number of refinements over existing methods. As in previous work, we use the framework of [Barak et al. FOCS 2016], where we express the moment matrix M as a sum of graph matrices, find a factorization $M \approx LQL^T$ using minimum vertex separators, and show that with high probability Q is positive semidefinite (PSD) while the errors are small. Our technical innovations involve the following. First, instead of the minimum weight separator used in prior work, we crucially make use of the minimum square separator. Second, proving that Q is PSD poses significant challenges due to an intrinsic reason. In all prior work, the major part of Q was always a constant term, meaning a matrix whose entries are constant functions of the input. Here, however, even after removing a small error term, Q remains a nontrivial linear combination of non-constant, equally dominating terms. We develop an algebraic method to address this difficulty, which may have wider applications. Specifically, we model the multiplications between the "important" graph matrices by an $\mathbb{R}$-algebra, construct a representation of this algebra, and use it to analyze Q. Via this approach, we show that the PSDness of Q boils down to the multiplicative identities of Hermite polynomials.

***Index Terms*—Non-Gaussian component analysis, sum of squares, robust estimation, lower bounds**

Ilias Diakonikolas was supported by NSF Medium Award CCF-2107079. Sushrut Karmalkar was supported by NSF under Grant #2127309 to the Computing Research Association for the CIFellows 2021 Project. Shuo Pang was funded by the European Union MSCA Postdoctoral Fellowships 2023 under project 101146273 NoShortProof. Aaron Potechin was supported by NSF grant CCF-2008920.

I. Introduction

Non-Gaussian Component Analysis (NGCA) is a statistical estimation task first considered in the signal processing literature [BKS+06] and subsequently extensively studied (see, e.g., Chapter 8 of [DK23] and references therein). As the name suggests, the objective of this task is to find a non-Gaussian direction (or, more generally, low-dimensional subspace) in a high-dimensional dataset. Since its introduction, the NGCA problem has been studied in a range of works from an algorithmic standpoint; see [TKM11], [SKBM08], [DJSS10], [DJNS13], [SNS16], [VNO16], [TV18], [GS19], [DH24], [CV23]. Here we explore this problem from a hardness perspective with a focus on Sum-of-Squares algorithms.

2575-8454/24/$31.00 ©2024 IEEE
DOI 10.1109/FOCS61266.2024.00063

The standard formulation of NGCA is the following. Fix a univariate distribution A. For a unit vector direction v, let P_v^A be the distribution on $\mathbb{R}^n$ defined as follows: The projection of P_v^A in the v-direction is equal to A, and its projection in the orthogonal complement is the standard Gaussian distribution. Observe that P_v^A is a product distribution with respect to a non-standard coordinate system. It is further assumed that, for some parameter k, the first $k-1$ moments of the univariate distribution A match those of the standard Gaussian $\mathcal{N}(0,1)$ and the k-th moment differs by a non-trivial amount. Given i.i.d. samples from a distribution P_v^A, for an unknown v, the goal is to estimate the hidden direction v. It is known that, under mild assumptions on the distribution A, this problem has sample complexity $O(n)$. Unfortunately, all known methods to achieve this sample upper bound run in time exponential in n by essentially using brute-force over a cover of the unit sphere to identify the hidden direction. On the other hand, if we have $\gg n^{k/2}$ samples, a simple spectral algorithm (on the k-th moment tensor) solves the problem in sample-polynomial time (see, e.g., [DH24]). A natural question is whether more sample-efficient polynomial-time algorithms exist or if the observed gap is inherent—i.e., the problem exhibits a statistical-computational tradeoff. As our main result, we show (roughly speaking) that the gap is inherent for SoS algorithms of degree $o(\sqrt{\frac{\log n}{\log\log n}})$.

In addition to being interesting on its own merits, further concrete motivation to understand the hardness of NGCA comes from its applications to various well-studied learning problems. Specifically, the NGCA problem captures interesting (hard) instances of several statistical estimation problems that superficially appear very different. The idea is simple: Let Π be a statistical estimation task. It suffices to find a univariate distribution A_Π such that for any direction v the high-dimensional distribution $P_v^{A_\Pi}$ is a *valid instance* of problem Π. Solving Π then requires solving NGCA on these instances. We provide two illustrative examples below.

Example 1: Robust Mean Estimation: Consider the following task, known as (outlier-)robust mean estimation: Given i.i.d. samples from a distribution D on $\mathbb{R}^n$ such that $\mathrm{d_{TV}}(D,\mathcal{N}(\mu,I)) \le \varepsilon$, for some small $\varepsilon > 0$, the goal is to approximate the mean vector μ in ℓ_2-norm. Suppose that A is an ε-corrupted one-dimensional Gaussian distribution in total variation distance, namely a distribution that satisfies $\mathrm{d_{TV}}(A,G) \le \varepsilon$, where $G \sim \mathcal{N}(\delta,1)$, where $\delta \in \mathbb{R}$. For any unit vector v, the distribution P_v^A is an ε-corrupted Gaussian on $\mathbb{R}^n$, i.e., $\mathrm{d_{TV}}(P_v^A,\mathcal{N}(\delta v,\mathrm{Id})) \le \varepsilon$. It is then easy to see that the NGCA task on this family of P_v^A's is an instance of robust mean estimation. Namely, approximating v is equivalent to approximating the target mean vector (once we know v, the one-dimensional problem of estimating δ robustly is easy).

Example 2: Learning Mixtures of Gaussians: A k-mixture of Gaussians (GMM) on $\mathbb{R}^n$ is a convex combination of Gaussians, i.e., a distribution of the form $F(x) = \sum_{i=1}^k w_i \mathcal{N}(\mu_i,\Sigma_i)$ where $\sum_{i=1}^k w_i = 1$. The prototypical learning problem for GMMs is the following: Given i.i.d. samples from an unknown GMM, the goal is to learn the underlying distribution in total variation distance (or, more ambitiously, approximate its parameters). Suppose that A is a k-mixture of *one-dimensional* Gaussians $\sum_{i=1}^k w_i \mathcal{N}(t_i,\delta^2)$ and the t_i are chosen to be sufficiently far apart such that each pair of components has small overlap. For any unit vector v, the distribution P_v^A is a mixture of k Gaussians on $\mathbb{R}^n$ of the form $\sum_{i=1}^k w_i \mathcal{N}(t_i v, \mathrm{Id} - (1-\delta^2)vv^T)$. For small δ, this can be thought of as k "parallel pancakes", in which the means lie in the direction v. All $n-1$ orthogonal directions to v will have an eigenvalue of 1, which is much larger than the smallest eigenvalue in this direction (which is δ). In other words, for each unit vector v, the k-GMM P_v^A will consist of k "skinny" Gaussians whose mean vectors all lie in the direction of v. Once again, the NGCA task on this family of P_v^A's is an instance of learning GMMs: once the direction v is identified, the corresponding problem collapses to the problem of learning a one-dimensional mixture, which is easy to solve.

By leveraging the aforementioned connection, hardness of NGCA can be used to obtain similar hardness for a number of well-studied learning problems that superficially appear very different. These include learning mixture models [DKS17], [DKPZ23], [DKS23], robust mean/covariance estimation [DKS17], robust linear regression [DKS19], learning halfspaces and other natural concepts with adversarial or semi-random label noise [DKZ20], [GGK20], [DK22a], [DKPZ21], [DKK+22], [Tie24], list-decodable mean estimation and linear regression [DKS18], [DKP+21], learning simple neural networks [DKKZ20], [GGJ+20], and even learning simple generative models [CLL22]. Concretely, to achieve this it suffices to find a distribution A *of the required form* that matches as many moments with the standard Gaussian as possible.

Prior Evidence of Hardness: Prior work [DKS17] established hardness of NGCA in a restricted computational model, known as the Statistical Query (SQ) model (see also [DKRS23] for a recent refinement). SQ algorithms are a class of algorithms that are allowed to query expectations of bounded functions on the underlying distribution through an SQ oracle rather than directly access samples. The model was introduced by Kearns [Kea98] as a natural restriction of the PAC model [Val84] in the context of

learning Boolean functions. Since then, the SQ model has been extensively studied in a range of settings, including unsupervised learning [FGR+17].

[DKS17] gave an SQ lower bound for NGCA under the moment-matching assumption and an additional regularity assumption about the distribution A (which was removed in [DKRS23]). Intuitively, the desired hardness result amounts to the following statistical-computational trade-off: Suppose that A matches its first $k-1$ moments with the standard Gaussian. Then any distinguishing SQ algorithm requires either $n^{\Omega(k)}$ samples, where n is the ambient dimension, or super-polynomial time in n.

While the SQ model is quite broad, it does not in general capture the class of algorithms obtained via convex relaxations. With this motivation, in this work we focus on establishing lower bounds for NGCA in the Sum-of-Squares (SoS) framework. It is interesting to note that the classes of SQ algorithms and SoS algorithms are incomparable in general. In the context of NGCA studied here, the work [BBH+21] implies that SQ algorithms are essentially equivalent to low-degree polynomial tests. As a corollary of [BBH+21], one can also deduce low-degree polynomial testing lower bounds for NGCA (see also [MW22] for a direct low-degree testing lower bound of an essentially equivalent problem). We remark that SoS lower bounds that are proved via pseudo-calibration subsume low-degree lower bounds [Hop18]. Interestingly, [GJJ+20] implicitly gave a direct low-degree polynomial lower bound as well. (Specifically, as noted in their Remark 5.9, Attempt 1 for the proof of Lemma 5.7 there shows that w.h.p. $\tilde{E}[1] = 1 \pm o(1)$, which is equivalent to a low-degree polynomial lower bound).

Informal Main Result: Since we are focusing on establishing hardness, we will consider the natural hypothesis testing version of NGCA, noticing that the learning version of the problem typically reduces to the testing problem. Specifically, our goal is to distinguish between a standard multivariate Gaussian and the product distribution that is equal to a pre-specified univariate distribution A in a hidden direction v and is the standard Gaussian in the orthogonal complement.

Problem I.1 (Non-Gaussian Component Analysis, Testing Version)**.** Let A be a one-dimensional distribution that matches the 1st to the $(k-1)$-th moments with $\mathcal{N}(0,1)$. Given m i.i.d. samples $\{x_1, \ldots, x_m\} \subseteq \mathbb{R}^n$ drawn from one of the following two distributions, the goal is to determine which one generated them.

1) (Reference distribution, $\mathcal{D}_{ref}$) The true n-dimensional multivariate normal distribution $\mathcal{N}(0, \mathrm{Id}_n)$.
2) (Planted distribution, $\mathcal{D}_{pl}$) Choose $v \in \{\pm 1/\sqrt{n}\}^n$ uniformly at random (called the hidden/planted direction) and draw $x' \sim \mathcal{N}(0, \mathrm{Id}_n)$, then we take $x = x' - \langle x', v\rangle v + av$, where $a \sim A$, to be the result. In other words, the distribution is $\mathcal{N}(0, \mathrm{Id}_{n-1})_{v^\perp} \times A_v$ where v is chosen uniformly at random from $\{\pm 1/\sqrt{n}\}^n$.

We use Boolean planted directions $v \in \{\pm 1/\sqrt{n}\}^n$ in Definition I.1 for technical convenience. Lower bounds in this setting imply, in particular, that NGCA is hard w.r.t. an adversarial distribution of v.

A necessary condition for this testing problem to be computationally hard is that the univariate distribution A matches its low-degree moments with the standard Gaussian. At a high-level, our main contribution is to prove that this condition is also *sufficient* in the SoS framework (subject to mild additional conditions). Here, we state an informal version of our main theorem, Theorem II.11.

Theorem I.2 (Main Theorem, Informal)**.** *Given n, suppose $2 \le k \le (\log n)^{O(1)}$ and A is a distribution on $\mathbb{R}$ such that:*

1) (Moment matching) The first $k-1$ moments of A match those of $\mathcal{N}(0,1)$.
2) (Moment bounds) $|\mathrm{E}_A[h_t(x)]| \le (\log n)^{O(t)}$ for all Hermite polynomials of degree up to $2k(\log n)^2$, and $\frac{\mathrm{E}_A[p^2(x)]}{\mathrm{E}_{x\sim\mathcal{N}(0,1)}[p^2(x)]} \ge (\log n)^{-O(\deg(p))}$ for all nonzero polynomial $p(x)$ of degree up to $\sqrt{\log n}$.

If n is sufficiently large, then with high probability, given fewer than $n^{(1-\varepsilon)k/2}$ many samples, Sum-of-Squares of degree $o(\sqrt{\frac{\varepsilon \log n}{\log\log n}})$ fails to distinguish between the random and planted distributions for the corresponding NGCA Problem I.1.[1]

The SoS algorithms we consider are semi-definite programs whose variables are all degree $\le D$ monomials in v which represents the unknown planted direction. The constraints are "$\mathrm{v}_i^2 = 1/n$" and that for any low-degree polynomial $p(\cdot)$, the average value of p evaluated on the inner product between v and the samples should be reasonably close to $\mathrm{E}_{x\sim A}[p(x)]$. As a corollary of Theorem I.2, any such algorithm requires either a large number of samples ($\ge n^{(1-\varepsilon)k/2}$ many) or super-polynomial time to solve the corresponding NGCA. Note that the bound here is sharp, since with $O(n^{k/2})$ samples it is possible to efficiently solve the problem (see, e.g., [DH24]).

Remark I.3. We highlight that NGCA can be viewed as a "meta-problem", parameterized by the "structure" of the one-dimensional distribution A, which captures hard instances of a wide variety of learning problems. Our main contribution is to establish SoS-hardness

[1]As is usual for SoS lower bounds for average-case problems, technically what we show is that if we apply pseudo-calibration, the moment matrix is PSD with high probability.

of NGCA for *any* moment-matching distribution A (under mild conditions). This is a powerful result for showing SoS-hardness for other learning problems via reductions. For certain special cases of A—specifically when A is (essentially) a mixture of Gaussians—there exists reduction-based hardness for the problem under cryptographic assumptions (namely, the sub-exponential hardness of LWE) [BRST21], [GVV22]. However, these reductions are tailored to that specific choice of A. For other choices of A, no such reduction-based hardness is known, and it appears that LWE may not be the right starting point. For all other applications in this work, with the exception of learning GMMs, the only prior evidence of hardness was from the aforementioned SQ or low-degree testing lower bounds.

Remark I.4. It is worth noting the special case where the distribution A is discrete. Recent works [DK22b], [ZSWB22] showed polynomial-time algorithms for this version of the problem with sample complexity $O(n)$, regardless of the number of matched moments. Such a result is not surprising, as these algorithms are based on the LLL-method for lattice basis reduction which is not captured by the SoS framework. Importantly, these algorithms are extremely fragile and dramatically fail if we add a small amount of "noise" to A.

Applications to Robust Statistics and Mixture Models: Our main result (Theorem I.2) implies information-computation tradeoffs in the SoS framework for a range of fundamental problems in learning theory and robust statistics. (See Table I for a description of the problems we consider and the guarantees we obtain.) For the problems we consider, SQ and low-degree testing lower bounds were previously known.

At a high level, for all our problems, our SoS lower bounds follow using the same template: we show that for *specific* choices of the distribution A, the problem NGCA is an instance of a hypothesis testing problem known to be efficiently reducible to the learning problem in question. As long as the one-dimensional moment-matching distribution A in question satisfies the hypotheses required for our main theorem to hold, we directly obtain an SoS lower bound for the corresponding hypothesis testing problem.

As an illustrative example, we explain how to reduce the hypothesis testing version of special NGCA instances to the problem of learning a mixture of k Gaussians in n dimensions. Consider the distribution $A = \sum_{k=1}^{n} \frac{1}{k} \mathcal{N}(\mu_i, \sigma_i^2)$, which is a mixture of Gaussians, and let the planted distribution be given by $\mathcal{N}(0, \mathrm{Id}_{n-1})_{v^\perp} \times A_v$ where the hidden direction v is from $\{\pm\frac{1}{\sqrt{n}}\}^n$. Expanding the expression, we see that the hypothesis testing problem is exactly to distinguish a true Gaussian from the mixture $\sum_{i=1}^{k} \frac{1}{k} \mathcal{N}(\mu_i v, \mathrm{Id} - (1-\sigma_i^2)vv^T)$.

Table I gives a list of problems where this work shows SoS lower bounds. It compares the information-theoretic sample complexity (the minimum sample size achievable by any algorithm) with the "computational" sample complexity implied by our SoS lower bound.

A. Technical Overview of the Lower Bound

In this section, we provide a brief high-level overview of our lower bound proof.

Pseudo-calibration technique and graph matrices. We employ the general technique of pseudo-calibration as introduced in [BHK+16] to produce a suitable candidate SoS solution $\widetilde{E}$. For a given degree D, this solution can be described by a moment matrix indexed by sets $I, J \subseteq [n]$ where $|I|, |J| \leq D$. The matrix entries are defined as $M_{\widetilde{E}}(I, J) := \widetilde{E}(\mathrm{v}^{I+J})$, with v^{I+J} representing the monomial $\prod_{i=1}^{n} \mathrm{v}_i^{I(i)+J(i)}$. These entries are functions of the input $x_1, \ldots, x_m \in \mathbb{R}^n$ and are expressed in terms of Hermite polynomials (refer to Definition II.2 and Equation (4)).

As in most SoS lower bounds, the most challenging part is to show that $M_{\widetilde{E}}$ is PSD. We provide an overview of the new ideas required for the proof in the proceeding discussion. Similar to prior works such as [BHK+16], [GJJ+20], [PR20], [JPR+21], [Pan21], [JPRX23], we expand $M_{\widetilde{E}}$ as a linear combination of special matrices called graph matrices [AMP16], whose spectral norm we can bound in terms of combinatorial properties of the underlying *shapes*[2]. Using graph matrices, we carefully factorize M as $M = LQL^\top +$ (error terms) where M is $M_{\widetilde{E}}$ rescaled for technical convenience, thus reducing the task to showing that $Q \succ 0$. Here, the construction of matrices L, Q in the factorization uses a recursive procedure like in previous works, where we repeatedly use minimum vertex separators of a shape to decompose the graph, and hence the graph matrix, in a canonical way.

Minimum square separators. The first technical novelty in this work is the introduction of the *minimum square vertex separators* in the factorization of M. Rather than using the minimum weight vertex separator or the sparse minimum vertex separator as in previous works, we define this new concept for bipartite graphs with two types of vertices——circles and squares——which naturally arise in our analysis of the NGCA problem.

Choosing the correct notion of vertex separators is a crucial first step in our analysis. This is because the combinatoroial properties of minimum square separators and minimum weight separators are key to controlling

[2]A shape is, roughly speaking, a graph plus two distinguished vertex subsets called the left and right indices. The two indices are used to identify rows and columns of the associated matrix. The reader is referred to the full version of the paper for a formal definition.

Sample Complexity versus Computational Sample Complexity

Statistical Estimation Task	Information-Theoretic	Degree-$O(\sqrt{\frac{\varepsilon \log n}{\log\log n}})$ SoS
RME ($\Sigma \preceq \mathrm{Id}$) to ℓ_2-error $\Omega(\sqrt{\tau})$	$O(n)$	$\Omega(n^{2(1-\varepsilon)})$
RME ($\Sigma = \mathrm{Id}$) to ℓ_2-error $\Omega(\frac{\tau \log(1/\tau)^{1/2}}{k^2})$	$O(n)$	$\Omega(n^{k(1-\varepsilon)/2})$
List-decodable Mean Estimation to error $O(\tau^{-1/k})$	$O(n)$	$\Omega(n^{k(1-\varepsilon)/2})$
RCE (multiplicative) to constant error	$O(n)$	$\Omega(n^{2(1-\varepsilon)})$
RCE (additive) to spectral error $O(\frac{\tau \log(1/\tau)}{k^4})$	$O(n)$	$\Omega(n^{k(1-\varepsilon)/2})$
Estimating k-GMM	$\widetilde{O}(kn)$	$\Omega(n^{k(1-\varepsilon)})$
Estimating 2-GMM (common unknown covariance)	$O(n)$	$\Omega(n^{2(1-\varepsilon)})$

TABLE I: A contrast between the information-theoretic sample complexity and the sample complexity required by degree-$O(\sqrt{\varepsilon \log n/\log\log n})$-SoS for a range of natural tasks in robust statistics and learning mixture models. This includes robust mean estimation (RME), robust covariance estimation (RCE), learning Gaussian mixture models, and list-decodable mean estimation. The parameter τ, when it appears, is related to the proportion of contamination.

the norms of all the error terms generated in the resulting factorization $M \approx LQL^\top$, which we will use throughout our analysis. Importantly, the use of minimum square separators leads to a characterization of the dominant terms in the expansion of Q, which we describe now.

The dominant family in Q and well-behaved products. Recall that our goal is to show that with high probability, the matrix Q from factorization $M \approx LQL^\top$ is positive-definite. We view Q as a linear combination, where each term is a graph matrix multiplied by its coefficient. In all prior works that utilize the factorization approach [BHK+16], [GJJ+20], [PR20], [JPR+21], [Pan21], [JPRX23], the dominant term in Q was a constant term, i.e., a matrix whose entries are numbers independent of the input. Here, however, we encounter a new and intrinsic difficulty: Q contains an entire family of non-constant terms that are almost equally dominant.

Using the refined tools developed in our error analysis, we are able to characterize the shapes of the dominant terms, which we refer to as *simple spider disjoint unions (SSD)*. The formal definition is given in the full paper. A related but less restrictive family of shapes, called "spiders", was introduced in [GJJ+20] in the context of the Sherrington-Kirkpatrick problem, which can be seen as a special case of NGCA where the unknown distribution A is the uniform distribution on $\{\pm 1\}$. Their technique of using the null-space to annihilate all spiders relies on A being a discrete distribution, which does not apply to our setting. Additionally, we note that their work establishes a sample complexity lower bound of $n^{3/2}$, in contrast to the $O(n^2)$ upper bound [DH24]. To achieve an almost optimal lower bound of $n^{(1-\varepsilon)k/2}$ (see the second paragraph of the introduction), we need to study of the 'rigid' structure of these shapes and their linear combinations.

As discussed earlier, the dominant terms in Q are simple spider disjoint union graph matrices. To prove that their sum in Q is positive-definite with high probability, we begin by examining the recursive factorization procedure that generates Q. Roughly speaking, Q is a sum of numerous matrix products derived during the factorization. Among these products, we identify those that significantly impact Q—referred as the well-behaved intersection configurations —and show that the remaining other terms altogether contribute minimally. This leads to an expression of a dominating part of Q, which we denote by Q_{SSD}, along with a characterizing equation $L_{\mathrm{SSD}} *_{\mathrm{wb}} Q_{\mathrm{SSD}} *_{\mathrm{wb}} L_{\mathrm{SSD}}^\top = M_{\mathrm{SSD}}$. Here, $L_{\mathrm{SSD}}, M_{\mathrm{SSD}}$ denotes the SSD part of L, M respectively, and $*_{\mathrm{wb}}$ denotes what we call the *well-behaved product* between graph matrices. The formal definitions and statements are given in the full paper.

Before overviewing the proof of the positive-definiteness of Q_{SSD}, we make two important remarks. First, the coefficients of the graph matrices in L and M are delicate. For instance, in L, the coefficient of a simple spider is $n^{-|E|/2} \cdot \mathrm{E}_A[h_j]$, where j denotes the degree of the unique circle vertex of the spider, and h_j is a Probabilist's Hermite polynomial; for disjoint union shapes, the coefficient is the product of those of its components. When matrices multiply, the coefficients multiply as well, and at several places we need to handle them in an exact way rather than doing mere magnitude estimates. Second, and more subtly, we will not analyze the matrix L_{SSD} or M_{SSD} in the same way we analyze Q_{SSD}, as both of them contain terms with larger norms. Instead, we focus our analysis on Q_{SSD}.

PSDness via representation. To show that Q_{SSD} is positive-definite, we start with simple spiders. We use an algebraic method to study their multiplicative structure. The multiplication of general graph matrices is very complicated, but for simple spiders, an algebraic study turns out to be feasible. It goes as follows.

First, we show that Q_{SS}—a further restricted matrix

that collects all simple spider terms in Q_{SSD}—is positive-definite in a non-standard sense. Specifically, we model the multiplications of simple spiders as an associative $\mathbb{R}$-algebra, which we call SA_D (*simple-spider algebra of degree* D). The multiplication in SA_D is the well-behaved product restricted by taking only simple spiders in the result. This is a non-commutative algebra, and it approximates the major terms in the multiplication of simple spiders graph matrices in special cases, although not always. To understand the structure of SA_D, we construct essentially all its irreducible representations and obtain a **concrete** Artin-Wedderburn decomposition as a direct sum of matrix algebras. Details of the construction are given in the full paper. This decomposition greatly simplifies the objects under study: it maps elements of SA_D, which represent graph matrices of dimension $n^{\Theta(D)}$, to real matrices of dimension at most $D+1$, while preserving algebra operations and matrix transposes. Using this decomposition, we prove that the matrix $L_{\text{SS}}Q_{\text{SS}}L_{\text{SS}}^\top$ is "positive-definite", and hence so is Q_{SS}. Here, $L_{\text{SS}}, M_{\text{SS}}$ are the simple spider part of L, M respectively. The proof of this fact somewhat surprisingly boils down to the multiplicative identities of Hermite polynomials. The quotation mark around "positive-definite" means that we obtain a sum-of-squares expression of $L_{\text{SS}}Q_{\text{SS}}L_{\text{SS}}^\top$ in the approximation algebra, SA_D. In reality, since the well-behaved product only approximates real matrix products of certain simple spiders but not all, to extend the "positive-definiteness in SA_D" to the positive-definiteness of the matrix Q_{SS}, we need to make sure that Q_{SS} contains only special simple spiders where this approximation works well.

The second step is to extend the positive-definiteness to Q_{SSD}. Recall that it is the dominant part of Q and is a linear combination of simple spider disjoint unions (SSD). This time, we do not have to model the multiplication of SSD shapes algebraically (as we did for simple spiders); instead, given the sum-of-squares expression of Q_{SS} obtained from the above, we can directly construct a square root of Q_{SSD} by an operation we call the D-combination. The intuition is that given a linear combination α of simple spiders, its D-combination linearly combines all possible disjoint union of shapes in α with their coefficients multiplied together.[3] This construction is combinatorial rather than algebraic, but it turns out to have a useful algebraic property: D-combination commutes with well-behaved products in a sense. The formal statement is given in the full paper. Using this property, we prove that if $X \cdot X^\top \approx Q_{\text{SS}}$ then $[X]^D \cdot ([X]^D)^\top \approx Q_{\text{SSD}}$, where $[X]^D$ means the D-combination of X. This helps us prove the positive-definiteness of Q_{SSD} . Again, additional analytic arguments are required in the actual proof. We also need to show that $[X]^D$ is not too close to being singular so that we can use $[X]^D \cdot ([X]^D)^\top$ to compensate for the error terms.

To summarize, from the error analysis we have $\|Q - Q_{\text{SSD}}\| \le n^{-\Omega(\varepsilon)}$. By the above steps, we show that $Q_{\text{SSD}} \succ n^{-o(\varepsilon)}\text{Id}$ assuming that A satisfies some mild conditions besides matching $(k-1)$ moments. Together, we get the positive-definiteness of Q. From here, it is not hard to show that $M \approx LQL^\top$ is PSD. We now turn to handling the error terms.

Handling error terms via configurations. As described more precisely in the proof overview in the full version, there are two main sources of error terms:

1) The error $Q - Q_{\text{SSD}}$ in the approximation of Q by the sum of well-behaved configurations.
2) The truncation error $M - LQL^\top$.

We also need to analyze the error in our PSD approximation $Q_{\text{SSD}} \approx [X]^D \cdot ([X]^D)^\top$ of Q_{SSD}.

Our framework for handling these error terms is as follows. We formalize the way an error term can be generated as a *configuration*, whose definition is given in the full paper. The goal is then to show that the following number is small: the product of the coefficients from all shapes in the configuration, multiplied with the norm of any graph matrix that results from the configuration. To estimate this number, we use a charging argument that assigns edge factors to vertices. The idea is that to calculate, for example, the exponent over n in the expression $n^{-|E_\alpha|/2}$ times the norm bound on a graph matrix M_α, we take $\log_{\sqrt{n}}(\cdot)$ of the expression. We imagine that each edge in shape α has an additive factor of 1, and we assign the edge factors to its endpoints so that each vertex receives a sufficient amount of factors. The main result we prove is a dichotomy. Either the configuration is a well-behaved SSD product and has approximate norm 1 or the configuration has norm $o(1)$. This allows us to show that the errors $Q - Q_{\text{SSD}}$ and $Q_{\text{SSD}} - [X]^D \cdot ([X]^D)^\top$ have norm $n^{-\Omega(\varepsilon)}$. The design and analysis of the edge factors assignment scheme relies on properties of the minimum square separators and the minimum weight separators, which might be of independent interest.

To handle the truncation error, we observe that the truncation error only contains configurations which are very large and all such configurations have norm $n^{-\Omega(\varepsilon D_{trunc})}$, where D_{trunc} is a "total size" threshold on shapes that we set in pseudo-calibration.

II. Formal Statement of the Main Result

In this section, we formally define the problem statement.and state our main result. In Section II-A, we set

[3]Technically, we require α to satisfy a certain consistency condition which we state precisely in the full version of the paper.

up basic notation. In Section II-B, we recall the notions of pseudo-expectation values and the moment matrix, formally define the NGCA problem, and formulate it in the sum-of-squares framework. In Section II-C, we recall the technique of pseudo-calibration introduced by [BHK+16] and identify the pseudo-calibrated moment matrix whose PSDness we want to prove. In Section II-D we show that our pseudo-expectation values satisfy the desired constraints except for PSDness of the moment matrix.

A. Notation

Basic Notation: $\mathbb{R}$ is the set of real numbers. For $t \in \mathbb{Z}_+$, $[t] := \{1, \ldots, t\}$. We will use n to denote the dimension of the input data, and m for the number of samples. The given m samples are denoted by $x_1, \ldots, x_m$, where each $x_u = (x_{u1}, \ldots, x_{un}) \in \mathbb{R}^n$. We will use index symbols $u \in [m]$, $i \in [n]$. The SoS degree is D. For an integer vector $a \in \mathbb{N}^n$, $\|a\|_1 = \sum_{i=1}^{n} a(i)$, $a! := \prod_{i=1}^{n} a(i)!$. For a sequence of m such vectors, accordingly, $a = (a_1, \ldots, a_m) \in (\mathbb{N}^n)^m$, $\|a\|_1 = \sum_u \|a_u\|_1 = \sum_{u,i} a_u(i)$ and $a! := \prod_u a_u! = \prod_{u,i} a_u(i)!$. It might be helpful to think of $a \in (\mathbb{N}^n)^m$ as an edge-weighted and undirected bipartite graph on vertex sets $[n], [m]$ where $a_u(i)$ is the weight of edge $\{i, u\}$. For matrices M, N and a number $C > 0$, we use $M \overset{C}{\approx} N$ to denote that $\|M - N\| \leq C$ where $\|\cdot\|$ on matrices always means the operator norm. If M, N are square matrices, $M \succcurlyeq N$ denotes that $M - N$ is positive-semidefinite (PSD). We use $\mathrm{poly}(\cdot)$ to indicate a quantity that is polynomially upper-bounded in its arguments. Similarly, $\mathrm{polylog}(\cdot)$ denotes a quantity that is polynomially upper-bounded in the logarithm of its arguments. By $\log(\cdot)$ we mean $\log_2(\cdot)$.

Probability Notation: For a random variable X, we write $\mathrm{E}[X]$ for its expectation. $\mathcal{N}(\mu, \sigma^2)$ denotes the 1-dimensional Gaussian distribution with mean μ and variance σ^2. When $\mathcal{D}$ is a distribution, we use $X \sim \mathcal{D}$ to denote that the random variable X is distributed according to $\mathcal{D}$. When S is a set, we let $\mathrm{E}_{X \sim S}[\cdot]$ denote the expectation under the uniform distribution over S.

Hermite Polynomials: The probabilist's Hermite polynomial $He_i(x)$ will be denoted by $h_i(x)$. The n-dimensional Hermite polynomials are $h_a = \prod_{i=1}^{n} h_{a(i)}$ for $a \in \mathbb{N}^n$. Recall that $h_a/\sqrt{a!}$ ($a \in \mathbb{N}^n$) form an orthonormal basis of polynomials under the inner product

$$\langle f, g \rangle := \mathop{\mathrm{E}}_{x \sim \mathcal{N}(0, \mathrm{Id}_n)} \big[f(x) g(x) \big]$$

where $\mathcal{N}(0, \mathrm{Id}_n)$ is the n-dimensional multivariate normal distribution with mean 0 and covariance matrix Id_n. For $a \in (\mathbb{N}^n)^m$, we let $h_a = \prod_{u=1}^{m} h_{a_u}$.

B. Problem Statement and Sum-of-Squares Solutions

We now define the SoS formulation for NGCA that we use. The inputs to our SoS program are i.i.d. samples $x_1, \ldots, x_m$ drawn from the reference distribution. We denote the SoS variables by $\mathrm{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_n)$.

A true solution to the NGCA problem would assign a real value to each v_i such that the following constraints are satisfied:

1) (Booleanity) For all $i \in [n]$, $\mathrm{v}_i^2 - \frac{1}{n} = 0$.
2) (Soft NGCA constraints)

$$\big| (1/m) \sum_{i=1}^{m} (h_j(x \cdot \mathrm{v})) - \mathop{\mathrm{E}}_{a \sim A} [h_j(a)] \big| = \widetilde{O}(1/\sqrt{m}) \,,$$

where $x_1, \ldots, x_m$ are the input samples.

Degree-D SoS gives a relaxation of the problem where instead of assigning a real value to each v_i, we have an $\mathbb{R}$-linear map $\widetilde{E}$ called *pseudo-expectation values* which assigns a real value $\widetilde{E}[p]$ to each polynomial $p(\mathrm{v}_1, \ldots, \mathrm{v}_n)$ of degree at most D. We can think of $\widetilde{E}[p]$ as the estimate given by degree-D SoS for the expected value of p over a (possibly fictitious) distribution of solutions.

Definition II.1. *We define $\mathbb{R}^{\leq d}(\mathrm{v})$ to be the set of all polynomials of degree at most d in the variables $\mathrm{v}_1, \ldots, \mathrm{v}_n$.*

Definition II.2 (Pseudo-expectation Operator for NGCA)**.** *Given input samples $x_1, \ldots, x_m$ and a target distribution A, we say that an $\mathbb{R}$-linear map $\widetilde{E} : \mathbb{R}^{\leq D}(\mathrm{v}_1, \ldots, \mathrm{v}_n) \to \mathbb{R}$ is a degree D pseudo-expectation operator for NGCA if it satisfies the following conditions.*

1) *(Oneness) $\widetilde{E}(1) = 1$;*
2) *(Booleanity) $\widetilde{E}\Big(f(\mathrm{v}) \cdot (\mathrm{v}_i^2 - \frac{1}{n})\Big) = 0$ for all $i \in [n]$ and all $f \in \mathbb{R}^{\leq D-2}(\mathrm{v})$;*
3) *(Soft NGCA constraints) $\Big| \frac{1}{m} \sum_{u=1}^{m} \widetilde{E}\,(h_j(x_u \cdot \mathrm{v})) - \mathop{\mathrm{E}}_{a \sim A} [h_j(a)] \Big| = \widetilde{O}(\frac{1}{\sqrt{m}})$ for all $j \leq D$;*
4) *(Positivity) $\widetilde{E}[p^2] \geq 0$ for all $p \in \mathbb{R}^{\leq D}(\mathrm{v})$.*

If this relaxation is infeasible then degree-D SoS can prove that there is no vector $\mathrm{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_n)$ such that the input has distribution A in the direction v. For our degree-D SoS lower bounds, we show that w.h.p. (with high probability) this does not happen. To do this, we design a candidate pseudo-expectation operator $\widetilde{E}$ and show that w.h.p. it is a degree D pseudo-expectation operator for NGCA.

Remark II.3. We use Roman text font for the SoS variables $\mathrm{v}_1, \ldots, \mathrm{v}_n$ to distinguish them other expressions such as the input variables which take fixed real values once the input is given.

Remark II.4 (Soft NGCA constraints)**.** The "soft" NGCA constraints (Item 3) indicate that we study generic SoS lower bounds, i.e., there is no strict polynomial identity constraint other than booleanity of the variables v. This is more or less an inevitable feature of the algorithms dealing with NGCA for general distributions A in Definition I.1, as opposed to cases where A is more restricted such as being discrete.

The positivity condition on $\widetilde{E}$ can be expressed using the following (pseudo-)moment matrix.

Definition II.5. *We define* v^I *to be the monomial* $\prod_{i=1}^{n} \mathrm{v}_i^{I(i)}$ *where* $I \in \mathbb{N}^n$*. We define the degree of* v^I *to be* $\|I\|_1 = \sum_i I(i)$.

Definition II.6 (Pseudo-moment Matrix)**.** *Given a linear map* $\widetilde{E} : \mathbb{R}^{\leq 2D}[\mathrm{v}] \to \mathbb{R}$*, its degree* D *pseudo-moment matrix, or moment matrix in short, is an* $\binom{[n]}{\leq D} \times \binom{[n]}{\leq D}$ *matrix* $M_{\widetilde{E}}$ *whose rows and columns indexed by subsects of* $I \subseteq [n]$ *of size at most* D*, and the entries are* $M_{\widetilde{E}}(I, J) := \widetilde{E}(\mathrm{v}^{I+J})$*, where* I, J *are viewed as the indicator functions from* $[n]$ *to* $\{0, 1\}$.

The verification of the following fact is straightforward.

Fact II.7. *Suppose* $M_{\widetilde{E}}$ *satisfies the Booleanity condition. Then the positivity condition, Item 4 in Definition II.2, is equivalent to the condition that* $M_{\widetilde{E}} \succcurlyeq 0$.

In the next section, we describe the standard pseudo-calibration technique used to prove SoS lower bounds.

C. Pseudo-calibration Technique for NGCA and our Main Result

Pseudo-calibration, introduced in [BHK+16], is a method to construct a candidate pseudo-expectation operator $\widetilde{E}$ for an input x drawn from the problem distribution (true distribution, "tr"). The idea is to show that there is another distribution (planted distribution, "pl") supported on feasible instances and solutions (x, v), such that it cannot be distinguished from the problem distribution via any low-degree test. Once we have this planted distribution, we will choose the candidate pseudo-expectation values $\widetilde{E}(\mathrm{v}^I)$ so that

$$\mathop{E}_{x \sim \mathcal{D}_{ref}} \left[t(x) \widetilde{E}(\mathrm{v}^I) \right] = \mathop{E}_{(x,v) \sim \mathcal{D}_{pl}} \left[t(x) v^I \right] \tag{1}$$

for all low-degree polynomials $t(x)$, where v^I is the monomial $\prod_{i=1}^{n} v_i^{I(i)}$, $I \in \mathbb{N}^n$. Moreover, we impose the condition:

Each $\widetilde{E}(\mathrm{v}^I)$ itself is a low-degree polynomial in x. (2)

For our problem, the input data $(x_1, \ldots, x_m) \in (\mathbb{R}^n)^m$ are i.i.d. samples drawn from the true Gaussian distribution, , and $\mathrm{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_n)$ are the SoS variables representing the unknown direction whose solution existence SoS wants to refute. In light of the NGCA problem in Problem I.1, our planted distribution D_{pl} is the following: first choose a planted vector $v \sim \{\pm \frac{1}{\sqrt{n}}\}^n$ uniformly at random, then choose i.i.d. samples $x_1, \ldots, x_m$ where

$$x_i = \big((x_i)_{v^\perp},\ (x_i)_v\big) \sim \mathcal{N}(0, \mathrm{Id}_{n-1})_{v^\perp} \times A_v,$$

where A_v is the one-dimensional distribution of interest in direction v matching $k-1$ moments with $\mathcal{N}(0,1)$. Concretely, conditions Equation (1) and Equation (2) enforce the pseudo-calibration to have the following form:

$$\widetilde{E}(\mathrm{v}^I) := \sum_{\substack{a \in (\mathbb{N}^n)^m : \text{"total size" of } a \\ \text{is upper bounded by } D_{trunc}}} \mathop{E}_{(x,v) \sim \mathcal{D}_{pl}} \left[v^I \frac{h_a}{\sqrt{a!}} \right] \cdot \frac{h_a}{\sqrt{a!}}, \tag{3}$$

where D_{trunc} is a parameter deciding the meaning of "low-degree" in Equation (1) and Equation (2). We will choose D_{trunc} based on the technical analysis. It turns out that we can choose D_{trunc} to be any value between $C_1 \log n$ and n^{C_2} for some constants C_1, C_2 depending on ε, although there will be a trade-off between D and D_{trunc}.

For any fixed $I \in \mathbb{N}^n$, we let the *I-total size* of $a = (a_1, \ldots, a_u) \in (\mathbb{N}^n)^m$ be $\mathrm{total}^I(a) :=$

$$\|a\|_1 + |\{i \in [n] :\ I(i) > 0 \text{ or } (\exists u \in [m])\ a_u(i) > 0\}| + |\{u \in [m] :\ (\exists i \in [n])\ a_u(i) > 0\}|.$$

We use this to measure the "total size" of a in the above equation. The calculation of (3) is similar to the one in [GJJ+20], giving the following expression.

Lemma II.8 (Pseudo-calibration)**.** *For any* $I \in \mathbb{N}^n$*,* $\widetilde{E}(\mathrm{v}^I)$ *is given by,*

$$\sum_{\substack{a \in (\mathbb{N}^n)^m :\ \mathrm{total}^I(a) \leq D_{trunc}, \\ \text{and } (\forall i \in [n])\ I(i) + \sum_u a_u(i) \text{ is even}}} n^{-\frac{\|I\|_1 + \|a\|_1}{2}} \prod_{u=1}^{m} \mathop{\mathrm{E}}_{A} \left[h_{|a_u|} \right] \frac{h_{a_u}}{a_u!} \tag{4}$$

Remark II.9 (Only moments matter)**.** By Equation (4), the pseudo-expectation values are determined by the moments of A up to the truncation threshold D_{trunc}. In particular, if D_{trunc} is smaller than the number of matched moments (i.e., $k-1$), then Equation (4) will

be a constant function and the resulting matrix will be diagonal and trivially PSD.

Definition II.10 (C_U, C_L)**.** *Given n, D, D_{trunc} and distribution A, we let C_U, C_L be the minimum values such that $C_U, C_L \geq 1$, and*

1) *For all $t \leq 3D_{trunc}$, $\left|\mathop{\mathrm{E}}_{A}[h_t(x)]\right| \leq C_U^t$.*
2) *For all polynomials $p(x)$ of degree at most D such that $\int_{\mathcal{N}(0,1)} p^2(x) = 1$, $\mathop{\mathrm{E}}_{A}[p^2(x)] \geq C_L^{-\deg(p)}$.*

We can now state the main theorem formally.

Theorem II.11 (Main Theorem)**.** *There is a universal constant $C_{\text{univ}} \geq 1$ such that for any $\delta \in (0,1)$, if n is sufficiently large then the following holds. Suppose $\varepsilon \in (0,1)$, A is a 1-dimensional distribution, $k \geq 2$, and D and D_{trunc} are integer parameters (where ε, A, k, D, and D_{trunc} may all depend on n) such that:*

$$A \text{ matches the first } k-1 \text{ moments with } \mathcal{N}(0,1). \quad (5)$$

$$D_{trunc} \geq \max\{50D^2, \frac{500}{\varepsilon}D, 2k\log n\}, \text{ and} \quad (6)$$

$$(5C_U)^{20D^2} C_L^{2D} (10D_{trunc})^{256C_{\text{univ}}D^2} < n^{\varepsilon/30}. \quad (7)$$

Then if we draw $m < n^{(1-\varepsilon)k/2}$ i.i.d. samples from $\mathcal{N}(0, \mathrm{Id}_n)$, with probability greater than $1-\delta$, the degree-D pseudo-calibration moment matrix with truncation threshold D_{trunc} is positive-definite.

For example, we can set $k \leq (\log n)^{O(1)}$, $D = o(\sqrt{\frac{\varepsilon \log n}{\log\log n}})$, and $D_{trunc} = 2kD\log n$. Then if the distribution A matches $k-1$ moments with $\mathcal{N}(0,1)$ and satisfies $C_U, C_L \leq (\log n)^{O(1)}$, Theorem II.11 provides an almost optimal $n^{\frac{(1-\varepsilon)k}{2}}$ sample lower bound for the corresponding NGCA problem in degree-D SoS.

The proof of Theorem II.11 is in the full version of the paper.

D. Properties of Pseudo-calibration

In addition to Theorem II.11, we show that the pseudo-calibration "solution" of v from Equation (4) satisfies the Booleanity constraints and the soft NGCA constraints.

It is not hard to check directly that $\widetilde{E}$ satisfies the Booleanity constraints as multiplying v^I by v_i^2 increases the number $I(i)$ by 2. This is also a special case of the following more general fact (Cf. [BHK+16], [GJJ+20]), whose proof is a simple expansion of the definition Equation (3).

Lemma II.12 (Pseudo-expectation preserves zero)**.** *If $f(\mathrm{v})$ is a polynomial only in the SoS variables v such that $\deg(f) \leq D$ and $f(v) = 0$ for all v in the planted distribution, then $\widetilde{E}(f) = 0$ independent of the input x.*

As for the soft NGCA constraints, we will show that for any low-degree Hermite polynomial evaluated on the inner product $x \cdot v$, with high probability the pseudo-expectation value is close to the expectation under A (Lemma II.13). Below, recall that $h_j(x)$ denotes $He_j(x)$.

Lemma II.13 (Hermite Tests in the Hidden Direction)**.** *For all Hermite polynomials h_j of degree at most D, with high probability, $\left|\frac{1}{m}\sum_{i=1}^{m}\widetilde{E}\left(h_j(x\cdot \mathrm{v})\right) - \mathop{\mathrm{E}}_{a\sim A}[h_j(a)]\right|$ is $o(1)$. More precisely,*

$$\left|\frac{1}{m}\sum_{i=1}^{m}\widetilde{E}\Big(h_j(x_i\cdot \mathrm{v})\Big) - \mathop{\mathrm{E}}_{a\sim A}[h_j(a)]\cdot\widetilde{E}(1)\right| \leq \frac{\mathrm{polylog}(m)}{\sqrt{m}}.$$

The proof of this lemma is in the full version of the paper.

References

[AMP16] K. Ahn, D. Medarametla, and A. Potechin. Graph matrices: norm bounds and applications. *arXiv preprint arXiv:1604.03423*, 2016.

[BBH+21] M. Brennan, G. Bresler, S. B. Hopkins, J. Li, and T. Schramm. Statistical query algorithms and low-degree tests are almost equivalent. *34th Annual Conference on Learning Theory (COLT)*, 2021.

[BHK+16] B. Barak, S. B. Hopkins, J. A. Kelner, P. Kothari, A. Moitra, and A. Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *FOCS*, 2016.

[BKS+06] G. Blanchard, M. Kawanabe, M. Sugiyama, V. Spokoiny, and K.-R. Müller. In search of non-gaussian components of a high-dimensional distribution. *Journal of Machine Learning Research*, 2006.

[BRST21] J. Bruna, O. Regev, M. J. Song, and Y. Tang. Continuous lwe. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 2021.

[CLL22] S. Chen, J. Li, and Y. Li. Learning (very) simple generative models is hard. In *NeurIPS*, 2022.

[CV23] X. Cao and S. Vempala. Contrastive moments: Unsupervised halfspace learning in polynomial time. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[DH24] R. Dudeja and D. Hsu. Statistical-computational trade-offs in tensor pca and related problems via communication complexity. *The Annals of Statistics*, 2024.

[DJNS13] E. Diederichs, A. Juditsky, A. Nemirovski, and V. Spokoiny. Sparse non gaussian component analysis by semidefinite programming. *Machine learning*, 2013.

[DJSS10] E. Diederichs, A. Juditsky, V. Spokoiny, and C. Schutte. Sparse non-gaussian component analysis. *IEEE Transactions on Information Theory*, 2010.

[DK22a] I. Diakonikolas and D. Kane. Near-optimal statistical query hardness of learning halfspaces with massart noise. In *Conference on Learning Theory (COLT)*, 2022.

[DK22b] I. Diakonikolas and D. Kane. Non-gaussian component analysis via lattice basis reduction. In *Conference on Learning Theory (COLT)*, 2022.

[DK23] I. Diakonikolas and D. M. Kane. *Algorithmic High-Dimensional Robust Statistics*. Cambridge university press, 2023. Full version available at https://sites.google.com/view/ars-book.

[DKK+22] I. Diakonikolas, D. M. Kane, V. Kontonis, C. Tzamos, and N. Zarifis. Learning general halfspaces with general massart noise under the gaussian distribution. In *54th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, 2022.

[DKKZ20] I. Diakonikolas, D. M. Kane, V. Kontonis, and N. Zarifis. Algorithms and SQ lower bounds for PAC learning one-hidden-layer relu networks. In *Conference on Learning Theory, (COLT)*, 2020.

[DKP+21] I. Diakonikolas, D. M. Kane, A. Pensia, T. Pittas, and A. Stewart. Statistical query lower bounds for list-decodable linear regression. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[DKPZ21] I. Diakonikolas, D. M. Kane, T. Pittas, and N. Zarifis. The optimality of polynomial regression for agnostic learning under gaussian marginals in the SQ model. In *Conference on Learning Theory, COLT*, 2021.

[DKPZ23] I. Diakonikolas, D. M. Kane, T. Pittas, and N. Zarifis. SQ lower bounds for learning mixtures of separated and bounded covariance gaussians. In *The Thirty Sixth Annual Conference on Learning Theory, COLT*, 2023.

[DKRS23] I. Diakonikolas, D. Kane, L. Ren, and Y. Sun. Sq lower bounds for non-gaussian component analysis with weaker assumptions. *Advances in Neural Information Processing Systems*, 2023.

[DKS17] I. Diakonikolas, D. M. Kane, and A. Stewart. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In *IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, 2017.

[DKS18] I. Diakonikolas, D. M. Kane, and A. Stewart. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC*, 2018.

[DKS19] I. Diakonikolas, W. Kong, and A. Stewart. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2019.

[DKS23] I. Diakonikolas, D. M. Kane, and Y. Sun. SQ lower bounds for learning mixtures of linear classifiers. *CoRR*, abs/2310.11876, 2023.

[DKZ20] I. Diakonikolas, D. M. Kane, and N. Zarifis. Near-optimal SQ lower bounds for agnostically learning halfspaces and relus under gaussian marginals. *CoRR*, abs/2006.16200, 2020.

[FGR+17] V. Feldman, E. Grigorescu, L. Reyzin, S. Vempala, and Y. Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *J. ACM*, 2017.

[GGJ+20] S. Goel, A. Gollakota, Z. Jin, S. Karmalkar, and A. R. Klivans. Superpolynomial lower bounds for learning one-layer neural networks using gradient descent. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, 2020.

[GGK20] S. Goel, A. Gollakota, and A. R. Klivans. Statistical-query lower bounds via functional gradients. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[GJJ+20] M. Ghosh, F. G. Jeronimo, C. Jones, A. Potechin, and G. Rajendran. Sum-of-squares lower bounds for sherrington-kirkpatrick via planted affine planes. In *IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, 2020.

[GS19] N. Goyal and A. Shetty. Non-gaussian component analysis using entropy methods. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC*, 2019.

[GVV22] A. Gupte, N. Vafa, and V. Vaikuntanathan. Continuous lwe is as hard as lwe & applications to learning gaussian mixtures. In *IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, 2022.

[Hop18] S. B. K. Hopkins. Statistical inference and the sum of squares method. 2018.

[JPR+21] C. Jones, A. Potechin, G. Rajendran, M. Tulsiani, and J. Xu. Sum-of-squares lower bounds for sparse independent set. In *IEEE 62nd Annual Symposium on Foundations of Computer Science*, FOCS, 2021.

[JPRX23] C. Jones, A. Potechin, G. Rajendran, and J. Xu. Sum-of-squares lower bounds for densest k-subgraph. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC, 2023.

[Kea98] M. J. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 1998.

[MW22] C. Mao and A. S. Wein. Optimal spectral recovery of a planted vector in a subspace. *CoRR*, 2022.

[Pan21] S. Pang. Sos lower bound for exact planted clique. In *36th Computational Complexity Conference (CCC)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

[PR20] A. Potechin and G. Rajendran. Machinery for proving sum-of-squares lower bounds on certification problems. *arXiv preprint arXiv:2011.04253*, 2020.

[SKBM08] M. Sugiyama, M. Kawanabe, G. Blanchard, and K.-R. Muller. Approximating the best linear unbiased estimator of non-gaussian signals with gaussian noise. *IEICE transactions on information and systems*, 2008.

[SNS16] H. Sasaki, G. Niu, and M. Sugiyama. Non-gaussian component analysis with log-density gradient estimation. In *Artificial Intelligence and Statistics*. PMLR, 2016.

[Tie24] S. Tiegel. Improved hardness results for learning intersections of halfspaces, 2024.

[TKM11] F. J. Theis, M. Kawanabe, and K.-R. Muller. Uniqueness of non-gaussianity-based dimension reduction. *IEEE Transactions on signal processing*, 2011.

[TV18] Y. S. Tan and R. Vershynin. Polynomial time and sample complexity for non-gaussian component analysis: Spectral methods. In *Conference On Learning Theory, (COLT) 2018*, 2018.

[Val84] L. Valiant. A theory of the learnable. *Communications of the ACM*, 1984.

[VNO16] J. Virta, K. Nordhausen, and H. Oja. Projection pursuit for non-gaussian independent components. *arXiv preprint arXiv:1612.05445*, 2016.

[ZSWB22] I. Zadik, M. J. Song, A. S. Wein, and J. Bruna. Lattice-based methods surpass sum-of-squares in clustering. In *Conference on Learning Theory (COLT)*, 2022.