Intro to the Sum-of-Squares Hierarchy

Tselil Schramm

This note is intended to introduce the Sum-of-Squares Hierarchy. We start by SDP relaxation, using the Goemans-Williamson Max-Cut SDP as a jumping off point. We then discuss duality and sum-of-squares proofs. Finally, we give an example non-trivial application of the sum-of-squares hierarchy: an algorithm for finding planted sparse vectors inside random subspaces. We will give no historical context, but in the final section there will be pointers to other resources which give a better sense of the history and alternative expositions of the same content.

1 A Relaxation for Polynomial Optimization

Suppose we are interested in some polynomial optimization problem:

$$Q = \left\{ \max_{x \in \mathbb{R}^n} \; p(x), \qquad s.t. \quad g_i(x) = 0 \quad \forall \; i \in [m] \right\}.$$

That is, we want to maximize our objective function, the polynomial $p : \mathbb{R}^n \to \mathbb{R}$, subject to the polynomial constraints $g_1(x) = 0, g_2(x) = 0, \ldots, g_m(x) = 0$.

The problem Q may be non-convex, and solving such programs is NP-complete (i.e. this captures integer programming). A standard approach for a situation like this is to relax our problem Q to a semidefinite program (SDP). Perhaps the most famous example is the Goemans-Williamson relaxation for Max-Cut:

Example 1 (Max-Cut)**.** We can formulate the max cut problem on an n-vertex graph G as a polynomial optimization problem with objective function $p(x) = \sum_{(i,j) \in E(G)} \frac{1 - x_i x_j}{2}$ and the constraint polynomials $g_i(x) = x_i^2 - 1 = 0 \; \forall i \in [n]$, which ensure that each $x_i = \pm 1$.

The Goemans-Williamson SDP relaxation assigns program variables X_{ij} for each $i, j \in [n]$, where X_{ij} is a stand-in for the monomial $x_i x_j$. The SDP then becomes

$$\left\{ \max \sum_{(i,j) \in E(G)} \tfrac{1}{2}(1 - X_{ij}), \qquad s.t. \quad X_{ii} - 1 = 0 \quad \forall i \in [n], \quad X \succeq 0 \right\}$$

where X is the $n \times n$ matrix with variable X_{ij} in the (i, j)th entry.

The Sum-of-Squares SDP: Extending Goemans-Williamson. After seeing the Goemans-Williamson Max-Cut SDP, it seems natural to apply a similar relaxation to other polynomial optimization problems. Suppose that the maximum degree of any term in $p, g_1, \ldots, g_m$ is at most $2d$. The strategy is to relax the polynomial optimization problem by replacing each monomial $\prod_{i \in S \subset [n]} x_i$ which appears in the program Q with an SDP variable X_S. So for each $S \subset [n]$, $|S| \leq 2d$, we have an SDP variable. Then we arrange the variables into an $(n+1)^d \times (n+1)^d$ matrix X in the natural way, with rows and columns indexed by every ordered subset of at

most d variables:

$$X = \begin{array}{c} \\ \emptyset \\ \{1\} \\ \{2\} \\ \vdots \\ S \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccccc} \emptyset & \{1\} & \{2\} & \cdots & T & \cdots \end{array} \\ \begin{bmatrix} X_\emptyset & X_{\{1\}} & X_{\{2\}} & \cdots & X_T & \cdots \\ X_1 & X_{\{1,1\}} & X_{\{1,2\}} & \cdots & X_{\{1\}\cup T} & \cdots \\ X_2 & X_{\{2,1\}} & X_{\{2,2\}} & \cdots & X_{\{2\}\cup T} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ X_S & X_{S\cup\{1\}} & X_{S\cup\{2\}} & \cdots & X_{S\cup T} & \\ \vdots & \vdots & \vdots & & & \ddots \end{bmatrix} \end{array}$$

Now we enforce some natural constraints:

- "Commutativity" or "symmetry": If the ordered multisets $S, T, U, V \subset [n]$ are such that $S \cup T = U \cup V$ as unordered multisets, then $X_{S\cup T} = X_{U\cup V}$. This is meant to reflect the commutative property of monomials. That is, for any $x \in \mathbb{R}^n$

$$\prod_{i\in S} x_i \cdot \prod_{j\in T} x_j = \prod_{k\in U} x_k \cdot \prod_{\ell\in V} x_\ell.$$

- "Normalization": we set $X_\emptyset = 1$. This is the "scale" of the coefficients. One way to see that this is the correct scale is to think of $X_\emptyset$ as the monomial multiplier of a polynomial's constant term.

- "PSDness": we require that $X \succeq 0$, or that X is positive-semidefinite. This constraint is natural because for any point $y \in \mathbb{R}^n$, if we take the matrix $X = X_y$ given by setting $X_S = \prod_{i\in S} y_i$, the resulting matrix X is PSD. The proof is that if we take the vector $\tilde{y}$ so that $\tilde{y}^\top = [1\ y^\top]$, then $X_y = \tilde{y}^{\otimes d}(\tilde{y}^{\otimes d})^\top$ (where $y^{\otimes d}$ is the dth Kronecker power[1] of y), and thus for any vector v, $v^\top X_y v = \langle v, \tilde{y}^{\otimes d}\rangle^2 \geq 0$.

Remark 1. Notice that any feasible solution $y \in \mathbb{R}^n$ for the program Q yields a feasible solution to $\mathrm{sos}_d(Q)$: we just assign $X_S := \prod_{i\in S} y_i$, and the above arguments show that this is feasible.

One nice consequence of these constraints is that, if we evaluate the square of some degree-d polynomial q in the SDP monomials, the polynomial value will be non-negative! This is because, if $\hat{q}$ is the vector of coefficients of the polynomial q, then $q^2(X) = \hat{q}^\top X \hat{q} \geq 0$.

Example 2. Consider the square polynomial $(x_1 + c \cdot x_2)^2$. We would evaluate this square in X by taking the quadratic form

$$\begin{bmatrix} 0 & 1 & c \end{bmatrix} \begin{bmatrix} X_\emptyset & X_{\{1\}} & X_{\{2\}} \\ X_1 & X_{\{1,1\}} & X_{\{1,2\}} \\ X_2 & X_{\{2,1\}} & X_{\{2,2\}} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ c \end{bmatrix} = X_{1,1} + c \cdot X_{1,2} + c \cdot X_{2,1} + c^2 \cdot X_{2,2}.$$

We now formalize the above definition.

Definition 1 (Sum-of-Squares Relaxation for Q at degree $2d$)**.** Given a polynomial optimization problem Q with $\deg(p) \leq 2d$ and $\deg(g_i) \leq 2d\ \forall i \in [m]$, we define the *degree-$2d$ sum-of-squares relaxation* for Q, $\mathrm{sos}_d(Q)$.

[1]The Kronecker product of an $n \times m$ matrix A and a $\ell \times k$ matrix B is a $n\ell \times mk$ matrix $A \otimes B$, which we can naturally index by pairs so that the $(a, b), (c, d)$th entry is the product of $A_{ac}B_{bd}$. So, the Kronecker product of an $n \times 1$ vector x with itself is a $n^2 \times 1$ vector whose (i, j)th entry is simply the product $x_i x_j$.

We define a variable X_S for each unordered multiset $S \subset [n]$ of size $|S| \leq 2d$, and define the $(n+1)^d \times (n+1)^d$ matrix X, with rows and columns indexed by ordered multisets $U, V \subset [n]$, so that the U, Vth entry of X contains the variable $X_{U \cup V}$. Define the operator $\tilde{E} : \text{monomials}_{\leq 2d} \to \mathbb{R}$ such that $\tilde{E}[\prod_{i \in S} x_i] = X_S$ for $|S| \leq 2d$. Then,

$$\text{sos}_d(Q) = \left\{ \begin{array}{ll} \max \ \tilde{E}[p(x)] \quad s.t. & X \succeq 0, \\ & X_\emptyset = 1, \\ & \tilde{E}\left[g_i(X) \cdot \prod_{i \in U} x_i\right] = 0 \quad \forall i \in [m], U \subset [n], \deg(g_i) + |U| \leq 2d \end{array} \right\}$$

Sum-of-Squares Hierarchy. Earlier, we only mentioned that we must have $2d \geq \deg(p), \deg(g_i) \forall i \in [m]$. In fact, we can choose d to be as large as we wish–as long as we are willing to solve an SDP with $n^{O(d)}$ variables and $n^{O(d)}$ constraints. Taking successively larger values for d gives us a systematic way of adding constraints to our program, giving us a family of larger but more powerful programs as we increase the value of d; this is why we call the family of relaxations $\{\text{sos}_d\}_{d=1}^{\infty}$ the *sum-of-squares hierarchy.*

How to make sense of the Sum-of-Squares SDP relaxation? In the Goemans-Williamson SDP relaxation, there is a natural interpretation of the SDP as a vector program: if we view the positive semidefinite matrix solution to the SDP, X, according to its Cholesky decomposition $X = VV^\top$, then we can identify each node in the underlying graph G with a unit vector corresponding to a row of the matrix V, and we can see that the objective function tries to push vectors corresponding to adjacent nodes apart on the unit sphere.

This geometric intuition is extremely crisp, but unfortunately it is hard to come up with an analogue of this in programs where we care about more than 2 variables interacting at a time (i.e. when we have variables X_S with $|S| \geq 3$). *As of now, we do not have a similar geometric understanding of general sum-of-squares SDP relaxations.* We can instead develop alternative ways to (partially) understand these SDP relaxations.

Pseudomoments. As a start, one perspective is to think of the variables X_S as the "moments" of a fake distribution over solutions to the program Q.

If we were to actually solve the (non-convex) problem Q (using some inefficient algorithm), what we would have is either a single solution $y^* \in \mathbb{R}^n$, or a distribution over some set of solutions $Y \subset \mathbb{R}^n$, which maximize p, so that

$$OPT(Q) = \mathbb{E}_{y \in Y}[p(y)].$$

We cannot expect that the solution to the relaxation $\text{sos}_d(Q)$ comes from an *actual* distribution over feasible solutions, but our constraints ensure that it still satisfies some of the properties of actual distributions[2]–for this reason we can also call the solution to $\text{sos}_d(Q)$ a *pseudodistribution*, and that is why we use the notation

$$\tilde{\mathbb{E}}\left[\prod_{i \in S} x_i\right] = X_S.$$

[2]Because of our constraints on $\text{sos}_d(Q)$, the pseudoexpectation $\tilde{E}$ satisfies linearity of expectation,

$$\tilde{E}[q(x)^2] \geq 0 \quad \text{if} \quad \deg(q) \leq d,$$

and also the non-negativity of low-degree squares,

$$\tilde{\mathbb{E}}[\alpha \cdot q_1(x) + \beta \cdot q_2(x)] = \alpha \cdot \tilde{\mathbb{E}}[q_1(x)] + \beta \cdot \tilde{\mathbb{E}}[q_2(x)] \quad \text{if} \quad \deg(q_1), \deg(q_2) \leq 2d.$$

In other words, we interpret the variable X_S as being the *pseudomoment* of the monomial $\prod_{i \in S} x_i$ under a *pseudodistribution* over solutions to Q.

Thinking about the SDP solution in this way can be helpful in designing algorithms (and in proving lower bounds), but I will not discuss this perspective further here (maybe in a future post).

2 Sum-of-Squares Proofs

One immediate question is, why should the sum-of-squares relaxation be a good relaxation? When we design SDP algorithms for maximization problems, we want to bound

$$OPT(Q) \leq OPT(\text{sos}_d(Q)) \leq \alpha \cdot OPT(Q),$$

for α as close to 1 as possible. Why should we expect α to be small?

We can give a concrete but somewhat technical answer to this question by considering the dual program: the dual program will give us a "sum-of-squares" proof of an upper bound on the primal program. In my opinion this is most easily explained via demonstration, so let's write down the dual program.

For convenience, we'll start by re-writing the primal program $\text{sos}_d(Q)$ in a matrix-based notation. For two matrices A, B of equal dimension, define the inner product $\langle A, B \rangle = \sum_{(i,j)} A_{ij} B_{ij}$. Now, define $(n+1)^d \times (n+1)^d$ matrices $P, G_1, \ldots, G_m$ so that $\langle P, X \rangle = \tilde{E}[p(x)]$ and $\langle G_i, X \rangle = \tilde{E}[g_i(x)]$ (where have redefined the polynomial constraints $g_1, \ldots, g_m$ to include most of our SDP constraints: symmetry/commutativity, and $g_i(x) \cdot X_U = 0$).

Example 3. If we have $p(x) = \sum_i x_i^2$, then one could choose the matrix P to contain the identity in the submatrix indexed by sets of cardinality 1, and 0 elsewhere.

Our program can now be written as the minimization problem,

$$\text{sos}'_d(Q) = \left\{ \min_{X \succeq 0} -\langle P, X \rangle \quad s.t. \quad \langle G_i, X \rangle = 0 \quad \forall i \in [m], \langle J_\emptyset, X \rangle = 1 \right\},$$

where $J_\emptyset$ is the matrix with a single 1 in the entry $\emptyset, \emptyset$ and zeros elsewhere, and the constraint $\langle J_\emptyset, X \rangle = 1$ enforces normalization. The optimal value of $\text{sos}_d(Q)'$ is the negation of the optimal value of $\text{sos}_d(Q)$.

The dual is the SDP problem

$$\text{sos}_d^+(Q) = \left\{ \max_{y \in \mathbb{R}^{m+1}} y_\emptyset \qquad s.t. \quad \left(-P - y_\emptyset \cdot J_\emptyset - \sum_{j \in [m]} y_j \cdot G_j \right) = S \succeq 0 \right\}.$$

Fixing y^* to be the optimal dual point, from the dual constraints we have that

$$P = -y^*_\emptyset \cdot J_\emptyset - S + \sum_j y^*_j \cdot G_j.$$

By duality, we have that in the optimal solution of $\text{sos}_d^+(Q)$,

$$y^*_\emptyset = c + OPT(\text{sos}'_d(Q) = c - OPT(\text{sos}_d(Q))$$

for some $c \geq 0$, and therefore taking $S' = S + c \cdot J_\emptyset \succeq 0$,

$$P = OPT \cdot J_\emptyset - S' + \sum_j y^*_j \cdot G_j.$$

We will turn this matrix equation into a polynomial equation. Let $x \in \mathbb{R}^n$, and let $\tilde{x} = [1\ x^\top]^\top$. Now, let S' have the Cholesky decomposition $S' = \sum uu^\top$. We take the quadratic form of the Kronecker power of $\tilde{x}$ with the left- and right-hand sides,

$$(\tilde{x}^{\otimes d})^\top P(\tilde{x}^{\otimes d}) = OPT - \sum \langle s, \tilde{x}^{\otimes d}\rangle^2 + \sum_{j\in[m]} y_j^* \cdot (\tilde{x}^{\otimes d})^\top G_j(\tilde{x}^{\otimes d})$$

and re-writing each of the above vector products as polynomials, where q_s is the polynomial encoded by the vector of coefficients s

$$p(x) = OPT - \sum q_s(x)^2 - \sum_{j\in[m]} y_j^* \cdot g_j(x).$$

This final line is a *sum-of-squares proof* that the value of $p(x)$ cannot exceed $OPT(\text{sos}_d(Q))$ on the feasible region: any feasible point $x \in \mathbb{R}^n$ evaluates to 0 for each g_i, and the square polynomials $q_s(x)^2$ can never contribute positively to the right-hand side. We have thus proven the following theorem:

Theorem 1. *The dual of the SDP* $\text{sos}_d(Q)$ *provides a degree-d sum-of-squares proof that* $p(x) \le OPT(\text{sos}_d(Q))$ *for all* x *in the feasible region of* Q*.*

At the start of this section, our goal was to understand how to bound

$$OPT(Q) \le OPT(\text{sos}_d(Q)) \le \alpha \cdot OPT(Q).$$

This theorem gives us a primal-dual tool for bounding the value of $\text{sos}_d(Q)$–if we can provide a sum-of-squares proof of degree at most d that $p(x) \le \alpha \cdot OPT(Q)$, then that sum-of-squares proof is a valid dual certificate!

Degree of the proof. Notice that the dual can only use polynomials of degree at most d in the sum-of-squares proof. So, suppose now that we write down two SDP relaxations for Q: $\text{sos}_d(Q)$ and $\text{sos}_{d'}(Q)$ for some $d' > d$. Then clearly,

$$OPT(\text{sos}_d(Q)) \ge OPT(\text{sos}_{d'}(Q)) \ge OPT(Q),$$

since the degree-d' sum-of-squares program contains more constraints than the degree-d program.

In the primal, it is difficult to understand exactly what these additional constraints buy you. From the perspective of the dual, the power of these additional constraints becomes clearer: the dual now has access to sum-of-squares proofs that use polynomials of *higher degree*, and this additional power may allow the dual to prove a potentially tighter upper bound. This is still a relatively mysterious condition, but in a later post I will give some concrete examples of situations in which it helps.

3 Planted Sparse Vector

In this section, we give one algorithmic application: the planted sparse vector problem.

Problem 1 (Planted k-sparse vector in a random d-dimensional subspace of $\mathbb{R}^n$)**.** *Given an* $n \times d$ *matrix* A*, distinguish between the following two cases:*

- *If the columns of* A *are uniformly sampled from a* d*-dimensional subspace of* $\mathbb{R}^n$ *which contains a vector with at most* $k < n/100$ *nonzero entries, return YES,*

- *If the columns of A are sampled from a uniformly random d-dimensional subspace of $\mathbb{R}^n$, return NO with high probability.*

This is a somewhat simple variant of the problem–other variants ask you to find the sparse vector as well. The exposition for this pared-down "distinguishing" version is simpler, and gets the main ideas across.

Without loss of generality, we may apply a random rotation $R \in \mathbb{R}^{d\times d}$ to the columns of A, then normalize by the maximum column norm, so that we work with $A \leftarrow \max_{i\in[d]} \frac{1}{\|ARe_i\|} AR$. This is to ensure that the columns of A have roughly the same norm, are roughly orthogonal, and all have norm roughly 1–for the remainder of the post we will assume that these conditions all hold.

We introduce the following polynomial optimization problem Q_{sparse} for the planted sparse vector problem:

$$Q_{sparse}(A) = \left\{ \max_{x\in R^d} \|Ax\|_4^4 \qquad s.t. \qquad \|x\|_2^2 = 1 \right\}$$

In other words, we want to find the linear combination of the columns of A that will maximize the 4-norm of A, while having 2-norm roughly 1. This program picks out sparse vectors over balanced vectors: a unit vector e_i with only one nonzero entry has $\|e_i\|_2^2 = \|e_i\|_4^4 = 1$, while a unit vector v with all n entries of the same magnitude has $\|v\|_4^4 = n \cdot (1/\sqrt{n})^4 = 1/n \ll \|v\|_2^2$.

We prove the following theorem:

Theorem 2. *If $1/k \geq \tilde{O}(\sqrt{d^3/n^3} + 1/n)$, then $\mathrm{sos}_4(Q_{sparse}(A))$ solves the planted k-sparse vector in a random subspace problem.*

We will prove this by showing that the value of the program is large in the planted case, and small in the random case. It is actually possible to prove a better tradeoff between k, d and n, but to simplify the arguments, we prove a weaker theorem. For the full details, see [Barak-Brandao-Harrow-Kelner-Steurer-Zhao '12].

Proof. If the span of the columns of A actually contains a k-sparse vector v^*, then $\|v^*\|_4^4$ is minimized when all entries of v^* have equal magnitude. So, if we normalize v^* so that $\|v^*\| = 1$,

$$\|v^*\|_4^4 \leq k \cdot \left(\frac{1}{\sqrt{k}}\right)^4 = \frac{1}{k}.$$

We will show that in the random case, the value is bounded by a function of n and d:

Lemma 1. *If A has iid Gaussian columns with $\mathbb{E}[A_{ij}^2] = \frac{1}{n}$, then with high probability the program $\mathrm{sos}_4(Q_{sparse}(A))$ has optimal value $\tilde{O}(\sqrt{d^3/n^3} + 1/n)$.*

Given this lemma, the proof of Theorem 2 is essentially trivial–we know that the objective value at most $\tilde{O}(\sqrt{d^3/n^3}+1/n)$ with high probability in the random case, and at least $1/k$ in the planted case, and so the objective value of $\mathrm{sos}_4(Q)$ distinguishes so long as $1/k \geq \tilde{O}(\sqrt{d^3/n^3} + 1/n)$. □

Now, we prove the lemma, using sum-of-squares proofs to bound the objective value of the SDP in the random case.

Proof of Lemma 1. For any $d^2 \times d^2$ matrix M, there is a sum-of-squares proof of the following fact:

$$\left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ M \right\rangle \leq \left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ \|M\| \cdot \mathrm{Id} \right\rangle.$$

The proof simply follows because $\|M\| \cdot \mathrm{Id} \succeq M$, and therefore $M = \|M\| \cdot \mathrm{Id} - S$ for some $S \succeq 0$; by taking the Cholesky decomposition of S and using the vectors as polynomial coefficients, this gives us a sum-of-squares proof of the inequality.

We will use this sum-of-squares fact to bound the SDP value of our objective function. First, we re-interpret our objective function as an inner product of two matrices. Let $a_1, \ldots, a_n$ be the rows of A. We will re-write our objective function as a matrix inner-product:

$$\|Ax\|_4^4 = \sum_i \langle a_i, x\rangle^4 = \left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ \sum_i (a_i \otimes a_i)(a_i \otimes a_i)^\top \right\rangle.$$

At this point, we could apply the above trick, but unfortunately, the maximum eigenvalue of $\sum_i (a_i \otimes a_i)(a_i \otimes a_i)^\top$ is $\approx d/n$–much larger than our goal of $\sqrt{d^3/n^3}$. This is because at indices $(\alpha\beta, \gamma\delta)$ where $\alpha = \beta$ and $\gamma = \delta$, our matrix has positive entries, whereas most entries have a random sign. These positive entries create a large eigenvalue in the matrix.

So, we will decompose this further–we will separate the portion of the matrix with entries corresponding to even-multiplicity indices. Define $B_{\neq}$ to be the matrix $\sum_i (a_i \otimes a_i)(a_i \otimes a_i)^\top$ in which all even-multiplicity entries are zeroed out.

$$\|Ax\|_4^4 = \left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ B_{\neq} \right\rangle + \sum_{\alpha,\beta \in [n]} x_\alpha^2 x_\beta^2 \cdot \sum_i a_i(\alpha)^2 a_i(\beta)^2.$$

By the above arguments, there is a sum-of-squares proof that

$$\begin{aligned}\left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ B_{\neq} \right\rangle &\leq \left\langle x^{\otimes 2}(x^{\otimes 2})^\top,\ \|B_{\neq}\| \cdot \mathrm{Id} \right\rangle \\ &= \|B_{\neq}\| \cdot \sum_{\alpha,\beta \in [d]} x_\alpha^2 x_\beta^2 = \|B_{\neq}\| \cdot \left(\sum_{\alpha \in [d]} x_\alpha^2 \right)^2.\end{aligned}$$

For the other term, we will use an even simpler bound. Let $c_{\alpha,\beta} = \sum_i a_i(\alpha)^2 a_j(\beta)^2$, for convenience. Also, let $c^* = \max_{\alpha,\beta} c_{\alpha,\beta}$. The following equality,

$$\sum_{\alpha,\beta \in [d]} c_{\alpha,\beta} \cdot x_\alpha^2 x_\beta^2 = c^* \cdot \sum_{\alpha,\beta} x_\alpha^2 x_\beta^2 - \left(\sum_{\alpha,\beta} (c^* - c_{\alpha,\beta}) \cdot x_\alpha^2 x_\beta^2 \right),$$

is a sum-of-squares proof that

$$\sum_{\alpha,\beta \in [d]} c_{\alpha,\beta} \cdot x_\alpha^2 x_\beta^2 \leq c^* \cdot \sum_{\alpha,\beta} x_\alpha^2 x_\beta^2,$$

because $c^* - c_{\alpha,\beta} \geq 0$ for all α, β by definition, and thus the parenthesized term is a sum-of-squares.

Putting the two arguments together, we have a sum-of-squares proof that

$$\|Ax\|_4^4 \leq (\|B_{\neq}\| + c^*) \cdot \left(\sum_\alpha x_\alpha^2 \right)$$

Because we have the SDP constraint that $\|x\|_2^2 = 1$, the objective value is thus bounded by

$$\tilde{E}[\|Ax\|_4^4] = (\|B_{\neq}\| + c^*) \cdot V\left[\left(\sum_{\alpha \in [d]} x_\alpha^2 \right)^2 \right] = \|B_{\neq}\| + c^*.$$

The final step in the proof consists of showing that with high probability over the choice of A,

$$\|B_{\neq}\| \leq \tilde{O}(\sqrt{d^3/n^3}) \quad \text{and} \quad c^* \leq \tilde{O}(1/n).$$

The first fact we can prove using a matrix Chernoff bound, and the second fact we can prove using a Chernoff bound and a union bound. This concludes the proof! □

4 Other Resources

Check out the following other resources for historical details and more sum-of-squares algorithms/lower bounds:

- For notes about SDPs and duality, I like these notes by Lap Chi Lau:

  `https://cs.uwaterloo.ca/~lapchi/cs270/notes.html`

  I also like these notes by Anupam Gupta and Ryan O'Donnell:

  `https://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15859-f11/www/`

- Lecture notes from Boaz Barak on sum-of-squares:

  `http://www.boazbarak.org/sos/`

- Lecture notes from Massimo Lauria on sum-of-squares and other relaxations for polynomial optimization `http://www.csc.kth.se/~lauria/sos14/`

- The introduction of this paper by Barak, Kelner and Steurer: `https://arxiv.org/pdf/1312.6652v1.pdf`

  The appendix of the paper also contains many sum-of-squares proofs of basic inequalities (e.g. Cauchy-Schwarz) that can be of use for providing good dual certificates.