Sum of Squares

Arpon Basu

Last updated December 4, 2023

Contents

Acknowledgements

This expository report has been made out of Pravesh Kothari's lecture series on the Sum of Squares hierarchy. I'd like to thank Pravesh Sir for making such a cutting edge topic so accessible.

Notation

Let $n \in \mathbb{N} = \{1, 2, \ldots\}$. Then we refer to the set $\{1, 2, \ldots, n\}$ as $[n]$.

We denote by $\mathbb{N}_0$ the set $\{0, 1, \ldots, \}$.

For any set S, we denote by 2^S the powerset of S.

Suppose we are given variables $x_1, \ldots, x_n$. Then for any set $S \subseteq [n]$, we denote by x_S the product $\prod_{i \in S} x_i$. We set $x_\varnothing = 1$.

Throughout this report, the (unsubscripted) variable x shall denote the n-dimensional vector $\begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix}^\mathsf{T}$.

We shall often overload the symbol 0, ie:- in different contexts, the same symbol 0 could mean the scalar 0 in $\mathbb{R}$, or the vector $0 \in \mathbb{R}^n$. However, we assure the reader that the meaning of any particular 0 will be clear from the context.

We shall often refer to $\{-1, 1\}^n, n \in \mathbb{N}$ as the *boolean hypercube* (*of dimension* n).

We shall often treat a function $f : A \mapsto B$ as a vector in the space B^A.

Unless mentioned otherwise, the topology on $\mathbb{R}^{n \times n}$ shall always be assumed to be the topology induced by the Frobenius norm, and the topology on $\mathbb{R}^{\{-1,1\}^n}$ [1] shall be assumed to be the topology induced by the ℓ_2 metric.

Let $\mu \in \mathbb{R}^n$ be any vector and let $\Sigma \in \mathbb{R}^{n \times n}$ be a symmetric PSD matrix. Then we denote by $\mathcal{N}(\mu, \Sigma)$ the Gaussian distribution with mean μ and covariance Σ.

We shall sometimes denote the symmetric difference of two sets S, T as $S \oplus T$ (instead of the more usual notation $S \triangle T$), ie:- $S \oplus T = S \triangle T = (S \setminus T) \cup (T \setminus S)$.
For multiple sets $S_1, S_2, \ldots, S_n$, we define

$$\bigoplus_{i=1}^{n} S_i := \{s : s \text{ occurs in an odd number of sets among } S_1, \ldots, S_n\}$$

For example, $\{1, 3\} \oplus \{2, 3, 4\} \oplus \{1, 4\} = \{2\}$.

All logarithms should be assumed to be base e unless mentioned otherwise.

[1] Note that $\mathbb{R}^{\{-1,1\}^n}$ is the space of all functions from $\{-1, 1\}^n$ to $\mathbb{R}$

§0. Preliminaries

Linear Algebra

We recall some basic facts from linear algebra.

Positive Semi Definite Matrices

Definition 0.1. Recall that real symmetric matrices have real eigenvalues. A real symmetric matrix $A \in \mathbb{R}^{n\times n}$ is called *Positive Semi Definite* (PSD) if all eigenvalues of A are non-negative. We write $A \succcurlyeq 0$ to denote that A is PSD.

Lemma 0.1. A real symmetric matrix $A \in \mathbb{R}^{n\times n}$ is PSD if and only if $\langle v, Av\rangle = v^{\mathsf{T}}Av \geq 0$ for every $v \in \mathbb{R}^n \setminus \{\mathbf{0}\}$.

The above innocuous rephrasing of the PSD condition has the following very important consequence.

Corollary 0.2. Let A, B be real symmetric PSD matrices of the same order, and let $\alpha, \beta \geq 0$ be real numbers. Then $\alpha A + \beta B$ is a PSD matrix.

Lemma 0.3. Every real symmetric PSD matrix $A \in \mathbb{R}^{n\times n}$ is equal to $B^{\mathsf{T}}B$ for some $B \in \mathbb{R}^{n\times n}$.

Lemma 0.4. Let $v_1, \ldots, v_m \in \mathbb{R}^n$ be vectors. Then $\sum_{i=1}^m v_i v_i^{\mathsf{T}}$ is a PSD matrix.

Frobenius Norm

Definition 0.2 (Matrix Norms)**.** For any matrix $X \in \mathbb{R}^{n\times n}$, we define the Frobenius norm of X to be

$$\|X\|_F^2 := \sum_{i=1}^n \sum_{j=1}^n |X_{ij}|^2 = \mathrm{tr}(XX^{\mathsf{T}})$$

Note that the Frobenius norm is just the ℓ_2 norm on $\mathbb{R}^{n^2}$.

Definition 0.3. For any two matrices $A, B \in \mathbb{R}^{n\times n}$, we define the inner product of those two matrices as

$$\langle A, B\rangle := \mathrm{tr}(AB^{\mathsf{T}})$$

It is easy to see that this inner product is associated with the Frobenius norm.

Remark. Matrix inner products are very convenient for expressing linear relations between the entries of a matrix. For example, suppose I have a 3×3 matrix A, which is given as

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Then I can encode the equation $0.5a_{11} - 0.2a_{21} + a_{22} - a_{23} + 0.9a_{33} = 5$ as $\langle C, A\rangle = 5$, where

$$C = \begin{bmatrix} 0.5 & 0 & 0 \\ -0.2 & 1 & -1 \\ 0 & 0 & 0.9 \end{bmatrix}$$

Lemma 0.5 (Trace-Eigenvalue Identity)**.** For a symmetric $X \in \mathbb{R}^{n\times n}$, $\mathrm{tr}(X) = \sum_{i=1}^n \lambda_i$, where $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ are the eigenvalues of X. Furthermore, $\sum_{i=1}^n \lambda_i^2 = \|X\|_F^2$.

Corollary 0.6. Let X be a real PSD matrix. Then $\|X\|_F \leq \mathrm{tr}(X)^2$.

Proof. Since X is PSD, all of its eigenvalues $(\lambda_1, \ldots, \lambda_n)$ are non-negative. Consequently, since $\mathrm{tr}(X) = \sum_{i=1}^n \lambda_i$, $\max_{i\in[n]} \lambda_i \leq \mathrm{tr}(X)$. Thus

$$\|X\|_F^2 = \sum_{i=1}^n \lambda_i^2 \leq \left(\max_{i\in[n]} \lambda_i\right) \cdot \sum_{i=1}^n \lambda_i \leq \mathrm{tr}(X)^2$$

as desired. ■

Operator Norms

Definition 0.4. Let $\|\cdot\|_p$ denote the ℓ_p-norm on $\mathbb{R}^n$. Then we define the ℓ_p-norm of a matrix $X \in \mathbb{R}^{n\times n}$ to be

$$\|X\|_p := \sup_{v\in\mathbb{R}^n\setminus\{\mathbf{0}\}} \frac{\|Xv\|_p}{\|v\|_p}$$

It can be shown that the above supremum exists (ie:- the supremum is some finite quantity). Also, by its very definition, we have $\|Xv\|_p \leq \|X\|_p \cdot \|v\|_p$ for every $v \in \mathbb{R}^n$.

Lemma 0.7. For any matrix $X \in \mathbb{R}^{n\times n}$, we have

$$\|X\|_2 \leq \|X\|_F \leq \sqrt{n}\|X\|_2$$

We can generalize the notion of operator norms to "interpolate" between two different norms on $\mathbb{R}^n$.

Definition 0.5 (Generalized Operator Norms)**.** Consider two norms $\|\cdot\|$ and $\|\cdot\|$ on $\mathbb{R}^n$. Then the generalized operator norm of a matrix A, w.r.t the aforementioned norms is defined as

$$\|A\| := \sup_{v \in \mathbb{R}^n \setminus \{\mathbf{0}\}} \frac{\|Av\|}{\|v\|}$$

When $\|\cdot\|$ is the ℓ_p norm, and $\|\cdot\|$ is the ℓ_q norm, $\|A\|$ is also denoted as $\|A\|_{q \to p}$.

Hadamard Products, Schur Product Theorem

Definition 0.6 (Hadamard Product)**.** Let $A, B \in \mathbb{R}^{m \times n}$ be two matrices. Their Hadamard product $A \circ B \in \mathbb{R}^{m \times n}$ is defined as

$$(A \circ B)_{ij} = A_{ij} B_{ij}$$

Theorem 0.8 (Schur Product Theorem)**.** Let A, B be PSD matrices of order n. Then $A \circ B$ is also PSD.

Proof. Since A, B are PSD, we have that

$$A = \sum_{i=1}^{n} \lambda_i u_i u_i^\mathsf{T}, B = \sum_{i=1}^{n} \mu_i v_i v_i^\mathsf{T}$$

where $\lambda_i, \mu_i, i \in [n]$ are the eigenvalues of A, B respectively.
Then

$$A \circ B \overset{\text{linearity of Hadamard product}}{=} \sum_{i,j} \lambda_i \mu_j (u_i u_i^\mathsf{T}) \circ (v_j v_j^\mathsf{T}) = \sum_{i,j} \lambda_i \mu_j (u_i \circ v_j)(u_i \circ v_j)^\mathsf{T}$$

where the last expression is PSD by Lemma 0.4. ■

Definition 0.7. Consider any $f : \mathbb{R} \mapsto \mathbb{R}$. For any $A \in \mathbb{R}^{n \times n}$, we define $f(A) := (f(a_{ij}))_{i,j \in [n]} \in \mathbb{R}^{n \times n}$.

Theorem 0.9. Let M be a PSD matrix all of whose diagonal entries are 1, and suppose f is a function all of whose Taylor series coefficients are positive. Furthermore, suppose the Taylor series of f is uniformly convergent on $[-1, 1]$. Then $f(M)$ is PSD.

Proof Sketch. Denote by $M^{(k)} := \underbrace{M \circ \cdots \circ M}_{k \text{ times}}$ for all $k \in \mathbb{N}_0$. By Theorem 0.8, $M^{(k)}$ is PSD for all $k \in \mathbb{N}_0$. Thus, for non-negative $c_k, k \in \mathbb{N}_0$, the (infinite) series $\sum c_k M^{(k)}$ is PSD [2]. But if $f = \sum c_k x^k$, then $f(M) = \sum c_k M^{(k)}$, and thus $f(M)$ is PSD. ■

[2] note that the the series defined by $S_\ell := \sum_{k \le \ell} c_k M^{(k)}$ converges to $f(M)$, ie:- $\lim_{\ell \to \infty} S_\ell = f(M)$. This follows directly from the fact that f is analytic, and thus the entries of S_ℓ converge to the respective entries of $f(M)$, as desired. Now, by Theorem 0.14, the set of PSD matrices is closed. Since S_ℓ is PSD for all $\ell \in \mathbb{N}_0$, the limit point of S_ℓ's, ie:- $f(M)$, must be PSD too.

Fourier Analysis

We will see some elementary Fourier analysis on the boolean hypercube.

Theorem 0.10 (Fourier Analysis on the boolean hypercube)**.** Let n be a natural number. Consider any function $f : \{-1,1\}^n \mapsto \mathbb{R}$. Then there exists a unique function $\widehat{f} : 2^{[n]} \mapsto \mathbb{R}$ such that

$$f(x) = \sum_{S \subseteq [n]} \widehat{f}(S) x_S$$

for every $x = (x_1, \ldots, x_n) \in \{-1,1\}^n$.
The function $\widehat{f}$ is also known as the *Fourier transform* of f.

Proof. We prove this statement by induction on n. For $n = 1$, note that any function $f : \{-1,1\} \mapsto \mathbb{R}$ can be written as $f(x) = \left(\frac{f(1)+f(-1)}{2}\right) + \left(\frac{f(1)-f(-1)}{2}\right) \cdot x$, and further note that this representation is the unique representation of the form $\widehat{f}(\varnothing) + \widehat{f}(\{1\}) \cdot x$.
Thus the base case of our induction hypothesis is verified. Now, suppose the statement is true for some $n = k-1, k \geq 2$. Then note that any function $f : \{-1,1\}^k \mapsto \mathbb{R}$ can be written as

$$f(x_1, x_2, \ldots, x_k) = \left(\frac{f(1, x_2, \ldots, x_k) + f(-1, x_2, \ldots, x_k)}{2}\right) + \left(\frac{f(1, x_2, \ldots, x_k) - f(-1, x_2, \ldots, x_k)}{2}\right) \cdot x_1$$

But $g(x_2, \ldots, x_k) := \frac{f(1,x_2,\ldots,x_k)+f(-1,x_2,\ldots,x_k)}{2}$ and $h(x_2, \ldots, x_k) := \frac{f(1,x_2,\ldots,x_k)-f(-1,x_2,\ldots,x_k)}{2}$ are functions on the $(k-1)$-dimensional boolean hypercube and thus by the induction hypothesis possess a unique Fourier transform. Then combining the Fourier transforms for those two functions yields a Fourier transform for f, and it is not too difficult to see that Fourier transform is unique too. ■

Definition 0.8 (Multilinear Polynomials)**.** A multivariate polynomial is called multilinear if it is linear (affine) in each of its variables. For example, $3x - 4xy + 5z - 2$ is a multilinear polynomial in x, y, z, but $x^2 + 4xy$ is not.

Corollary 0.11. *Any* function on the boolean hypercube is equivalent to a multilinear polynomial of degree at most n.

Lemma 0.12. Every polynomial of degree d over the boolean hypercube is equivalent to a multilinear polynomial of degree at most d. Furthermore, by the uniqueness of the Fourier transform, this multilinear polynomial is also the Fourier transform of our polynomial.

Proof. Note that over the boolean hypercube, every polynomial is equivalent to a multilinear polynomial of lower degree: One can see this even without invoking the Fourier expansion of the polynomial. Indeed, note that if $x_i \in \{-1,1\}$, then $x_i^2 = 1$. Consequently, every term $\prod_{i=1}^n x_i^{k_i}$ in the polynomial can be replaced by the multilinear term $\prod_{i=1}^n x_i^{k_i \bmod 2}$, and thus we get an equivalent multilinear polynomial with a degree at most the original polynomial, as desired. ■

Corollary 0.13. Multilinear polynomials are their own Fourier decompositions.

Topology

The reader is advised to recall the basic notions of topology such as open and closed sets, metric topologies, and compactness, before reading this section.
Now, note that both the Frobenius norm and the ℓ_p norms induce a topology on $\mathbb{R}^{n\times n}$. Furthermore, with the help of Lemma 0.7, it can be shown that the topologies induced by the Frobenius norm and the ℓ_2 norm are the same. Thus from now on we can freely use the Frobenius norm or the ℓ_2 norm, according to our convenience, without worrying about the underlying topology getting changed.
We shall always equip $\mathbb{R}^{\{-1,1\}^n}$ with the topology induced by the ℓ_2-metric. Just to clarify, through an example, what the ℓ_2-metric on $\mathbb{R}^{\{-1,1\}^n}$ entails, consider the functions f_1, f_2 below, mapping $\{-1,1\}^2$ to $\mathbb{R}$:

$$f_1(-1,-1) = 0.2, f_1(-1,1) = -0.5, f_1(1,-1) = 0, f_1(1,1) = 2$$

$$f_2(-1,-1) = -0.8, f_2(-1,1) = 0.5, f_2(1,-1) = 1, f_2(1,1) = 0$$

Then $\|f_1 - f_2\|_2 = \sqrt{(0.2-(-0.8))^2 + (-0.5-0.5)^2 + (0-1)^2 + (2-0)^2} = \sqrt{7}$.

Convex Analysis

Definition 0.9. Consider a set $C \subseteq V$, where V is a real vector space. Then

1. C is called convex if for any $c_1, c_2 \in C$ and $\lambda \in [0,1]$, we have $\lambda c_1 + (1-\lambda)c_2 \in C$.
2. C is called a cone if for any $c \in C, \lambda > 0$, we have $\lambda c \in C$.

Theorem 0.14. Let $\mathcal{A} \subseteq \mathbb{R}^{n\times n}$ be the set of real symmetric PSD matrices. Also, equip $\mathbb{R}^{n\times n}$ with the usual topology, ie:- the topology induced by the Frobenius norm. Then $\mathcal{A}$ is a closed convex cone.

Proof. The fact that $\mathcal{A}$ is a convex cone follows directly from Corollary 0.2. Now, note that

$$\mathcal{A} = \mathcal{S} \cap \bigcap_{v \in \mathbb{R}^n \setminus \{\mathbf{0}\}} f_v^{-1}([0,\infty))$$

where for every $v \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ we define the function $f_v : \mathbb{R}^{n\times n} \mapsto \mathbb{R}, f_v(X) := v^\mathsf{T} X v$, and $\mathcal{S} \subseteq \mathbb{R}^{n\times n}$ is the set of symmetric matrices.
Now, observe that

1. $\mathcal{S}$ is closed: Indeed, consider the function $g : \mathbb{R}^{n\times n} \mapsto \mathbb{R}^{n\times n}$, $g(X) := X - X^\mathsf{T}$. Then $\mathcal{S} = g^{-1}(\{O\})$. Now, since g is continuous, and since $\{O\}$ is a closed set, $\mathcal{S}$, being the pre-image of a closed set under a continuous function must be closed too.
2. $f_v^{-1}([0,\infty))$ is closed for every $v \in \mathbb{R}^n \setminus \{\mathbf{0}\}$: Once again, since f_v is continuous, and since $[0,\infty)$ is closed, $f_v^{-1}([0,\infty))$ must be closed too.

Thus $\mathcal{A}$ is a closed set since an intersection of closed sets is also closed. ■

Remark. The above proof was taken from here.

Closed convex sets are a central object of study in convex analysis, so we shall now develop some machinery to deal with them.

Lemma 0.15 (Separating Hyperplane Lemma)**.** Let V be a finite-dimensional vector space over reals. Let $C \subseteq V$ be a nonempty closed convex set. Consider any $y \in V \setminus C$. Then there exists $a \in V \setminus \{\mathbf{0}\}, b \in \mathbb{R}$ such that $\langle a, y\rangle < b < \langle a, x\rangle$ for every $x \in C$.

Proof. Proof can be found in section 2.5 of [BV04]. ■

Corollary 0.16. Let V be a finite-dimensional vector space over reals. Let $C \subseteq V$ be a nonempty closed convex *cone*. Consider any $y \in V \setminus C$. Then there exists $a \in V \setminus \{\mathbf{0}\}$ such that $\langle a, y\rangle < 0 \leq \langle a, x\rangle$ for every $x \in C$.

Proof. Apply Lemma 0.15 to C to get a, b such that $\langle a, y\rangle < b < \langle a, x\rangle$ for every $x \in C$. Now, since C is a cone, $\varepsilon x \in C$ and consequently, $b < \langle a, \varepsilon x\rangle = \varepsilon\langle a, x\rangle$ for every $\varepsilon > 0, x \in C$.
Now, if:

1. $b > 0$: Then we can choose a small enough $\varepsilon > 0$ to make $\varepsilon\langle a, x\rangle$ smaller than b, leading to a contradiction.
2. $b < 0$: Suppose there is some $x \in C$ such that $\langle a, x\rangle < 0$. Then we can choose a large enough $\varepsilon > 0$ to make $\varepsilon\langle x, a\rangle$ smaller than b, leading to a contradiction.

Thus we either have $b = 0$, or we have $\langle x, a\rangle \geq 0 > b$ for every $x \in C$. In either case, our result holds. ■

We now move on to the algorithmic aspects of convex analysis.

Definition 0.10 (ε-thickening)**.** Let $K \subseteq \mathbb{R}^N$ be a closed convex bounded set. Also, let $\varepsilon > 0$ be a parameter. We then define the ε-thickening of K to be:

$$K_\varepsilon := \{x + \tau : x \in K, \|\tau\|_2 < \varepsilon\}$$

Definition 0.11 (Weak Separation Oracle)**.** Let $K \subseteq \mathbb{R}^N$ be a closed convex bounded set. Also, let $\varepsilon > 0$ be a parameter.
A *weak separation oracle* for K takes as input some $x \in \mathbb{Q}^N \subseteq \mathbb{R}^N$, and:

1. Correctly asserts that $x \in K_\varepsilon$, or
2. Returns an "almost separating hyperplane", ie:- gives us some $a \in \mathbb{R}^n \setminus \{0\}$ such that $\langle a, x\rangle > \langle a, z\rangle - \varepsilon\|a\|_2$ for every $z \in K$.

Remark. One can guess why the term "weak" has been used: This is because the oracle doesn't tell us if some input x belongs to our convex body or not. It either tells us that x is inside K, with some ε margin, or it separates x from points sufficiently deep inside K.

We now state a seminal result by [GLS88].

Theorem 0.17. Let $K \subseteq \mathbb{R}^N$ be a closed convex bounded set. Also, let $\varepsilon > 0$ be a parameter.
Furthermore, let $p \in \mathbb{R}^N$ be such that $B(p,r) \subseteq K \subseteq B(p,R)$, where $B(p,t)$ denotes the open ball centered at p, of radius t, and $R > r > 0$ are real numbers. $\frac{R}{r}$ is also known as the aspect ratio of K.
Also suppose that we have access to a weak separation oracle for K, with parameter ε.
Then given any $v \in \mathbb{Q}^N$, one can compute $x \in \mathbb{R}^N$ such that:

1. $x \in K$,
2. $\langle v, x\rangle > \langle v, z\rangle - \varepsilon$ for every $z \in K$.

Furthermore, x can be computed in $\text{poly}\left(\log \frac{R}{r} + \log \frac{1}{\varepsilon} + N\right)$ time.

Remark. Note that the theorem says that x can be *computed* in $\text{poly}\left(\log \frac{R}{r} + \log \frac{1}{\varepsilon} + N\right)$ time: This implies that all entries of x are actually rational numbers with $\text{poly}\left(\log \frac{R}{r} + \log \frac{1}{\varepsilon} + N\right)$-bit complexity.
Thus WLOG when we invoke this theorem to solve any SDP later on, we can assume that its output complies with Convention 1.

Tensor Notation

Definition 0.12 (Tensor Product)**.** Consider a vector $v \in \mathbb{R}^k$. Then the tensor product of v with itself, t times, is denoted as $v^{\otimes t} \in \mathbb{R}^{k^t}$, where the elements of $v^{\otimes t}$ are indexed by the elements of $[k]^t$, and for any $(k_1, \ldots, k_t) \in [k]^t$, we define

$$(v^{\otimes t})_{(k_1,\ldots,k_t)} = \prod_{i=1}^{t} v_{k_i}$$

At this point, we make a very important observation: Consider the tensor product $(1,x)^{\otimes d} := \left(\begin{bmatrix}1 & x_1 & \ldots & x_n\end{bmatrix}^\intercal\right)^{\otimes d}$. Then the entries of $(1,x)^{\otimes d}$ contain all multilinear monomials on $x_1, \ldots, x_n$ of degree at most d. Indeed, consider the example

$$\left(\begin{bmatrix}1 & x_1 & x_2\end{bmatrix}^\intercal\right)^{\otimes 2} = \begin{bmatrix}1\cdot 1 & 1\cdot x_1 & 1\cdot x_2 & x_1\cdot 1 & x_1\cdot x_1 & x_1\cdot x_2 & x_2\cdot 1 & x_2\cdot x_1 & x_2\cdot x_2\end{bmatrix}^\intercal$$

$$= \begin{bmatrix}1 & x_1 & x_2 & x_1 & x_1^2 & x_1x_2 & x_2 & x_2x_1 & x_2^2\end{bmatrix}^\intercal$$

It is clear that this vector contains all monomials of degree at most 2 on the variables x_1, x_2. Also, note how many monomials such as x_1x_2 are repeated multiple times: This redundancy is the price we pay for the economy of expression that the tensor notation provides us. It is important to note though, that this redundancy doesn't (asymptotically) increase the space required to store all monomials of degree d. Indeed, there are $\mathcal{O}(n^d)$ monomials of degree at most d, and the size of $(1,x)^{\otimes d}$ is $(n+1)^d = \mathcal{O}(n^d)$.
In general, note that $(1,x)^{\otimes d}$ resides in $\mathbb{R}^{(n+1)^d}$.

Some Aspects of the Gaussian Distribution

We shall see various aspects of the Gaussian distribution which we shall use repeatedly in our proofs throughout the material.

Lemma 0.18 (Sheppard's Lemma)**.** Consider the Gaussian distribution $\mathcal{G} = \mathcal{N}(0, \Sigma)$ on $\mathbb{R}^2$, where $\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ and $\rho \in [-1, 1]$.
Suppose we sample $g = \begin{bmatrix} g_1 & g_2 \end{bmatrix}^\mathsf{T}$ from $\mathcal{G}$. Then

$$\Pr(\text{sign}(g_1) \neq \text{sign}(g_2)) = \frac{\arccos(\rho)}{\pi}$$

Proof. We'll come up with an alternate sampling procedure that yields the same distribution as $\mathcal{G}$, yet is easier to analyze.
Consider two vectors $v, w \in \mathbb{R}^2$ such that $\|v\|_2 = \|w\|_2 = 1$, and $\langle v, w \rangle = \rho$, ie:- v and w are two unit vectors with an angle of $\arccos(\rho)$ between them. WLOG we can assume that w can be obtained by rotating v clockwise by $\arccos(\rho)$ radians.
Sample $h \in \mathbb{R}^2$ from $\mathcal{N}(0, I_2)$, ie:- the standard 2-dimensional Gaussian.
Define $\widehat{g_1} := \langle h, v \rangle$, and $\widehat{g_2} := \langle h, w \rangle$. Now, note that

$$\widehat{g} := \begin{bmatrix} \widehat{g_1} \\ \widehat{g_2} \end{bmatrix} = Bh$$

where $B = \begin{bmatrix} v^\mathsf{T} \\ w^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{2 \times 2}$.

Consequently, $\widehat{g} \sim \mathcal{N}(B \cdot 0, B \cdot I_2 \cdot B^\mathsf{T}) = \mathcal{N}(0, BB^\mathsf{T}) = \mathcal{G}$, since $BB^\mathsf{T} = \begin{bmatrix} v^\mathsf{T}v & v^\mathsf{T}w \\ w^\mathsf{T}v & w^\mathsf{T}w \end{bmatrix} = \begin{bmatrix} \|v\|_2^2 & \langle v, w \rangle \\ \langle w, v \rangle & \|w\|_2^2 \end{bmatrix} = \Sigma$.

Thus the distribution of $\widehat{g}$ is same as the distribution of g, and

$$\Pr(\text{sign}(g_1) \neq \text{sign}(g_2)) = \Pr(\text{sign}(\widehat{g_1}) \neq \text{sign}(\widehat{g_2})) = \Pr(\text{sign}(\langle h, v \rangle) \neq \text{sign}(\langle h, w \rangle))$$

Consider the 4 possible values of $\widehat{h} := \frac{h}{\|h\|_2}$ for which $\langle \widehat{h}, v \rangle = 0$ or $\langle \widehat{h}, w \rangle = 0$. These 4 values of $\widehat{h}$ split the unit circle into 4 arcs, two of angle $\arccos(\rho)$, and two of angle $\pi - \arccos(\rho)$. Finally, note that the signs of $\langle h, v \rangle$ and $\langle h, w \rangle$ can be different only if $\widehat{h}$ lies in the arcs of size $\arccos(\rho)$. Thus

$$\Pr(\text{sign}(\langle h, v \rangle) \neq \text{sign}(\langle h, w \rangle)) = \frac{2 \arccos(\rho)}{2\pi}$$

where equality follows from the fact that $\widehat{h}$ is distributed uniformly on the unit circle. ■

Tail Bounds

We state some handy tail bounds here which help us deal with Gaussians.

Lemma 0.19. If $Z_1, \ldots, Z_t$ are jointly Gaussian random variables, then

$$\text{Var}\left(\max_{k \in [t]} Z_k \right) \leq \mathcal{O}(1) \max_{k \in [t]} \text{Var}\left(Z_k \right) \tag{0.1}$$

§1. Introduction and Basic Definitions

In this chapter, we shall be concerned with certificates of non-negativity of polynomials. The relevance of this investigation to the field of algorithms will become clearer in later chapters, where we shall also explain how these ideas fit into the larger framework of theoretical computer science. For now, the reader is asked to view these things from a purely abstract mathematical perspective.

1.1. Sum of Squares (SoS) Certificates

Definition 1.1. Let d be an even natural number. A degree d SoS certificate of non-negativity of a function $f : \{-1,1\}^n \mapsto \mathbb{R}$ is a list of polynomials $g_1, \dots, g_r$ on the boolean hypercube such that $\deg(g_i) \leq \frac{d}{2}$ for every $i \in [r]$, and

$$f(x) = \sum_{i=1}^{r} g_i(x)^2$$

for every $x \in \{-1,1\}^n$.

Remark. The motivation behind this definition is as follows: Suppose we want to *prove* that a function is non-negative on its domain. Then an easy way to do so is to express the function as a sum of squares. The above definition seeks to formalize exactly that.
The degree of the polynomials forming the certificate is emphasized because low-degree polynomials take *less space* to write down. For example, a n-degree polynomial in n variables may take $\mathcal{O}(n^n)$ space to write down, which, as we shall see later, is terrible for its algorithmic utility.
Consequently, we parametrize the sum of squares certificate through its degree. We shall expend some effort in determining when low-degree certificates of non-negativity exist, and when they do, how one should find them.
We now prove that for the boolean hypercube, SoS certificates of non-negativity can always be found [3].

Lemma 1.1. Every non-negative function $f : \{-1,1\}^n \mapsto \mathbb{R}_{\geq 0}$ has a degree $2n$ SoS certificate.

Proof. Consider the function $g(x) := \sqrt{f(x)}$. By Corollary 0.11, $g(x)$ is (equivalent to) a polynomial of degree at most n, and thus the lemma follows. ■

Over the boolean hypercube, we can achieve even more: Not only do SoS certificates always exist, but sufficiently low-degree certificates can also be found, if our polynomial itself is low-degree, and we are allowed to add large constants to our input polynomial.

Lemma 1.2. Consider any polynomial $f : \{-1,1\}^n \mapsto \mathbb{R}$ of degree at most d, where d is an even natural number. Then there exists a constant $L > 0$ such that $L + f$ has a degree d SoS certificate.

Proof. By Lemma 0.12, the Fourier decomposition of f contains terms of degree at most d.
Now, consider the Fourier decomposition of f:

$$f = \sum_{\substack{S \subseteq [n] \\ |S| \leq d}} \widehat{f}(S) x_S$$

[3] however, for general domains, this statement is not true: For example, the polynomial $f(x) = x_3^6 + x_1^4 x_2^2 + x_1^2 x_2^4 - 3x_1^2 x_2^2 x_3^2$ over $\mathbb{R}^3$ is non-negative everywhere, yet it is not expressible as a sum of squares of polynomials

Now suppose we show that for every term $\widehat{f}(S)x_S$ in the above summation, there is some constant L_S such that $L_S + \widehat{f}(S)x_S$ has a degree d SoS certificate.
Then by setting $L = \sum_{\substack{S\subseteq[n]\\|S|\leq d}} L_S$, we would have that

$$f + L = \sum_{\substack{S\subseteq[n]\\|S|\leq d}} (\widehat{f}(S)x_S + L_S)$$

We could then simply combine the certificates of each term in the above summation to get a certificate for $f + L$.
We now claim that $|\alpha| + \alpha x_S$ has a degree d SoS certificate for any $\alpha \in \mathbb{R}$ and any $S \subseteq [n]$ such that $|S| \leq d$. In particular, we can simply choose $L_S = |\widehat{f}(S)|$ in the above argument.
We thus focus on proving the aforementioned claim. Now, note that $|\alpha| + \alpha x_S = |\alpha| \cdot (1 + \text{sign}(\alpha)x_S)$. Finally, since $|S| \leq d$, one can find sets $T_1, T_2 \subseteq [n]$ such that $T_1 \cup T_2 = S, T_1 \cap T_2 = \varnothing, |T_1|, |T_2| \leq \frac{d}{2}$. Then note that

$$\frac{1}{2}(x_{T_1} \pm x_{T_2})^2 = \frac{1}{2}(x_{T_1}^2 + x_{T_2}^2 \pm 2x_{T_1}x_{T_2})$$

Since $x \in \{-1, 1\}^n$, $x_{T_1}^2 = x_{T_2}^2 = 1$. Also, $x_{T_1}x_{T_2} = x_S$. Thus

$$\frac{1}{2}(x_{T_1}^2 + x_{T_2}^2 \pm 2x_{T_1}x_{T_2}) = \frac{1}{2}(2 \pm 2x_{T_1}x_{T_2}) = 1 \pm x_S$$

$$|\alpha| \cdot (1 + \text{sign}(\alpha)x_S) = \frac{|\alpha|}{2}(x_{T_1} + \text{sign}(\alpha)x_{T_2})^2 = \left(\underbrace{\sqrt{\frac{|\alpha|}{2}}(x_{T_1} + \text{sign}(\alpha)x_{T_2})}_{\text{degree at most } \frac{d}{2}}\right)^2$$

as desired. ■

Theorem 1.3. A function f on the boolean hypercube has a degree d SoS certificate if and only if there exists a PSD matrix $A \in \mathbb{R}^{(n+1)^{\frac{d}{2}} \times (n+1)^{\frac{d}{2}}}$ such that $f(x) = \langle (1, x)^{\otimes \frac{d}{2}}, A(1, x)^{\otimes \frac{d}{2}} \rangle$ for every x in the hypercube.

Proof. Suppose $f(x) = \langle (1, x)^{\otimes \frac{d}{2}}, A(1, x)^{\otimes \frac{d}{2}} \rangle$ for every x in the hypercube for some PSD matrix A. Then by Lemma 0.3, there exists some matrix B such that $A = B^\mathsf{T}B$. Then

$$f(x) = \langle (1, x)^{\otimes \frac{d}{2}}, A(1, x)^{\otimes \frac{d}{2}} \rangle = \langle (1, x)^{\otimes \frac{d}{2}}, B^\mathsf{T}B(1, x)^{\otimes \frac{d}{2}} \rangle = \langle B(1, x)^{\otimes \frac{d}{2}}, B(1, x)^{\otimes \frac{d}{2}} \rangle = \|B(1, x)^{\otimes \frac{d}{2}}\|_2^2$$

Now note that every entry of $B(1, x)^{\otimes \frac{d}{2}}$ is a multilinear polynomial of degree at most $\frac{d}{2}$. Since $f(x)$ is the sum of the squares of the entries of $B(1, x)^{\otimes \frac{d}{2}}$, we get that the $(n+1)^{\frac{d}{2}}$ entries of $B(1, x)^{\otimes \frac{d}{2}}$ form a degree d SoS certificate for f.
Conversely, suppose f has a degree d certificate, ie:- $f = \sum_{i=1}^r g_i^2$, where $\deg(g_i) \leq \frac{d}{2}$. By Corollary 0.11, WLOG we can assume that g_i's are multilinear polynomials. Consequently, $g_i(x) = \langle v_i, (1, x)^{\otimes \frac{d}{2}} \rangle$, where the vector v_i just "chooses" the entries in $(1, x)^{\otimes \frac{d}{2}}$ that it needs. Thus

$$f = \sum_{i=1}^r g_i^2 = \sum_{i=1}^r \langle v_i, (1 + x)^{\otimes \frac{d}{2}} \rangle^2 = \sum_{i=1}^r \langle v_i v_i^\mathsf{T}(1, x)^{\otimes \frac{d}{2}}, (1, x)^{\otimes \frac{d}{2}} \rangle = \left\langle \sum_{i=1}^r v_i v_i^\mathsf{T}(1, x)^{\otimes \frac{d}{2}}, (1, x)^{\otimes \frac{d}{2}} \right\rangle$$

Now, by Lemma 0.4, we have that $A := \sum_{i=1}^r v_i v_i^\mathsf{T}$ is a PSD matrix. Then

$$f(x) = \langle A(1, x)^{\otimes \frac{d}{2}}, (1, x)^{\otimes \frac{d}{2}} \rangle = \langle (1, x)^{\otimes \frac{d}{2}}, A(1, x)^{\otimes \frac{d}{2}} \rangle$$

as desired. ■

Corollary 1.4. Suppose f has a degree d SoS certificate. Then f has a degree d SoS certificate with at most $(n+1)^{\frac{d}{2}}$ polynomials.

We conclude this section by stating a very important lemma concerning the *geometry* of polynomials with degree d SoS certificates.

Lemma 1.5. Let $\mathrm{SoS}_d \subseteq \mathbb{R}^{\{-1,1\}^n}_{\geq 0} \subseteq \mathbb{R}^{\{-1,1\}^n}$ be the set of functions having a degree d SoS certificate. Then SoS_d is a closed convex cone, where $\mathbb{R}^{\{-1,1\}^n}$ is equipped with the topology induced by the ℓ_2-metric.

Remark. The proof of this lemma is similar to the proof of Theorem 0.14.

1.2. Pseudo-Distributions

We shall now define the separate notion of pseudo-distributions. This might feel like a non-sequitur, but it is not: It turns out that pseudo-distributions are "dual" to SoS certificates, in some sense. As usual, we shall have to plod through some definitions before getting to the fun part.

Definition 1.2 (Formal Expectation)**.** For any $\mu \in \mathbb{R}^{\{-1,1\}^n}$, not necessarily a probability distribution, we define the formal expectation of $f \in \mathbb{R}^{\{-1,1\}^n}$ w.r.t. μ to be

$$\widetilde{\mathbb{E}}_\mu[f] := \langle \mu, f \rangle = \sum_{x \in \{-1,1\}^n} \mu(x) f(x)$$

Remark. Note that $\widetilde{\mathbb{E}}$ is linear, just like $\mathbb{E}$ is.

Definition 1.3 (Pseudo-Distributions)**.** A function $\mu : \{-1,1\}^n \mapsto \mathbb{R}$ is called a degree d pseudo-distribution (p.d.) over $\{-1,1\}^n$ if:

1. $\widetilde{\mathbb{E}}_\mu[1] = 1$, ie:- $\sum_{x \in \{-1,1\}^n} \mu(x) = 1$.
2. For all polynomials f of degree $\leq \frac{d}{2}$, we have $\widetilde{\mathbb{E}}_\mu[f^2] \geq 0$.

Pseudo-distributions are an attempt to extend the properties of actual distributions (which are non-negative vectors with entries summing to 1) to vectors in general, ie:- vectors potentially having some negative entries too. Obviously, that can't be done in entirety: We thus limit the scope of our rules and enforce similarity with distributions in that limited domain. Thus, for example, for an actual distribution ν, we would have had $\mathbb{E}_\nu[f^2] \geq 0$ for any function f: For pseudo-distributions of degree d, we restrict ourselves to polynomials of degree at most $\frac{d}{2}$.
While this restriction might seem a bit unnatural and artificial at this point, we shall soon see that these restrictions arise very naturally, especially when we explore pseudo-distributions in relation to SoS certificates.

Lemma 1.6. μ is a degree d pseudo-distribution if and only if $\widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\mathsf{T}]$ is a PSD matrix with ones on its diagonals.
Furthermore, for any $S \subseteq [n]$ such that $|S| \leq d$, $|\widetilde{\mathbb{E}}_\mu[x_S]| \leq 1$.

Remark. Note that the pseudo-expectation of a matrix/vector is taken entrywise.

Proof. Let μ be a degree d pseudo-distribution. Let $M = \widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\mathsf{T}]$. Now, if f is a degree $\leq \frac{d}{2}$ polynomial where $f = \sum_{S \subseteq [n]} \widehat{f}(S) x_S$, then note that $\widetilde{\mathbb{E}}_\mu[f^2] = v^\mathsf{T} M v$, where v is a vector composed of the Fourier coefficients $\widehat{f}(S)$.
Since $\widetilde{\mathbb{E}}_\mu[f^2] \geq 0$ for all functions of degree at most $\frac{d}{2}$, we get that $v^\mathsf{T} M v \geq 0$ for all vectors $v \in \mathbb{R}^{(n+1)^{\frac{d}{2}}}$. Consequently, M is PSD, as desired.
Now, note that the diagonal entries of M are of the form x_T^2 for some set $T \subseteq [n], |T| \leq \frac{d}{2}$. But $x_T^2 = 1$ for any T, and since $\widetilde{\mathbb{E}}_\mu[1] = 1$, we get that all the diagonal entries of M are 1, as desired.
Conversely, if $M := \widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\mathsf{T}]$ is a PSD matrix with ones on its diagonals, then $\widetilde{\mathbb{E}}_\mu[1] = 1$ (which is obtained by examining the top-left entry of M). Furthermore, by varying v in $\mathbb{R}^{(n+1)^{\frac{d}{2}}}$, we see that $\widetilde{\mathbb{E}}_\mu[f^2] \geq 0$ for any polynomial f of degree $\leq \frac{d}{2}$, and thus μ is a degree d pseudo-distribution.
Finally, suppose x_S is the $(i,j)^{\text{th}}$ entry of M for some i, j. Then by taking a v whose only non-zero entries are in the i^{th} and j^{th} indices, examining the identity $v^\mathsf{T} M v \geq 0$ yields that the 2×2 matrix $\begin{bmatrix} \widetilde{\mathbb{E}}_\mu[x_{T_1}^2] & \widetilde{\mathbb{E}}_\mu[x_{T_1}x_{T_2}] \\ \widetilde{\mathbb{E}}_\mu[x_{T_1}x_{T_2}] & \widetilde{\mathbb{E}}_\mu[x_{T_2}^2] \end{bmatrix}$ is PSD, where $x_{T_1}x_{T_2} = x_S$. Thus the determinant of $\begin{bmatrix} \widetilde{\mathbb{E}}_\mu[x_{T_1}^2] & \widetilde{\mathbb{E}}_\mu[x_{T_1}x_{T_2}] \\ \widetilde{\mathbb{E}}_\mu[x_{T_1}x_{T_2}] & \widetilde{\mathbb{E}}_\mu[x_{T_2}^2] \end{bmatrix} = \begin{bmatrix} 1 & \widetilde{\mathbb{E}}_\mu[x_S] \\ \widetilde{\mathbb{E}}_\mu[x_S] & 1 \end{bmatrix}$ must be non-negative, ie:- $1 - \widetilde{\mathbb{E}}_\mu[x_S]^2 \geq 0 \implies |\widetilde{\mathbb{E}}_\mu[x_S]| \leq 1$, as desired. ■

Remark. Let μ be a degree d pseudo-distribution, and let $M = \widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\mathsf{T}]$. Then M is a PSD matrix all of whose entries lie in $[-1,1]$. Furthermore, let $\lambda_1, \ldots, \lambda_{(n+1)^{\frac{d}{2}}} \geq 0$ be the eigenvalues of M. Then

$$\det(M) = \prod_{i \in \left[(n+1)^{\frac{d}{2}}\right]} \lambda_i \overset{\text{AM-GM Inequality}}{\leq} \left(\frac{\sum_{i \in \left[(n+1)^{\frac{d}{2}}\right]} \lambda_i}{(n+1)^{\frac{d}{2}}} \right)^{(n+1)^{\frac{d}{2}}} = \left(\frac{\text{tr}(M)}{(n+1)^{\frac{d}{2}}} \right)^{(n+1)^{\frac{d}{2}}} = 1$$

Thus the determinant of M is bounded above by 1.

Lemma 1.7. Let μ be a pseudo-distribution of degree $2n$ on the n-dimensional boolean hypercube. Then μ is actually a probability distribution.

Proof. Consider the indicator function $f_y = \mathbb{1}_y$ for any $y \in \{-1,1\}^n$. By Corollary 0.11, f_y is equivalent to a polynomial of degree at most $2n$, and consequently $\widetilde{\mathbb{E}}_\mu[f_y^2] \geq 0$. But $\widetilde{\mathbb{E}}_\mu[f_y^2] = \mu(y)$, and consequently, μ is non-negative. Since the entries of μ sum to 1 by definition, μ is actually a distribution. ■

We shall now prove some results that anticipate the duality between pseudo-distributions and SoS certificates. Before that, we prove a lemma.

Lemma 1.8. Suppose we have $S, T \subseteq [n], S \neq T$. Then

$$\sum_{x \in \{-1,1\}^n} x_S \cdot x_T = 0$$

Proof. Firstly, note that $x_S \cdot x_T = x_{S \Delta T}$, where $S \Delta T := (S \setminus T) \cup (T \setminus S)$ is the symmetric difference of S and T. This is because if $i \in S \cap T$, then x_i occurs (exactly) twice in $x_S \cdot x_T$, and $x_i^2 = 1$ since $x_i \in \{-1, 1\}$. Thus all indices common to S and T get annihilated and only the indices in the symmetric difference remain.
Now, since $S \neq T$, $R := S \Delta T \neq \varnothing$. Now, its easy to see that as x varies over $\{-1, 1\}^n$, x_R becomes 1 and -1 equal number of times. Consequently, $\sum_{x \in \{-1,1\}^n} x_S \cdot x_T = \sum_{x \in \{-1,1\}^n} x_R = 0$, as desired. ■

Lemma 1.9. Let μ be any degree d pseudo-distribution. Then there exists a multilinear polynomial μ' of degree at most d such that

$$\widetilde{\mathbb{E}}_{\mu}[f] = \widetilde{\mathbb{E}}_{\mu'}[f]$$

for every polynomial f of degree at most d.

Proof. Consider the Fourier decomposition of μ,

$$\mu = \sum_{S \subseteq [n]} \widehat{\mu}(S) x_S = \underbrace{\sum_{\substack{S \subseteq [n] \\ |S| \leq d}} \widehat{\mu}(S) x_S}_{:= \mu'} + \underbrace{\sum_{\substack{S \subseteq [n] \\ |S| > d}} \widehat{\mu}(S) x_S}_{:= \mu''}$$

ande define the degree d part of it as μ', and the remaining as μ''. Now, let f be a polynomial of degree at most d. Then by Lemma 0.12, the Fourier expansion of f contains terms of degree at most d.
Now,

$$\widetilde{\mathbb{E}}_{\mu}[f] = \sum_{x \in \{-1,1\}^n} \mu(x) f(x) = \sum_{x \in \{-1,1\}^n} (\mu'(x) + \mu''(x)) f(x) = \widetilde{\mathbb{E}}_{\mu'}[f] + \widetilde{\mathbb{E}}_{\mu''}[f]$$

Note that if we can show that $\widetilde{\mathbb{E}}_{\mu''}[f] = 0$, then we're done.
Now,

$$\widetilde{\mathbb{E}}_{\mu''}[f] = \sum_{x \in \{-1,1\}^n} \mu''(x) f(x) = \sum_{x \in \{-1,1\}^n} \left(\left(\sum_{\substack{S \subseteq [n] \\ |S| > d}} \widehat{\mu}(S) x_S \right) \cdot \left(\sum_{\substack{T \subseteq [n] \\ |T| \leq d}} \widehat{f}(T) x_T \right) \right)$$

$$= \sum_{x \in \{-1,1\}^n} \left(\sum_{\substack{S, T \subseteq [n] \\ |S| > d \\ |T| \leq d}} \widehat{\mu}(S) x_S \widehat{f}(T) x_T \right) = \sum_{\substack{S, T \subseteq [n] \\ |S| > d \\ |T| \leq d}} \widehat{\mu}(S) \widehat{f}(T) \sum_{x \in \{-1,1\}^n} x_S \cdot x_T$$

Now, since $|S| > d \geq |T|$, $S \neq T$. But then by Lemma 1.8, $\sum_{x \in \{-1,1\}^n} x_S \cdot x_T = 0$, ie:- each of the summands in the above summation is zero. Consequently, $\widetilde{\mathbb{E}}_{\mu''}[f] = 0$, as desired. ■

1.3. Duality between SoS certificates and pseudo-distributions

As promised earlier, we are now ready to establish the duality between SoS certificates and pseudo-distributions.

Theorem 1.10. For any $f \in \mathbb{R}^{\{-1,1\}^n}$ and any even natural number d, f has a degree d SoS certificate if and only if for *every* degree d pseudo-distribution $\mu \in \mathbb{R}^{\{-1,1\}^n}$, $\widetilde{\mathbb{E}}_\mu[f] \geq 0$.

Proof. If f has a degree d SoS certificate, then $f = \sum_{i=1}^r g_i^2$, with $\deg(g_i) \leq \frac{d}{2}$ for every $i \in [r]$. Now, for any p.d. μ, $\widetilde{\mathbb{E}}_\mu[f] = \sum_{i=1}^r \widetilde{\mathbb{E}}_\mu[g_i^2]$. But $\widetilde{\mathbb{E}}_\mu[g_i^2] \geq 0, i \in [r]$ by the very definition of degree d pseudo-distributions.
Conversely, suppose f doesn't have a degree d SoS certificate. Now, by Lemma 1.5, SoS_d is a closed convex cone, and, f lies outside that closed convex set. Then by Corollary 0.16, there exists some μ such that $\langle \mu, f \rangle < 0 \leq \langle \mu, g \rangle$ for every $g \in \mathrm{SoS}_d$.
Thus, if we can show that μ is a pseudo-distribution, then we're done. To that extent, note that by Lemma 1.2, there exists some $L > 0$ such that $L + f \in \mathrm{SoS}_d \implies \langle \mu, L + f \rangle \geq 0$. But

$$\langle \mu, L + f \rangle \geq 0 \implies \langle \mu, L \rangle \geq -\langle \mu, f \rangle > 0 \implies \langle \mu, L \rangle > 0 \implies \langle \mu, 1 \rangle > 0 \implies \widetilde{\mathbb{E}}_\mu[1] > 0$$

Thus WLOG we can rescale μ by $\frac{1}{\widetilde{\mathbb{E}}_\mu[1]}$ and suppose that $\widetilde{\mathbb{E}}_\mu[1] = 1$.
Finally, also note that $\widetilde{\mathbb{E}}_\mu[g^2] \geq 0$ for every g such that $\deg(g) \leq \frac{d}{2}$: Indeed, if $\deg(g) \leq \frac{d}{2}$, then $g^2 \in \mathrm{SoS}_d$ since g^2 has g as its degree d SoS certificate.
Thus μ is a pseudo-distribution such that $\langle \mu, f \rangle = \widetilde{\mathbb{E}}_\mu[f] < 0$, which proves our result. ■

Remark. Note that there are two reasons why some $f \in \mathbb{R}^{\{-1,1\}^n}$ may not have a degree d certificate: Either f is negative for some $x \in \{-1,1\}^n$, or f is non-negative everywhere yet doesn't have a small degree certificate for its non-negativity. In the former case, it is easy to find a pseudo-distribution that makes the expectation of f negative: We can simply take $\mu = \mathbb{1}_x$, and then $\widetilde{\mathbb{E}}_\mu[f] = f(x) < 0$. In fact, note that $\mathbb{1}_x$ is a *proper* probability distribution.
However, if f is a non-negative function, then no proper distribution can make the expectation of f negative: it is here that the true power of pseudo-distributions is utilized, where we make some entries of μ negative to make the expectation of f negative. Note that we can't do the trivial thing and make every entry of μ negative, since we still have to satisfy $\sum_{y \in \{-1,1\}^n} \mu(y) = 1$ by the definition of pseudo-distributions.

§2. Algorithmic Issues

We finally begin discussing the algorithmic issues regarding SoS certificates.
Right at the outset, one can observe a problem with adapting the SoS machinery to an algorithmic framework: Note that throughout the last chapter, we worked with $\mathbb{R}$ as the co-domain of our function. However, we obviously can't represent reals with infinite precision in any Turing machine. Thus, we are forced to restrict our domain to rationals. However, this gives rise to new issues: For example, [Sch13] showed that the polynomial

$$f(x) = f(x_1, x_2, x_3) = x_1^4 + x_1x_2^3 + x_2^4 - 3x_1^2x_2x_3 - 4x_1x_2^2x_3 + 2x_1^2x_3^2 + x_1x_3^3 + x_2x_3^3 + x_3^4 \in \mathbb{Q}[x]$$

was expressible as a sum of squares of *real*-coefficient polynomials, but not *rational*-coefficient polynomials.
Thus there exist SoS polynomials $f : \mathbb{R}^n \mapsto \mathbb{R}$ with no rational coefficient certificates. For the boolean hypercube, the situation is unknown, ie:- it is *not known* if there exists some SoS polynomial $f : \{-1, 1\}^n \mapsto \mathbb{Q}$ with no rational coefficient certificates.
However, we shall soon see a technique to (almost) bypass this issue. Before that though, we set down some ground rules for dealing with the SoS framework algorithmically.

2.1. Algorithmic Ground Rules

Thus, we set down some conventions for SoS algorithms as follows:

Convention 1. Let $f = \sum_{i=1}^{r} g_i^2$ be a SoS formula, where $f : \{-1, 1\}^n \mapsto \mathbb{R}$ and $g_i : \{-1, 1\}^n \mapsto \mathbb{R}, i \in [r]$ are polynomials.
Whenever we are talking of SoS certificates in *an algorithmic context*, we shall assume that the number of polynomials in the certificate, denoted by 'r', is at most $(n+1)^{\frac{d}{2}}$ (this assumption is justified in light of Corollary 1.4).
Furthermore, we shall assume that we have been given our input polynomial f in its multilinear form. Also, every coefficient of f will be assumed to be rational.
A rational number will be represented as a pair of co-prime integers, each of $\text{poly}(n^d)$ bits. Consequently, the magnitude of our rational number will be bounded above by $2^{\text{poly}(n^d)}$.
Finally, we also assume that f has $\text{poly}(n^d)$ non-zero coefficients.
All the assumptions stated above for f also hold for each $g_i, i \in [r]$: Thus we assume that g_i's are rational multilinear polynomials, with at most $\text{poly}(n^d)$ non-zero coefficients, and each coefficient is expressible in $\text{poly}(n^d)$ bits.

As pointed out above, even SoS polynomials may not have a certificate satisfying our algorithmic conventions above. However, the techniques of this chapter will establish that for every degree d SoS polynomial, one can make a very small perturbation of that polynomial to get another degree d SoS polynomial possessing a certificate satisfying all of the conventions laid down above.
This more or less fixes the problem of fitting the SoS framework in a discrete Turing machine setting, with the minor drawback that most SoS algorithms will usually achieve their desired goal upto some 'ε' slack, where ε is a memorial to the small perturbation we made to satisfy our algorithmic conventions.
Also, note another pedagogical advantage to the whole perturbation business: We can now freely prove all our results in the real domain, and finally, when we want to use that theorem to design some algorithm, we can simply perturb the results of our theorem slightly and get away with it.

2.2. SoS certificates are efficiently verifiable

Theorem 2.1. Let f be a polynomial satisfying Convention 1, and let $(g_1, \ldots, g_r)$ be a purported degree d SoS certificate for f, also satisfying Convention 1. Then the validity of this certificate can be verified in $\text{poly}(n^d)$ time.

Proof. Since the size of g_i is $\text{poly}(n^d)$, g_i^2 can be calculated in $\text{poly}(n^d)$ time. Similarly, since $r = \text{poly}(n^d)$, $\sum_{i=1}^r g_i^2$ can also be calculated in $\text{poly}(n^d)$ time. Furthermore, $\sum_{i=1}^r g_i^2$ has $\text{poly}(n^d)$ non-zero coefficients, all of which are rational.
Now, $\sum_{i=1}^r g_i^2$ may not be a multilinear polynomial: However, performing the multilinear reduction (see Lemma 0.12) of $\sum_{i=1}^r g_i^2$ takes $\text{poly}(n^d)$ time, so WLOG we assume that $\sum_{i=1}^r g_i^2$ is a multilinear polynomial.
By Theorem 0.10, since the Fourier decomposition of any function is unique, and since multilinear polynomials are their own Fourier decomposition, $\sum_{i=1}^r g_i^2$ equals f if and only if all coefficients of the two polynomials are the same (keep in mind that f has been given to us as a multilinear polynomial). Since both $\sum_{i=1}^r g_i^2$ and f have $\text{poly}(n^d)$ non-zero coefficients, one can compare their coefficients in $\text{poly}(n^d)$ time, and hence verify the correctness of the certificate.
It is clear that the entire verification process described above can be carried out in $\text{poly}(n^d)$ time, as desired. ■

2.3. Re-examining Theorem 1.3

Definition 2.1. Suppose we have some set E, and suppose we have two tuples $S, T \in E^n$. Let $S \uplus T$ be the multiset union of S and T. For example, if $E = \{e_1, e_2, e_3\}$, and $n = 4$, and $S = (e_3, e_1, e_2, e_1), T = (e_2, e_2, e_1, e_3)$, then $S \uplus T = \{e_1, e_1, e_1, e_2, e_2, e_2, e_3, e_3\}$.
Then define the multilinear reduction of S and T to be:

$$\nu(S, T) := \{e \in E : e \text{ occurs an odd number of times in } S \uplus T\}$$

For the S and T given above, we would have $\nu(S, T) = \{e_1, e_2\}$.

Lemma 2.2. Consider any $f \in \text{SoS}_d$, and let $f(x) = \langle (1, x)^{\otimes \frac{d}{2}}, A(1, x)^{\otimes \frac{d}{2}} \rangle$ for some PSD matrix A. Then for any $U \subseteq [n]$ such that $|U| \leq d$, we have

$$\widehat{f}(U) = \sum_{\substack{S,T \subseteq \{0,1,\ldots,n\}^d \\ \nu(S,T)\setminus\{0\} = U}} A_{S,T}$$

Proof. Recall that $(1, x)^{\otimes \frac{d}{2}}$ was indexed by elements of $\{0, 1, \ldots, n\}^d$: For example, the tuple $(2, 0, 1, 0) \in \{0, 1, 2\}^4$ would correspond to $x_2 \cdot 1 \cdot x_1 \cdot 1$. Consequently, A is indexed by pairs of tuples, say S and T, in $\{0, 1, \ldots, n\}^d$.
Now,

$$((1, x)^{\otimes \frac{d}{2}})^\intercal A (1, x)^{\otimes \frac{d}{2}} = \sum_{S,T \subseteq \{0,1,\ldots,n\}^d} x_S A_{S,T} x_T$$

As usual, we perform a multilinear reduction on $x_S \cdot x_T$: For example, say $S = (2, 1, 0, 1)$ and $T = (0, 2, 0, 2)$, ie:- $x_S = x_2 \cdot x_1 \cdot 1 \cdot x_1 = x_2 x_1^2$ and $x_T = 1 \cdot x_2 \cdot 1 \cdot x_2 = x_2^2$. Then note that x_1 occurs twice in $x_S \cdot x_T$, while x_2 occurs thrice. Consequently, x_1 would get annihilated, while one copy of x_2 would remain. In fact, the only elements in $\{x_1, \ldots, x_n\}$ which will survive in $x_S \cdot x_T$ are the ones whose indices are in $\nu(S, T)$.
For the example here, $\nu\big((2, 1, 0, 1), (0, 2, 0, 2)\big) = \{0, 2\}$. Since the zeroth index corresponds to 1, we don't care about it, and thus $x_S \cdot x_T$ would only contribute towards $\widehat{f}(\{2\})$.
Thus, performing the multilinear reduction of $\sum_{S,T \subseteq \{0,1,\ldots,n\}^d} x_S A_{S,T} x_T$, and equating it to $\sum_{U \subseteq [n]} \widehat{f}(U) x_U$, yields the desired identity. ■

Remark. Note that the only reason we have imposed the $|U| \leq d$ condition in the lemma is, since f is equivalent to a degree d polynomial, all the Fourier coefficients of f corresponding to sets of size greater than d is 0, so we don't bother about them.

Corollary 2.3.

$$\widehat{f}(\varnothing) = \sum_{S \subseteq \{0,1,\dots,n\}^d} A_{S,S} = \mathrm{tr}(A)$$

Theorem 2.4. Let $f = \sum_{i=1}^r g_i^2$ be a degree d SoS formula satisfying Convention 1. Then $f(x) = ((1,x)^{\otimes \frac{d}{2}})^\mathsf{T} A (1,x)^{\otimes \frac{d}{2}}$, where A is a PSD matrix, all of whose entries are rational and bounded by $2^{\mathrm{poly}(n^d)}$. Thus, A is an alternative $\mathrm{poly}(n^d)$ space representation of our SoS certificate.

Proof. By Corollary 0.6 and Corollary 2.3,

$$\|A\|_F \leq \mathrm{tr}(A) = \widehat{f}(\varnothing)$$

By Convention 1, $\widehat{f}(\varnothing) = \mathrm{poly}(n^d)$, and thus $\|A\|_F$ is $\mathrm{poly}(n^d)$.
Now, viewing A from the point of view of the proof of Theorem 1.3, we also see that the entries of A must be rational. Thus combining the two points of view yields that all entries of A are rational numbers expressible in $\mathrm{poly}(n^d)$ bits, as desired. ■

2.4. Finding SoS certificates

Suppose we are given some f and asked to find a degree d SoS certificate for it. By Theorem 1.3, we may as well find a PSD matrix A such that $f(x) = \langle (1,x)^{\otimes \frac{d}{2}}, A(1,x)^{\otimes \frac{d}{2}} \rangle$.
Now, by Lemma 2.2, the entries of A satisfy some linear relations. For example, if $n = 2, d = 2$, then $A \in \mathbb{R}^{(2+1)^{\frac{2}{2}} \times (2+1)^{\frac{2}{2}}} = \mathbb{R}^{3\times 3}$ satisfies the following relations:

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} \\ a_{1,0} & a_{1,1} & a_{1,2} \\ a_{2,0} & a_{2,1} & a_{2,2} \end{bmatrix}$$

$$\begin{aligned} a_{0,0} + a_{1,1} + a_{2,2} &= \widehat{f}(\varnothing) \\ a_{0,1} + a_{1,0} &= \widehat{f}(\{1\}) \\ a_{0,2} + a_{2,0} &= \widehat{f}(\{2\}) \\ a_{1,2} + a_{2,1} &= \widehat{f}(\{1,2\}) \end{aligned}$$

Thus, for every $U \subseteq [n], |U| \leq d$, we write down a linear relation involving the entries of A, and express it in terms of matrix inner products (see the remark after Definition 0.3). Thus, the set of candidate PSD matrices is

$$\mathcal{A}' := \{A \in \mathbb{R}^{(n+1)^{\frac{d}{2}} \times (n+1)^{\frac{d}{2}}} : A \succcurlyeq 0, \langle C_U, A \rangle = b_U\}$$

where the $\langle C_U, A \rangle = b_U$ condition encodes the equation $\widehat{f}(U) = \sum_{\substack{S,T \subseteq \{0,1,\dots,n\}^d \\ \nu(S,T)\setminus\{0\} = U}} A_{S,T}$.
Now, we ultimately want to apply Theorem 0.17 to $\mathcal{A}'$: However, note that $\mathcal{A}'$ doesn't contain any balls inside it, because if any matrix in $\mathcal{A}'$ is perturbed slightly, some linear equality will be violated.
Thus, in order to apply Theorem 0.17, we must consider the following "thickening" of $\mathcal{A}'$:

$$\mathcal{A} := \{A \in \mathbb{R}^{(n+1)^{\frac{d}{2}} \times (n+1)^{\frac{d}{2}}} : A \succcurlyeq 0, \langle C_U, A \rangle \in [b_U - \varepsilon, b_U + \varepsilon]\}$$

Fortunately, $\mathcal{A}$ is a closed convex bounded set containing a ball inside it [4], and the aspect ratio of $\mathcal{A}$ is $\text{poly}\left(n^d + \frac{1}{\varepsilon}\right)$ [5]. Thus Theorem 0.17 becomes applicable [6], and we can find an $\mathcal{A}$ satisfying the aforementioned SDP.
Equivalently, in $\text{poly}\left(n^d + \frac{1}{\varepsilon}\right)$ time we have found out coefficients $\widehat{f'}(U)$ such that $\widehat{f'}(U) = \sum_{\substack{S,T \subseteq \{0,1,\dots,n\}^d \\ \nu(S,T)\setminus\{0\}=U}} A_{S,T} + \delta_U$, where $|\delta_U| \leq \varepsilon$.
We have thus effectively found out a polynomial f' such that f' has a degree d SoS certificate (which we also found out as a result of solving the PSD program), and $|\widehat{f}(U) - \widehat{f'}(U)| \leq \varepsilon$ for all U such that $|U| \leq d$.
Thus if we define

$$L := \sum_{\substack{U \subseteq \{0,1,\dots,n\} \\ |U| \leq d}} |\widehat{f}(U) - \widehat{f'}(U)|$$

then

$$L \leq \varepsilon \binom{n+1}{d} \leq \varepsilon (n+1)^d$$

Now, recall from the proof of Lemma 1.2 the fact that $|\alpha| + \alpha x_S$ has a degree d SoS certificate for any $\alpha \in \mathbb{R}$ and any $S \subseteq [n]$ such that $|S| \leq d$.
Thus $L + f - f'$ has a degree d SoS certificate. Since f' has a degree d SoS certificate too, we get that $(L + f - f') + f' = L + f$ has a degree d SoS certificate. Furthermore, the certificate for $L + f$ is efficiently evaluable: Indeed, we already have the certificate for f' at hand, and the certificates for $|\widehat{f}(U) - \widehat{f'}(U)| + \widehat{f}(U)x_U - \widehat{f'}(U)x_U$ can be constructed in $\mathcal{O}(1)$ time for each U, using the process described in the proof of Lemma 1.2. Thus in $\text{poly}(n^d)$ time we can construct a degree d SoS certificate for $L + f$, where $L \leq \varepsilon\,\text{poly}(n^d)$.
Conversely, suppose our function f does *not* have a degree d SoS certificate. In that case, we would be interested in finding a degree d pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[f] < 0$: Once again, we can't solve for our solution exactly, but what we can do is as follows: Let our set of variables be $v_U := \widetilde{\mathbb{E}}_\mu[x_U]$ for all $U \subseteq \{0, 1, \dots, n\}, |U| \leq d$. From the definition of a degree d pseudo-distribution, we know that $\widetilde{\mathbb{E}}_\mu[1] = 1$ (which translates to the equation $v_\varnothing = 1$), and $\widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\intercal] \succcurlyeq 0$, which translates to the fact that a matrix composed of the variables v_U is positive semi-definite.
Thus, we once again have a SDP over the variables $\{v_U : U \subseteq \{0, 1, \dots, n\}, |U| \leq d\}$, which we solve upto some slack ε (ie:- our SDP returns some μ such that $\widetilde{\mathbb{E}}_\mu[f] \leq \varepsilon$) in $\text{poly}\left(n^d, \frac{1}{\varepsilon}\right)$ time.
Thus, summarizing the paragraph above, we see that we have managed to achieve what we promised in Section 2.1, which we now state as a theorem.

Theorem 2.5. Let $f : \{-1, 1\}^n \mapsto \mathbb{R}$ be a function complying with Convention 1. Given a d and a $\varepsilon > 0$, in $\text{poly}\left(n^d, \frac{1}{\varepsilon}\right)$ time, we can:

1. Either find a degree d SoS certificate for $f + L$, where $L = \varepsilon\,\text{poly}(n^d)$,
2. Or we can find a degree d pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[f] \leq \varepsilon$.

2.5. Another Useful Application of SDP solvers

Before we conclude this chapter, a very interesting consequence of Theorem 0.17 ought to be mentioned (once again, without proof).

[4] the ball is of radius $\mathcal{O}\left(\left(\frac{\varepsilon}{\mathcal{L}}\right)^n\right)$, where $\mathcal{L} = \sqrt{\sum_U \|C_U\|_F^2}$
[5] Note that the norm of any matrix $A \in \mathcal{A}$ is $\text{poly}(n^d)$ by Theorem 2.4, and thus $\mathcal{A}$ lies within a ball of radius $\text{poly}(n^d)$
[6] the application of the theorem is not very straightforward, the details are involved, and unnecessary for the larger discussion, so we skip it

Theorem 2.6. Consider any $f : \{-1,1\}^n \mapsto \mathbb{R}$. Let opt be the largest possible pseudo-expectation $\widetilde{\mathbb{E}}_\mu[f]$ of degree d, ie:-

$$\mathrm{opt} := \max_{\mu \text{ is a degree } d \text{ pseudo-distribution}} \widetilde{\mathbb{E}}_\mu[f]$$

Then for any $\varepsilon > 0$, in $\mathrm{poly}\left(n^d, \frac{1}{\varepsilon}\right)$ time we can calculate a degree d pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[f] \geq \mathrm{opt} - \varepsilon$. Note that by taking the negative of our function, we can also *minimize* the pseudo-expectation of a function, ie:- if $\mathrm{opt}' := \min_{\mu \text{ is a degree } d \text{ pseudo-distribution}} \widetilde{\mathbb{E}}_\mu[f]$, then for any $\varepsilon > 0$, in $\mathrm{poly}\left(n^d, \frac{1}{\varepsilon}\right)$ time we can calculate a degree d pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[f] \leq \mathrm{opt}' + \varepsilon$.

Remark. Note that when we say that some algorithm gives to us a degree d pseudo-distribution μ, we always assume (in light of Lemma 1.9) that we have been given μ in form of a multilinear polynomial of degree d.
Having our pseudo-distribution in the multilinear polynomial format has another advantage: It makes the *moment matrix* $\widetilde{\mathbb{E}}_\mu[(1,x)^{\otimes \frac{d}{2}}((1,x)^{\otimes \frac{d}{2}})^\intercal]$ computable in $\mathrm{poly}(n^d)$ time. In fact, as we shall see later, we are usually more interested in the moment matrix of μ than in μ itself.

§3. The Max-Cut Problem

So far, we have seen the SoS framework in an abstract sense. We shall slowly see how it relates to theoretical computer science as a whole.
Our first application of the SoS framework will be in getting an approximation algorithm for the so-called Max-Cut problem.

Problem (Max-Cut Problem)**.** Let $G = G(V, E)$ be a simple undirected graph. For any $S \subseteq V$, we define $E_S := \{\{s, t\} \in E : s \in S, t \notin S\}$ to be the cut induced by the subset S.
The max-cut problem asks us the largest value of $|E_S|$ over all possible $S \subseteq V$.

Now, the Max-Cut problem is known to be NP-hard. Thus, we start looking for approximation algorithms for the same.
A very simple approximation algorithm goes as follows (refer to [Max23] for more details): Choose the set S randomly, by choosing each vertex $v \in V$ to be belonging in S with probability $\frac{1}{2}$. In expectation, $|E_S| = \frac{|E|}{2}$. Since the size of the maximum cut is at most $|E|$, the approximation factor of this algorithm is $\frac{1}{2}$. This algorithm can be derandomized to yield a deterministic algorithm too.
For multiple decades, $\frac{1}{2}$ stood to be the best approximation factor one could achieve. Then, in 1993, Goemans and Williamson [GW93] improved the approximation factor to 0.87856, and then in 2004, Khot, Kindler, Mossel, and O'Donnell [KKMO04] showed that this was the best approximation factor achievable in polynomial time, conditional on some well-known hardness conjectures such as the Unique Games Conjecture.
In this chapter, we shall explore Goemans and Williamson's algorithm, from a "SoS" point of view.
Before that, we develop some probabilistic machinery we'll need later on.

3.1. The Gaussian Sampling Lemma

Lemma 3.1 (Gaussian Sampling Lemma)**.** For any degree 2 pseudo-distribution $\mu : \{-1, 1\}^n \mapsto \mathbb{R}$, there exists an actual distribution $\nu : \mathbb{R}^n \mapsto \mathbb{R}$ such that:

1. $\widetilde{\mathbb{E}}_\mu[x] = \mathbb{E}_\nu[x]$, ie:- the first moments of μ and ν are the same.
2. $\widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}] = \mathbb{E}_\nu[xx^\mathsf{T}]$, ie:- the second moments of μ and ν are the same.

Keep in mind that $x = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^\mathsf{T}$, and as usual, the expectations of vectors and matrices are taken elementwise. Furthermore, when we take expectation w.r.t μ, x varies over $\{-1, 1\}^n$, while when we take expectation w.r.t ν, x varies over $\mathbb{R}^n$.

Proof. First of all, note that $M := \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}] \succcurlyeq 0$: Indeed, consider any $v \in \mathbb{R}^n$. Then

$$v^\mathsf{T} M v = v^\mathsf{T} \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}] v = \widetilde{\mathbb{E}}_\mu[v^\mathsf{T} x x^\mathsf{T} v] = \widetilde{\mathbb{E}}_\mu\left[\langle x, v\rangle^2\right]$$

Now, $\langle x, v\rangle$ is a linear polynomial in $x_1, \dots, x_n$, and consequently, $\langle x, v\rangle^2$ is a degree 2 SoS polynomial. Since μ is a degree 2 p.d., $\widetilde{\mathbb{E}}_\mu\left[\langle x, v\rangle^2\right] \geq 0$. Consequently, for every $v \in \mathbb{R}^n$, $v^\mathsf{T} M v \geq 0$, implying that M is PSD, as desired.
Finally, note that we can simply choose $\nu = \mathcal{N}(\widetilde{\mathbb{E}}_\mu[x], \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}])$, ie:- we can choose ν to be the Gaussian distribution with the specified parameters, which by its definition will have the desired first and second moments. Note that the covariance matrix of a Gaussian is PSD, which is why we needed to verify first that $\widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}]$ is indeed a PSD matrix. ■

Remark. Both the first and second-moment conditions can be merged and expressed as $\widetilde{\mathbb{E}}_\mu[(1, x)(1, x)^\mathsf{T}] = \mathbb{E}_\nu[(1, x)(1, x)^\mathsf{T}]$.

3.2. SoS formulation

Let our graph be $G = G(V, E)$, which has n vertices. Construct the polynomial $f_G : \{-1, 1\}^n \mapsto \mathbb{R}$, where

$$f_G(x) = f_G(x_1, \dots, x_n) := \frac{1}{4} \sum_{\{i,j\} \in E} (x_i - x_j)^2$$

It is clear that if we set $x_i = 1$ for every $i \in S$, and $x_j = -1$ for every $j \notin S$, then $f_G(x)$ gives us the size of the cut induced by S. Thus maximizing f_G over the boolean hypercube gives us the maximum cut over G.

Lemma 3.2 (Gaussian Rounding)**.** Let μ be degree 2 pseudo-distribution on $\{-1, 1\}^n$ with zero mean, ie:- $\widetilde{\mathbb{E}}_\mu[x] = 0$. Then there exists a probability distribution μ', also on $\{-1, 1\}^n$, such that

$$\mathbb{E}_{\mu'}[f_G(x)] \geq \alpha_{GW} \widetilde{\mathbb{E}}_\mu[f_G(x)]$$

where $\alpha_{GW} := \min_{\rho \in [-1,1]} \frac{2 \arccos(\rho)}{\pi(1-\rho)} \approx 0.878$.

Proof. Consider the normal distribution $\mathcal{G} := \mathcal{N}(\widetilde{\mathbb{E}}_\mu[x], \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}]) = \mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}])$ as in Lemma 3.1, and construct the distribution μ' as follows:

1. Sample $g \sim \mathcal{G}$, where $g = \begin{bmatrix} g_1 & g_2 & \dots & g_n \end{bmatrix}^\mathsf{T} \in \mathbb{R}^n$.
2. For every $i \in [n]$, set $\widehat{x_i} := \text{sign}(g_i)$.

Then $\widehat{x} := \begin{bmatrix} \widehat{x_1} & \widehat{x_2} & \dots & \widehat{x_n} \end{bmatrix}^\mathsf{T} \in \{-1, 1\}^n$ is a random vector in $\{-1, 1\}^n$ with probability distribution denoted by μ'.
Now,

$$\mathbb{E}_{\mu'}[f_G(x)] = \frac{1}{4} \mathbb{E}_{\mu'} \left[\sum_{\{i,j\} \in E} (\widehat{x_i} - \widehat{x_j})^2 \right] = \sum_{\{i,j\} \in E} \Pr(\text{sign}(g_i) \neq \text{sign}(g_j))$$

where the last equality follows from the fact that $(\widehat{x_i} - \widehat{x_j})^2$ is non-zero if and only if $\text{sign}(g_i) \neq \text{sign}(g_j)$.
Now, by Lemma 0.18, we have that

$$\Pr(\text{sign}(g_i) \neq \text{sign}(g_j)) \overset{\text{Lemma 0.18}}{=} \frac{\arccos(\mathbb{E}[g_i g_j])}{\pi} \geq \alpha_{GW} \frac{1 - \mathbb{E}[g_i g_j]}{2}$$

Now, by Lemma 3.1, $\mathbb{E}[g_i g_j] = \widetilde{\mathbb{E}}_\mu[x_i x_j]$, and thus

$$\mathbb{E}_{\mu'}[f_G(x)] = \sum_{\{i,j\} \in E} \Pr(\text{sign}(g_i) \neq \text{sign}(g_j)) \geq \alpha_{GW} \sum_{\{i,j\} \in E} \frac{1 - \widetilde{\mathbb{E}}_\mu[x_i x_j]}{2}$$

Finally, note that

$$\widetilde{\mathbb{E}}_\mu[f_G(x)] = \frac{1}{4} \widetilde{\mathbb{E}}_\mu \left[\sum_{\{i,j\} \in E} (x_i - x_j)^2 \right] = \frac{1}{4} \sum_{\{i,j\} \in E} \widetilde{\mathbb{E}}_\mu \left[x_i^2 + x_j^2 - 2x_i x_j \right] = \frac{1}{4} \sum_{\{i,j\} \in E} \widetilde{\mathbb{E}}_\mu \left[2 - 2x_i x_j \right]$$

$$= \frac{1}{2} \sum_{\{i,j\} \in E} \widetilde{\mathbb{E}}_\mu \left[1 - x_i x_j \right] = \sum_{\{i,j\} \in E} \frac{1 - \widetilde{\mathbb{E}}_\mu \left[x_i x_j \right]}{2}$$

Thus $\mathbb{E}_{\mu'}[f_G(x)] \geq \alpha_{GW} \widetilde{\mathbb{E}}_\mu[f_G(x)]$ holds, as desired. ∎

Remark. A few remarks are in order:

1. The reason why we took the mean of our pseudo-distribution to be 0 goes as follows: Note that
$$\widetilde{\mathbb{E}}_\mu[f_G(x)] = \frac{1}{2}\sum_{\{i,j\}\in E}(1 - \widetilde{\mathbb{E}}_\mu\left[x_i x_j\right])$$
Thus, nowhere in the expression for $\widetilde{\mathbb{E}}_\mu[f_G(x)]$ is the first moment of μ coming into play: Only second-order terms such as $x_i x_j$ are involved.
Thus if we replace $\mu(x)$ by $\frac{\mu(x)+\mu(-x)}{2}$, then μ continues to remain a pseudo-distribution, and moreover, $\widetilde{\mathbb{E}}_\mu[f_G]$ doesn't change.
Thus WLOG we can take the mean of μ to be 0.
2. Note that the distribution μ' is *efficiently sampleable*, ie:- samples from the distribution μ' can be drawn in $\text{poly}(n)$ time: Indeed, it is well known that a normal distribution is efficiently sampleable, and consequently the sampleability of μ' follows.

We finally arrive at the moment of truth, where we establish that an algorithm with approximation factor α_{GW} is achievable.

Theorem 3.3. Let $G = G(V, E)$ be *any* simple undirected graph. Denote by β_G the value of the largest cut of G. Then $\frac{\beta_G}{\alpha_{GW}} - f_G(x)$ has a degree 2 SoS certificate.

Proof. Assume for the sake of contradiction that $\frac{\beta_G}{\alpha_{GW}} - f_G(x)$ is not in SoS_2. Then by Theorem 1.10, there exists a p.d. μ such that
$$\widetilde{\mathbb{E}}_\mu\left[\frac{\beta_G}{\alpha_{GW}} - f_G(x)\right] < 0 \implies \widetilde{\mathbb{E}}_\mu[f_G(x)] > \frac{\beta_G}{\alpha_{GW}}$$
Now, by Lemma 3.2, there exists a probability distribution μ' such that $\mathbb{E}_{\mu'}[f_G(x)] \geq \alpha_{GW}\widetilde{\mathbb{E}}_\mu[f_G(x)] \implies \mathbb{E}_{\mu'}[f_G(x)] > \beta_G$. But then
$$\beta_G = \max_{x\in\{-1,1\}^n} f_G(x) \geq \mathbb{E}_{\mu'}[f_G(x)] > \beta_G$$
leading to a contradiction. ∎

3.3. Tying everything up

By this point, most of the work has been done. We just need to invoke the theorems developed in the correct order. Define
$$\text{opt}_{\text{SoS}_2} := \max_{\mu \text{ is a degree 2 pseudo-distribution}} \widetilde{\mathbb{E}}_\mu[f_G(x)]$$
By Theorem 2.6, we can find a pseudo-distribution μ of degree 2 (in $\text{poly}(n, 1/\varepsilon)$ time) such that $\widetilde{\mathbb{E}}_\mu[f] \geq \text{opt}_{\text{SoS}_2} - \varepsilon$.
By Lemma 3.2, we can then find a distribution μ' such that $\mathbb{E}_{\mu'}[f] \geq \alpha_{GW}\widetilde{\mathbb{E}}_\mu[f] \geq \alpha_{GW}(\text{opt}_{\text{SoS}_2} - \varepsilon)$.
Now, suppose f_G attains its maximum at x^*. Then the distribution $\mathbb{1}_{x^*}$ is also a degree 2 pseudo-distribution, and thus $\text{opt}_{\text{SoS}_2} \geq \beta_G$, where recall that $\beta_G = f_G(x^*)$ was the largest cut of G.
Thus $\mathbb{E}_{\mu'}[f] \geq \alpha_{GW}(\beta_G - \varepsilon) \geq (\alpha_{GW} - \varepsilon)\beta_G$.
Thus our $(\alpha_{GW} - \varepsilon)$-approximation algorithm goes as follows:

1. Compute a pseudo-distribution μ of degree 2 (in $\text{poly}(n, 1/\varepsilon)$ time) such that $\widetilde{\mathbb{E}}_\mu[f] \geq \text{opt}_{\text{SoS}_2} - \varepsilon$.
2. Let μ' be the distribution generated by the Gaussian rounding of μ. Sample a point from μ'. For example, suppose the point chosen is $(-1, 1, 1, -1, -1, 1)$. Then our cut-set $S = \{2, 3, 6\} \subseteq [6]$. Let β be the size of the cut induced by S.
In expectation, β is atleast $(\alpha_{GW} - \varepsilon)$ times the optimal max-cut.

Note that the above algorithm is a randomized algorithm.

3.4. A Further Look into the Goemans-Williamson algorithm

It is in the very nature of research to ask if the current state of the art can be improved upon.
Now, as mentioned earlier, due to the [KKMO04] paper (which proved, assuming the Unique Games Conjecture, that approximating Max-Cut by a factor of $(\alpha_{GW} + \varepsilon)$ is NP-hard), there are very plausible reasons to believe that α_{GW} is the best we can get.
However, there is no *unconditional* proof that α_{GW} indeed is the ceiling. For example, it is unknown if we can outperform this bound using higher degree SoS's, such as SoS_4, or even $\mathrm{SoS}_{\log n}$.
Fortunately, though, it can be shown that one can't use SoS_2 to outperform the Goemans-Williamson algorithm, ie:- if we are only allowed to solve instances of SoS_2, then the Goemans-Williamson algorithm is the best we can get.
Let's prove the above assertion. Before that, we first prove a lemma characterizing the performance of the GW algorithm based on the size of the output.

Lemma 3.4. Let G be a graph such that $\beta_G = (1-\delta)|E|$. Let μ' be the distribution that the GW algorithm gives us. Then $\mathbb{E}_{\mu'}[f_G(x)] \geq (1 - \sqrt{\delta})|E|$.

Proof. Define

$$h_G(x) := |E| - f_G(x) = \frac{1}{2} \sum_{\{i,j\} \in E} (1 + x_i x_j)$$

Let μ be a degree 2 p.d. such that $\widetilde{\mathbb{E}}_{\mu}[h_G] = \delta|E|$. Let μ' be the distribution generated from μ as described in Lemma 3.2. Then we must show that $\mathbb{E}_{\mu'}[h_G] \leq \sqrt{\delta}|E|$.
Define $g, \widehat{x}$ as in the proof of Lemma 3.2. Then

$$\mathbb{E}_{\mu'}[h_G] = \mathbb{E}_{\mu'} \left[\frac{1}{2} \sum_{\{i,j\} \in E} (1 + \widehat{x}_i \widehat{x}_j) \right]$$

Given the definition of h_G, it is not difficult to see that

$$\mathbb{E}_{\mu'}[h_G] = \sum_{\{i,j\} \in E} 1 - \frac{1 - \widehat{x}_i \widehat{x}_j}{2} = \sum_{\{i,j\} \in E} 1 - \Pr(\mathrm{sign}(g_i) \neq \mathrm{sign}(g_j)) = \sum_{\{i,j\} \in E} 1 - \frac{\arccos(\mathbb{E}[g_i g_j])}{\pi}$$

$$= \frac{1}{2} \sum_{\{i,j\} \in E} 1 + \frac{2}{\pi} \arcsin(\mathbb{E}[g_i g_j])$$

Finally, note that

$$\sup_{\rho \in [-1,1]} \frac{\left(1 + \frac{2}{\pi} \arcsin(\rho)\right)^2}{1 + \rho} = 2 \tag{3.1}$$

Thus

$$\mathbb{E}_{\mu'}[h_G]^2 = \frac{1}{4} \left(\sum_{\{i,j\} \in E} 1 + \frac{2}{\pi} \arcsin(\mathbb{E}[g_i g_j]) \right)^2 \overset{\text{Cauchy-Schwartz}}{\leq} \frac{|E|}{4} \sum_{\{i,j\} \in E} \left(1 + \frac{2}{\pi} \arcsin(\mathbb{E}[g_i g_j]) \right)^2$$

$$\overset{\text{Eq. (3.1)}}{\leq} \frac{|E|}{2} \sum_{\{i,j\} \in E} 1 + \mathbb{E}[g_i g_j] = |E| \cdot \mathbb{E} \left[\sum_{\{i,j\} \in E} \frac{1 + g_i g_j}{2} \right] = |E| \cdot \widetilde{\mathbb{E}}_{\mu}[h_G] = \delta |E|^2$$

Thus $\mathbb{E}_{\mu'}[h_G]^2 \leq \delta |E|^2$, as desired. ∎

Remark. Note that this lemma effectively says that for $\frac{\beta_G}{|E|} = 1 - \delta$, we get an approximation algorithm of factor $\geq \frac{1-\sqrt{\delta}}{1-\delta} = \frac{1}{1+\sqrt{\delta}}$. For $\delta \leq 0.019$, $\frac{1}{1+\sqrt{\delta}} \geq \alpha_{GW}$, and thus in fact, this result is a *better analysis* of the Goemans-Williamson algorithm in the "low δ régime".

Now, we'll show the optimality of the GW algorithm by demonstrating a degree 2 pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu \left[\left(1 - \mathcal{O}\left(\frac{1}{n^2}\right)\right) |E_{C_n}| - f_{C_n}(x) \right] \leq 0$ [7], where C_n is the cycle on n vertices, where n is odd. Note that $\beta_{C_n} = n-1 = \left(1 - \frac{1}{n}\right)|E_{C_n}|$.
Consequently, in the light of Lemma 3.4, atleast upto constant factors [8], no degree 2 SoS certificate outperforms the Goemans-Williamson algorithm, *for all graphs*. Since we use the cycle graph to demonstrate this, we say that the cycle graph demonstrates an *integrality gap* in SoS_2.
Let L_{C_n} be a symmetric matrix such that $f_{C_n}(x) = \frac{1}{4} x^\mathsf{T} L_{C_n} x$ [9]. Now, the eigenvalue of L_G with maximum absolute value is equal to $\lambda := 4\cos^2\left(\frac{\pi}{2n}\right)$, and it has algebraic multiplicity 2. Consequently, we can choose two vectors v_1, v_2 from the eigenspace of λ such that $M := v_1 v_1^\mathsf{T} + v_2 v_2^\mathsf{T}$ has only ones on its diagonal. By Lemma 0.4, M is PSD too, and consequently there exists some degree 2 pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}] = M$. Then note that

$$\widetilde{\mathbb{E}}_\mu[f_{C_n}(x)] = \frac{1}{4}\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} L_{C_n} x] = \frac{1}{4}\widetilde{\mathbb{E}}_\mu\left[\langle L_{C_n}, xx^\mathsf{T}\rangle\right] = \frac{1}{4}\langle L_{C_n}, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}]\rangle = \frac{1}{4}\langle L_{C_n}, v_1 v_1^\mathsf{T} + v_2 v_2^\mathsf{T}\rangle$$

Now,

$$\langle L_{C_n}, v_1 v_1^\mathsf{T}\rangle = \mathrm{tr}(L_{C_n}^\mathsf{T} v_1 v_1^\mathsf{T}) = \mathrm{tr}(L_{C_n} v_1 v_1^\mathsf{T}) = \mathrm{tr}(\lambda v_1 v_1^\mathsf{T}) = \lambda\,\mathrm{tr}(v_1 v_1^\mathsf{T})$$

Thus

$$\widetilde{\mathbb{E}}_\mu[f_{C_n}(x)] = \frac{1}{4}\langle L_{C_n}, v_1 v_1^\mathsf{T} + v_2 v_2^\mathsf{T}\rangle = \frac{\lambda}{4}\mathrm{tr}(M) = \underbrace{\cos^2\left(\frac{\pi}{2n}\right)}_{=1-\mathcal{O}\left(\frac{1}{n^2}\right)} \cdot n = \left(1 - \mathcal{O}\left(\frac{1}{n^2}\right)\right) \cdot |E_{C_n}|$$

Consequently, $\widetilde{\mathbb{E}}_\mu \left[\left(1 - \mathcal{O}\left(\frac{1}{n^2}\right)\right) |E_{C_n}| - f_{C_n}(x) \right] = 0 \leq 0$, as desired.

[7] we will in fact construct a p.d. μ such that this quantity is exactly 0
[8] the cycle C_n can be viewed as the "discretization" of the 2-sphere, ie:- a circle. By considering the discretization of higher dimensional spheres, tightness of the GW algorithm, even in constant factors, can be established.
[9] this matrix is also known as the *Laplacian*. We shall study it in detail in the next chapter

§4. Quadratic Optimization over the Hypercube

Recall the 'f_G' polynomial from the last chapter:

$$f_G(x) := \frac{1}{4}\sum_{ij\in E}(x_i - x_j)^2$$

This may be also written as $\frac{1}{4}x^\mathsf{T} L_G x$, where L_G is known as the *Laplacian* matrix of the graph G. Note that $L_G = D_G - A_G$, where D_G is the diagonal matrix containing the degrees of vertices in G, while A_G is the adjacency matrix of G.
It is easy to see that L_G is PSD for any G. Now, that prompts us to ask the question: Given any $B \succcurlyeq 0$, how well can we approximate $\mathrm{opt}(B) := \max_{x\in\{-1,1\}^n} x^\mathsf{T} B x$?
This question was answered by the famous "$\frac{\pi}{2}$-theorem" of Nesterov, which we shall now prove.

Theorem 4.1 (Nesterov's Theorem)**.** Let $B \in \mathbb{R}^{n\times n}$ be PSD. Then $\frac{\pi}{2}\,\mathrm{opt}(B) - x^\mathsf{T} B x$ has a degree 2 SoS certificate. Consequently, for any $\varepsilon > 0$, there exists a $\left(\frac{2}{\pi} - \varepsilon\right)$-approximation algorithm for calculating $\max_{x\in\{-1,1\}^n} x^\mathsf{T} B x$, and this algorithm runs in $\mathrm{poly}\left(n, \frac{1}{\varepsilon}\right)$ time.

Proof. We will mimic the proof of Lemma 3.2, just that our objective function will be different this time.
Thus, let μ be a zero-mean degree 2 pseudo-distribution on $\{-1,1\}^n$, let $g \sim \mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}])$, and let the distribution of $\widehat{x} := \mathrm{sign}(g)$ be μ'. As noted in Lemma 3.4, $\mathbb{E}_{\mu'}[\widehat{x}_i\widehat{x}_j] = \frac{2}{\pi}\arcsin(\mathbb{E}[g_i g_j])$.
Thus

$$\mathbb{E}_{\mu'}[\widehat{x}^\mathsf{T} B\widehat{x}] = \sum_{i,j\in[n]} B_{ij}\mathbb{E}_{\mu'}[\widehat{x}_i\widehat{x}_j] = \sum_{i,j\in[n]} B_{ij}\frac{2}{\pi}\arcsin(\mathbb{E}[g_i g_j]) = \frac{2}{\pi}\left\langle B, \arcsin\left(\mathbb{E}[gg^\mathsf{T}]\right)\right\rangle$$

By Theorem 0.9 [10], $\arcsin\left(\mathbb{E}[gg^\mathsf{T}]\right) - \mathbb{E}[gg^\mathsf{T}]$ is a PSD matrix.
Now, if X, Y are PSD matrices, then $\langle X, Y\rangle \geq 0$. In particular,

$$\left\langle B, \arcsin\left(\mathbb{E}[gg^\mathsf{T}]\right) - \mathbb{E}[gg^\mathsf{T}]\right\rangle \geq 0$$

Thus

$$\mathbb{E}_{\mu'}[\widehat{x}^\mathsf{T} B\widehat{x}] = \frac{2}{\pi}\left\langle B, \arcsin\left(\mathbb{E}[gg^\mathsf{T}]\right)\right\rangle \geq \frac{2}{\pi}\left\langle B, \mathbb{E}[gg^\mathsf{T}]\right\rangle = \frac{2}{\pi}\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} B x] \implies \mathbb{E}_{\mu'}[\widehat{x}^\mathsf{T} B\widehat{x}] \geq \frac{2}{\pi}\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} B x]$$

We have thus proved an analog of Lemma 3.2 for general PSD matrices B. The rest of the discussion of the last chapter follows verbatim to yield the desired conclusions. ∎

4.1. Quadratic Optimization for General Matrices

Once we have conquered PSD matrices, why stop there? Why not consider *all* matrices?
Indeed, that's what we'll do. Thus, let B be any matrix. Note that

$$x^\mathsf{T} B x = x^\mathsf{T}\left(\frac{B + B^\mathsf{T}}{2}\right)x$$

Thus WLOG we can assume B to be symmetric. Furthermore, note that if $B = D + N$, where D is a diagonal matrix and all of N's diagonal entries are 0, then

$$x^\mathsf{T} B x = \mathrm{tr}(B) + x^\mathsf{T} N x$$

[10] Applied on the function $f(x) := \arcsin(x) - x$. Also, note that the diagonal entries of $\mathbb{E}[gg^\mathsf{T}]$ are equal to $\mathbb{E}[g_i^2] = \sin\left(\frac{\pi}{2}\mathbb{E}_{\mu'}[\widehat{x}_i^2]\right) = \sin\left(\frac{\pi}{2}\mathbb{E}_{\mu'}[1]\right) = \sin\left(\frac{\pi}{2}\right) = 1$, and thus Theorem 0.9 is indeed applicable

Thus WLOG we can assume that all of B's diagonal entries are 0.
Finally, before we state our approximation algorithm for general matrices, we shall need a lemma that converts our discrete optimization problem to a continuous optimization problem.

Lemma 4.2. For any $y \in [-1,1]^n$, there exists some $\widehat{y}_* \in \{-1,1\}^n$ such that $\widehat{y}_*^\mathsf{T} B \widehat{y}_* \geq y^\mathsf{T} B y$.

Proof. Consider the random variable $\widehat{y}$ on $\{-1,1\}^n$, where $\Pr(\widehat{y}_i = \pm 1) = \frac{1 \pm y_i}{2}$, independently, for each $i \in [n]$. Then $\mathbb{E}[\widehat{y}_i \widehat{y}_j] = \mathbb{E}[\widehat{y}_i]\mathbb{E}[\widehat{y}_j] = y_i y_j$ for $i \neq j$. Thus $\mathbb{E}[\widehat{y}^\mathsf{T} B \widehat{y}] = \mathrm{tr}(B) + \sum_{i \neq j} B_{ij} \mathbb{E}[\widehat{y}_i \widehat{y}_j] = \sum_{i \neq j} B_{ij} y_i y_j = y^\mathsf{T} B y$, where we use the fact that $\mathrm{tr}(B) = 0$ (since all of B's diagonal entries were assumed to be 0).
Thus $\mathbb{E}[\widehat{y}^\mathsf{T} B \widehat{y}] = y^\mathsf{T} B y$, which means there exists some $\widehat{y}_*$ such that $\widehat{y}_*^\mathsf{T} B \widehat{y}_* \geq \mathbb{E}[\widehat{y}^\mathsf{T} B \widehat{y}] = y^\mathsf{T} B y$, as desired. ■

Corollary 4.3. $\mathrm{opt}(B) = \max_{x \in \{-1,1\}^n} x^\mathsf{T} B x \geq 0$.

Proof. Invoking Lemma 4.2 with $y = 0 \in [-1,1]^n$ works. ■

Remark. Note that if $\mathrm{opt}(B) = 0$, then $-B$ is PSD.

Finally, we also state a useful tail bound for the Gaussian rounding of a pseudo-distribution.

Lemma 4.4. Let μ be a degree 2 pseudo-distribution, and let $g = \begin{bmatrix} g_1 & g_2 & \dots & g_n \end{bmatrix}^\mathsf{T} \sim \mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}])$. Then there is a constant $C = \mathcal{O}(1) > 0$ such that

$$\Pr(g_i \geq C\sqrt{\log n}) \leq \frac{1}{n^3}$$

Consequently, $\Pr(\|g\|_\infty \geq C\sqrt{\log n}) \leq \frac{1}{n^2}$.

Proof. Recall from Lemma 1.6 that all the diagonal entries of $\widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}]$ are 1, ie:- g_i's are Gaussian RVs with unit variance (and zero mean). A simple analysis of the CDF of a Gaussian RV X with variance σ^2 tells us

$$\Pr(|X - \mathbb{E}[X]| \geq t) \leq 2e^{-\frac{t^2}{2\sigma^2}}$$

Thus, taking $t = C\sqrt{\log n}$ for some appropriate constant C yields the first result, and then a union bound over $i \in [n]$ yields the second result. ■

We can now state our approximation algorithm for evaluating $\max_{x \in \{-1,1\}^n} x^\mathsf{T} B x$.

Theorem 4.5. For sufficiently large n, and for $c = \mathcal{O}(\log n)$, $\frac{\mathrm{opt}(B)}{c} - x^\mathsf{T} B x$ has a degree 2 SoS certificate.
We thus have a $\mathcal{O}(\log n)$-approximation algorithm for calculating $\mathrm{opt}(B)$.

Proof. As usual, we show that for any degree 2 pseudo-distribution μ, there is some (efficiently sample-able) distribution μ' on $\{-1,1\}^n$ such that $\mathbb{E}_{\mu'}[\widehat{x}^\mathsf{T} B \widehat{x}] \geq \frac{\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} B x]}{\mathcal{O}(\log n)}$. Now, by Lemma 4.2, it suffices if we choose μ' to be a distribution on $[-1,1]^n$ instead of $\{-1,1\}^n$: Indeed, suppose we go through the whole algorithm as in the Max-Cut/PSD case to get some $y \in [-1,1]^n$ such that $y^\mathsf{T} B y \geq \frac{\mathrm{opt}(B)}{\mathcal{O}(\log n)}$. Then the proof technique of Lemma 4.2 gives us a random variable $\widehat{y}$ such that $\widehat{y}^\mathsf{T} B \widehat{y}$, equals, in expectation, $y^\mathsf{T} B y$. Thus we can simply sample some point from $\widehat{y}$ and return that as the output of our algorithm.

Sample $g \sim \mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}])$. Then $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}Bx] = \langle B, \widetilde{\mathbb{E}}_\mu[xx^\mathsf{T}]\rangle = \langle B, \mathbb{E}[gg^\mathsf{T}]\rangle = \mathbb{E}[g^\mathsf{T}Bg]$. Since we are going to be choosing μ such that $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}Bx] \geq \mathrm{opt}(B) - \varepsilon$, and since $\mathrm{opt}(B) > 0$ [11], by choosing ε small enough we can assume that $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}Bx] = \mathbb{E}[g^\mathsf{T}Bg] > 0$. But then by Lemma 4.4,

$$\Pr(\|g\|_\infty \leq C\sqrt{\log n})\mathbb{E}\left[g^\mathsf{T}Bg \middle| \|g\|_\infty \leq C\sqrt{\log n}\right] \geq \left(1 - \frac{1}{n^2}\right)\mathbb{E}[g^\mathsf{T}Bg]$$

Now, consider the rounding algorithm [12]

$$\widehat{x}_i := \begin{cases} \frac{g_i}{C\sqrt{\log n}}, |g_i| \leq C\sqrt{\log n} \\ \frac{g_i}{|g_i|}, \text{otherwise} \end{cases}$$

Clearly, $\widehat{x} := \begin{bmatrix}\widehat{x}_1 & \widehat{x}_2 & \dots & \widehat{x}_n\end{bmatrix}^\mathsf{T} \in [-1,1]^n$. Let the probability distribution of $\widehat{x}$ be μ'.
Then

$$\mathbb{E}_{\mu'}[\widehat{x}^\mathsf{T}B\widehat{x}] \geq \Pr(\|g\|_\infty \leq C\sqrt{\log n})\mathbb{E}\left[\widehat{x}^\mathsf{T}B\widehat{x} \middle| \|g\|_\infty \leq C\sqrt{\log n}\right]$$

$$\geq \frac{1}{C^2\log n}\Pr(\|g\|_\infty \leq C\sqrt{\log n})\mathbb{E}\left[g^\mathsf{T}Bg \middle| \|g\|_\infty \leq C\sqrt{\log n}\right] \geq \frac{1}{C^2\log n}\left(1 - \frac{1}{n^2}\right)\mathbb{E}\left[g^\mathsf{T}Bg\right] = \frac{1}{\mathcal{O}(\log n)}\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}Bx]$$

■

Remark. In this theorem, we can replace "sufficiently large" n by $n > 60$, and $c = \mathcal{O}(\log n)$ by $c = 4\log n$ for the sake of concreteness.

Note that since B is a general matrix, we could obtain only a $\mathcal{O}(\log n)$-approximation algorithm as opposed to a constant-factor approximation algorithm. However, there is another class of matrices, which is also fairly general, for which a constant factor approximation algorithm for finding $\mathrm{opt}(B)$ exists, which we shall describe now.

4.2. Quadratic Optimization for Matrices with bipartite support

As usual, let B be a symmetric matrix. We define the *support* of B to be

$$\mathrm{supp}(B) := \{\{i,j\} : B_{ij} \neq 0\}$$

Clearly, $\mathrm{supp}(B)$ describes the edge set of some undirected graph on $[n]$. We are interested in finding an approximation algorithm for $\mathrm{opt}(B)$ for the case where $\mathrm{supp}(B)$ *is a bipartite graph*.
Now consider a B such that $\mathrm{supp}(B)$ is a bipartite graph, say with partitions X, Y such that $X \cup Y = [n], X \cap Y = \varnothing$. WLOG assume $x < y$ for all $x \in X, y \in Y$. Then note that B is a block matrix, ie:-

$$B = \begin{bmatrix} 0_{|X|\times|X|} & B' \\ (B')^\mathsf{T} & 0_{|Y|\times|Y|} \end{bmatrix}$$

For any vector $z \in \mathbb{R}^n$, denote by z_S the vector $\begin{bmatrix} z_{s_1} & z_{s_2} & \dots & z_{s_{|S|}}\end{bmatrix}^\mathsf{T} \in \mathbb{R}^{|S|}$, where $S = \{s_1, s_2, \dots, s_{|S|}\}$. Then note that $x^\mathsf{T}Bx = 2x_X^\mathsf{T}B'x_Y$.
Thus, upto some tweaking and fiddling, we can focus on a modified problem: Given an *arbitrary* matrix M (with *no* restrictions such as symmetry/zeros on the diagonal, etc.), evaluate

$$\max_{x,y\in\{-1,1\}^n} x^\mathsf{T}My$$

Recall Definition 0.5. We shall now connect our optimization problem to generalized operator norms.

[11] If $\mathrm{opt}(B) = 0$, then $-B$ is PSD, and thus the statement of this theorem holds trivially, since the Cholesky decomposition of $-B$ yields a degree 2 SoS proof for $-x^\mathsf{T}Bx$. One might also view this as a "pre-processing step": We first check if $-B$ is Cholesky decomposable: If yes, we're done. Otherwise we invoke the algorithm mentioned here.

[12] this is a special case of a class of roundings known as RPR^2 roundings

Theorem 4.6. Let B be an arbitrary matrix. Then

$$\max_{x,y\in\{-1,1\}^n} x^\mathsf{T} My = \|M\|_{\infty\to 1}$$

Proof. Fix some $y \in \{-1,1\}^n$. Then

$$\max_{x\in\{-1,1\}^n} x^\mathsf{T} My = \max_{x\in\{-1,1\}^n} \langle x, My\rangle = \|My\|_1$$

Indeed, $\langle x, My\rangle$ can be easily seen to be maximized when $x = \operatorname{sign}(My)$.
Now, note that

$$\|M\|_{\infty\to 1} = \max_{y\in\mathbb{R}^n, \|y\|_\infty = 1} \|My\|_1$$

Since $y \mapsto My$ is a convex function, it is maximized on some vertex of the cube $[-1,1]^n$. Thus

$$\max_{y\in\mathbb{R}^n, \|y\|_\infty = 1} \|My\|_1 = \max_{y\in\{-1,1\}^n} \|My\|_1$$

But

$$\max_{y\in\{-1,1\}^n} \|My\|_1 = \max_{y\in\{-1,1\}^n} \max_{x\in\{-1,1\}^n} x^\mathsf{T} My = \max_{x,y\in\{-1,1\}^n} x^\mathsf{T} My$$

as desired. ■

We thus have to find an approximation algorithm for evaluating $\|M\|_{\infty\to 1}$ for an arbitrary matrix $M \in \mathbb{R}^{n\times n}$. Before stating the main theorem, we pass through a small proposition.

Lemma 4.7. Let M be an arbitrary matrix.
Let μ' be a distribution on $\{-1,1\}^n$, and μ be a pseudo-distribution on $\{-1,1\}^n$ such that $\mathbb{E}_{\mu'}[xy^\mathsf{T}] = \gamma\widetilde{\mathbb{E}}_\mu[xy^\mathsf{T}]$. Then $\mathbb{E}_{\mu'}[x^\mathsf{T} My] = \gamma\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} My]$.

Proof. Note that

$$\mathbb{E}_{\mu'}[x^\mathsf{T} My] = \left\langle M, \mathbb{E}_{\mu'}[xy^\mathsf{T}]\right\rangle = \left\langle M, \gamma\widetilde{\mathbb{E}}_\mu[xy^\mathsf{T}]\right\rangle = \gamma\left\langle M, \widetilde{\mathbb{E}}_\mu[xy^\mathsf{T}]\right\rangle = \gamma\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} My]$$

■

Theorem 4.8. $(K_G\|M\|_{\infty\to 1} - x^\mathsf{T} My)$ has a degree 2 SoS certificate, where K_G is the so-called Grothendieck's constant. The reader may look up Grothendieck's inequality ([Gro23]) for further details.

Remark. A few remarks are in order:

1. $(K_G\|M\|_{\infty\to 1} - x^\mathsf{T} My)$ is to be treated as a polynomial in $x_1, \dots, x_n, y_1, \dots, y_n$.
2. Let κ be a distribution/pseudo-distribution on $\{-1,1\}^n$. When we write $\mathbb{E}_\kappa[f(x,y)]$ or $\widetilde{\mathbb{E}}_\kappa[f(x,y)]$, we are taking the (pseudo)expectation of $f(x,y)$, where x and y are *independent* of each other.
3. The exact value of Grothendieck's constant is unknown: We only have the bounds $\frac{\pi}{2} \le K_G < \frac{2\ln(1+\sqrt{2})}{\pi}$.

Proof. We actually prove a weaker statement, ie:- $\frac{2\ln(1+\sqrt{2})}{\pi}\|M\|_{\infty\to 1} - x^\intercal My$ has a degree 2 SoS certificate.
Let μ be a degree 2 pseudo-distribution. We describe an (efficiently sampleable) distribution μ' on $\{-1,1\}^n$ such that $\mathbb{E}_{\mu'}[x^\intercal My] = \frac{2c}{\pi}\widetilde{\mathbb{E}}_\mu[x^\intercal My]$, where $c = \ln(1+\sqrt{2})$. When we would have proven this, we will be done. In the light of Lemma 4.7, showing that $\mathbb{E}_{\mu'}[xy^\intercal] = \frac{2c}{\pi}\widetilde{\mathbb{E}}_\mu[xy^\intercal]$ suffices.
Consider the matrix

$$\Sigma := \begin{bmatrix} \sinh\left(c\widetilde{\mathbb{E}}_\mu[xx^\intercal]\right) & \sin\left(c\widetilde{\mathbb{E}}_\mu[xy^\intercal]\right) \\ \sin\left(c\widetilde{\mathbb{E}}_\mu[yx^\intercal]\right) & \sinh\left(c\widetilde{\mathbb{E}}_\mu[yy^\intercal]\right) \end{bmatrix}$$

We claim that Σ is PSD: Indeed, note that if $\begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}$ is PSD, then $\begin{bmatrix}\Sigma_{11} & -\Sigma_{12}\\ -\Sigma_{21} & \Sigma_{22}\end{bmatrix}$ is PSD too, since

$$\begin{bmatrix}v\\w\end{bmatrix}^\intercal \begin{bmatrix}\Sigma_{11} & -\Sigma_{12}\\ -\Sigma_{21} & \Sigma_{22}\end{bmatrix}\begin{bmatrix}v\\w\end{bmatrix} = \begin{bmatrix}v\\-w\end{bmatrix}^\intercal \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\begin{bmatrix}v\\-w\end{bmatrix}$$

Thus, for any $k \in \mathbb{N}_0$, both $\begin{bmatrix}\Sigma_{11}^{(k)} & \Sigma_{12}^{(k)}\\ \Sigma_{21}^{(k)} & \Sigma_{22}^{(k)}\end{bmatrix}$ and $\begin{bmatrix}\Sigma_{11}^{(k)} & -\Sigma_{12}^{(k)}\\ -\Sigma_{21}^{(k)} & \Sigma_{22}^{(k)}\end{bmatrix}$ are PSD, where recall that $X^{(k)}$ was the k-wise Hadamard product of X with itself.
Now, note that $\sinh(x), \sin(x)$ are analytic functions given by

$$\sinh(x) = \sum_{n=0}^{\infty}\frac{x^{2n+1}}{(2n+1)!}$$
$$\sin(x) = \sum_{n=0}^{\infty}\frac{(-1)^n x^{2n+1}}{(2n+1)!}$$

Thus, mimicking the proof of Theorem 0.9, we can conclude that if $\begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}$ is PSD, then so is $\begin{bmatrix}\sinh(\Sigma_{11}) & \sin(\Sigma_{12})\\ \sin(\Sigma_{21}) & \sinh(\Sigma_{22})\end{bmatrix}$.
Now, note that $\begin{bmatrix}\widetilde{\mathbb{E}}_\mu[xx^\intercal] & \widetilde{\mathbb{E}}_\mu[xy^\intercal]\\ \widetilde{\mathbb{E}}_\mu[yx^\intercal] & \widetilde{\mathbb{E}}_\mu[yy^\intercal]\end{bmatrix} = \widetilde{\mathbb{E}}_\mu\left[\begin{bmatrix}x\\y\end{bmatrix}\cdot\begin{bmatrix}x\\y\end{bmatrix}^\intercal\right]$, which is PSD since μ is a *degree 2* pseudo-distribution.
Thus Σ is PSD. Consider the normal distribution $\mathcal{G} := \mathcal{N}(0,\Sigma)$, and let μ' be the Gaussian Rounding of $\mathcal{G}$, ie:- we sample some x from $\mathcal{G}$, and set $\widehat{x} = \text{sign}(x)$.
Now, consider $\mathbb{E}_{\mu'}[xy^\intercal]$. Suppose $x = \text{sign}(g), y = \text{sign}(h)$, where $g, h \sim \mathcal{G}$. Now, recall from the proofs of Lemma 3.2 or Lemma 3.4 that $\mathbb{E}_{\mu'}[xy^\intercal] = \frac{2}{\pi}\arcsin(\mathbb{E}_{\mathcal{G}}[gh^\intercal])$. However, there is a small thing to be taken care of: The relation "$\mathbb{E}_{\mu'}[xy^\intercal] = \frac{2}{\pi}\arcsin(\mathbb{E}_{\mathcal{G}}[gh^\intercal])$" was a consequence of Lemma 0.18, which in turn required that all the diagonal elements of the Gaussian's covariance matrix be 1. Now, note that the covariance matrix of $\mathcal{G}$ is Σ, and all diagonal entries of Σ equal $\sinh(c)$. Consequently, we must have $c = \sinh^{-1}(1) = \ln(1+\sqrt{2})$, as desired.
Since g, h are independent draws from $\mathcal{G}$, $\mathbb{E}_{\mathcal{G}}[gh^\intercal]$ is just the off-diagonal block of the covariance matrix of $\mathcal{G}$.
Thus

$$\mathbb{E}_{\mathcal{G}}[gh^\intercal] = \sin\left(c\widetilde{\mathbb{E}}_\mu[xy^\intercal]\right)$$

Consequently,

$$\mathbb{E}_{\mu'}[xy^\intercal] = \frac{2}{\pi}\arcsin(\mathbb{E}_{\mathcal{G}}[gh^\intercal]) = \frac{2}{\pi}\arcsin\left(\sin\left(c\widetilde{\mathbb{E}}_\mu[xy^\intercal]\right)\right) = \frac{2c}{\pi}\widetilde{\mathbb{E}}_\mu[xy^\intercal]$$

as desired. ∎

Though we will not see it here, [AMMN06] showed that if the support of a matrix B is some general graph G, then there is an $\mathcal{O}(\log(\chi(G)))$-approximation algorithm for evaluating $\text{opt}(B)$. They also showed that there can be *no* $o(\log(\omega(G)))$-approximation algorithm for the same. Recall that $\chi(G)$ was the minimum number of colors needed to color a graph, while $\omega(G)$ is the size of the largest clique in G.

§5. Higher Degree Sum of Squares

So far we have seen quite a few examples of degree 2 SoS. We will now see some applications of degree 4 SoS. Before that, we present a very useful lemma that pops up when we try to deal with the degree 4 Sum of Squares.

Lemma 5.1 (Squared Triangle Inequality)**.** For any $a, b, c \in \{-1, 1\}$,

$$(a-c)^2 \leq (a-b)^2 + (b-c)^2$$

Proof. Just note that

$$(a-b)^2 + (b-c)^2 - (a-c)^2 =_{\{-1,1\}} \left(\frac{(b-c)(a-b)}{\sqrt{2}}\right)^2$$

■

As our first example of degree 4 SoS, we see an algorithm for solving the Min-Cut problem [13].
Let $f_G(x)$ be our usual cut function, ie:-

$$f_G(x) = \frac{1}{4}\sum_{ij \in E}(x_i - x_j)^2$$

Now, by Theorem 2.6, we can find (in polynomial time) a degree 4 pseudo-distribution μ such that $\widetilde{\mathbb{E}}_\mu[f_G] \leq \text{opt}_{\text{SoS}_4} + \varepsilon$, where

$$\text{opt}_{\text{SoS}_4} := \min_{\kappa \text{ is a degree 4 pseudo-distribution}} \widetilde{\mathbb{E}}_\kappa[f_G]$$

Clearly, $\text{opt}_{\text{SoS}_4}$ is at most the size of the minimum cut. We will now describe a distribution μ' on $\{-1,1\}^n$ such that $\mathbb{E}_{\mu'}[f_G] \leq \widetilde{\mathbb{E}}_\mu[f_G]$, and thus if we sample a cut from μ', then in expectation, we would be sampling the minimum cut.
Now, for any $i, j \in [n]$, define

$$D(i,j) := \frac{1}{4}\widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2]$$

We argue that $D : [n]^2 \mapsto \mathbb{R}$ is a metric on $[n]$: Since μ is a degree 4 pseudo-distribution, it is a degree 2 pseudo-distribution too, and consequently $\widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2] \geq 0$ for all $i, j \in [n]$. Thus D is non-negative. Clearly, D is symmetric. Finally, note that

$$4(D(i,j) + D(j,k) - D(i,k)) = \widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2] + \widetilde{\mathbb{E}}_\mu[(x_j - x_k)^2] - \widetilde{\mathbb{E}}_\mu[(x_k - x_i)^2] = \widetilde{\mathbb{E}}_\mu\left[\left(\frac{(x_j - x_k)(x_i - x_j)}{\sqrt{2}}\right)^2\right]$$

But $\widetilde{\mathbb{E}}_\mu\left[\left(\frac{(x_j - x_k)(x_i - x_j)}{\sqrt{2}}\right)^2\right]$ is non-negative since μ is a degree 4 pseudo-distribution.
Thus D satisfies the triangle inequality too, as desired.
Finally, also note that $D(i,j) \leq 1$ for all i, j: Indeed, $1 - D(i,j) = \frac{1}{4}\widetilde{\mathbb{E}}_\mu[4 - (x_i - x_j)^2] = \frac{1}{4}\widetilde{\mathbb{E}}_\mu[(x_i + x_j)^2] \geq 0$, since μ is a degree 2 pseudo-distribution.
Now, consider the "line" map $\ell : [n] \mapsto [0,1]$, $\ell(i) := D(i,1)$. Uniformly sample a t from $[0,1]$, and output the cut $\{i \in [n] : \ell(i) \leq t\}$.
The probability that some edge $\{i,j\} \in E$ is cut is given by

$$\Pr(\ell(i) \leq t \leq \ell(j) \vee \ell(j) \leq t \leq \ell(i)) = |\ell(i) - \ell(j)| = |D(i,1) - D(j,1)| \leq D(i,j)$$

Thus the expected size of a cut, by our sampling procedure [14], is at most $\sum_{ij \in E} D(i,j) = \widetilde{\mathbb{E}}_\mu[f_G]$, as desired.
With this warm-up, we are now ready to attack the big problems of the day.

[13] note that the Min-Cut problem has a polynomial time algorithm, namely the Ford-Fulkerson algorithm, but looking at Min-Cut is instructive in this context, so we do it

[14] note how we implicitly described a distribution μ' by directly describing its sampling procedure instead

5.1. Approximating Conductance

We will see some applications of degree 4 SoS in approximating the *conductance* of a graph.
In keeping with our tradition, we will continue to investigate cuts of graphs. We already know that Min-Cut is solvable in polynomial time, and while Max-Cut is NP-hard, a good constant factor approximation for Max-Cut is achievable in polynomial time.
Today we will investigate something called the *normalized cut* of a graph, which we define below.

Definition 5.1. Consider a d-regular graph $G = G(V, E)$ with n vertices, and consider some non-empty $S \subsetneq V$. Then the normalized cut corresponding to S, also known as the *conductance* of S, is defined as

$$\Phi_G(S) := \frac{E(S, V \setminus S)}{\frac{d}{n}|S| \cdot |V \setminus S|} = \frac{|\{\{i, j\} \in E : i \in S, j \notin S\}|}{\frac{d}{n}|S| \cdot |V \setminus S|}$$

Remark. It should be quite apparent why $\Phi_G(S)$ is called the "normalized" cut of S: Note that the denominator in the expression for $\Phi_G(S)$ contains the expected size of the cut induced by S, had G been a random graph.
Thus the normalized cut seeks to measure, with respect to some "standard", how big/small the cut induced by S is.

Definition 5.2 (Conductance of Graph)**.** Given a d-regular graph $G = G(V, E)$, the conductance of G is defined to be

$$\Phi_G := \min_{\varnothing \neq S \subsetneq V} \Phi_G(S)$$

Remark. The conductance of a graph is a very important object, especially in the study of random walks over graphs. A low conductance means that the graph has a "bottleneck": If we begin a random walk on the graph, then it takes quite some time to get to the other side of the bottleneck.

The problem of finding the conductance of a graph is also known as the *sparsest cut problem*.
Quite surprisingly, unlike Max-Cut, which has a constant factor approximation algorithm, [CKK+05] showed that (assuming the Unique Games Conjecture) finding *any* constant factor approximation algorithm for the sparsest cut problem is NP-hard.
We now formally state the sparsest cut problem.

Problem. Given any graph G, calculate

$$\min_{x \in \{-1,1\}^n} \frac{f_G(x)}{\frac{d}{n} f_{K_n}(x)}$$

It is easy to see the equivalence of this formulation with the definition given above.
Now, note that optimizing rational functions such as $\frac{f_G(x)}{\frac{d}{n} f_{K_n}(x)}$ is hard: We shall thus focus on getting as *large* α as we can such that $f_G(x) - \alpha \frac{d}{n} f_{K_n}(x)$ has a Sum of Squares certificate.
Some progress in this direction was already made by [AM85], who proved Cheeger's inequality for graphs.

Theorem 5.2 (Cheeger's Inequality)**.** There is a degree 2 SoS certificate for

$$f_G(x) - \frac{\Phi_G^2}{2} \cdot \frac{d}{n} f_{K_n}(x)$$

A more algorithmic version of the above inequality goes as follows:

Theorem 5.3. Consider any pseudo-distribution μ of degree ≥ 2 such that $\widetilde{\mathbb{E}}_\mu[f_G(x) - C\frac{d}{n}f_{K_n}(x)] \leq 0$ for some constant C. Then one can find a set S with $\Phi_G(S) = \mathcal{O}(\sqrt{C})$.

Note that when $\Phi(G) = \mathcal{O}(1)$, ie:- when $\Phi(G)$ is a constant, Theorem 5.2 gives us a constant factor approximation algorithm. This is nice, because graphs with constant-sized conductances are known as expander graphs, and they are very important objects throughout graph theory and computer science. Alternatively stated, expander graphs are accompanied by a degree 2 SoS proof that they are expander graphs.
However, when $\Phi(G) = o(1)$, then the approximation algorithm given by Theorem 5.2 is rather weak: For example, if $\Phi(G) = \Theta(1/\sqrt{n})$, then Theorem 5.2 is a $\sqrt{n}$-approximation algorithm, which is not good!
This situation is remedied by the Arora-Rao-Vazirani (ARV) algorithm, which is a $\mathcal{O}(\sqrt{\log n})$-approximation algorithm for the sparsest cut problem. We will first prove a *Global structure theorem* en route to this algorithm.

5.2. Global Structure Theorem

Throughout this section, let μ denote a degree 4 pseudo-distribution. Since we'll be dealing with the (pseudo)expectations of quadratic functions such as $(x_i - x_j)^2$ throughout, WLOG we can assume that $\widetilde{\mathbb{E}}_\mu[x] = 0$.
As in the Min-Cut algorithm, we define $D(i,j) := \frac{1}{4}\widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2]$.

Definition 5.3. $A, B \subseteq V$ are said to be Δ-*separated sets* if for every $i \in A, j \in B$, $D(i,j) \geq \Delta$, and $|A| \cdot |B| = \Omega(n^2)$.

Theorem 5.4 (Weak Global Structure Theorem)**.** Let G be a d-regular graph such that $\sum_{i,j} D(i,j) = \Omega(n^2)$. Then one can find, in $\mathrm{poly}(n)$ time, sets A and B which are $\Omega(1/\log n)$-separated.

Proof. Recall that $\widetilde{\mathbb{E}}_\mu[x] = 0$, and denote by $\mathcal{G}$ the normal distribution $\mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\intercal])$, and sample g from $\mathcal{G}$. Define

$$A^{(0)} := \{i \in [n] : g_i \leq -1\}, B^{(0)} := \{j \in [n] : g_j \geq 1\}$$

Now, for any i, j, $\begin{bmatrix} g_i \\ g_j \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} 1 & \widetilde{\mathbb{E}}_\mu[x_ix_j] \\ \widetilde{\mathbb{E}}_\mu[x_ix_j] & 1 \end{bmatrix}\right)$ [15]. Using some elementary calculus, it is easy to see that there is some constant C such that $\Pr(g_i \leq -1, g_j \geq 1) \geq \frac{C}{4}(2 - 2\widetilde{\mathbb{E}}_\mu[x_ix_j]) = CD(i,j)$.
Thus

$$\mathbb{E}[|A^{(0)}| \cdot |B^{(0)}|] = \sum_{i,j} \Pr(g_i \leq -1, g_j \geq 1) \geq C\sum_{i,j} D(i,j) = \Omega(n^2)$$

Now, suppose for some $i \in A^{(0)}, j \in B^{(0)}$ we have $D(i,j) \leq \Delta$, ie:- $\widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2] \leq 4\Delta$. Now, $\widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2] = \mathbb{E}[(g_i - g_j)^2]$. Note that $g_i - g_j$ is a Gaussian RV too, whose variance is at most 4Δ. Yet the value of $g_i - g_j$, in this particular instance is ≤ -2, since $g_i \leq -1, g_j \geq 1$. The probability of this happening is $\Theta\left(e^{-\frac{\mathcal{O}(1)}{\Delta}}\right)$. Consequently, the probability that for *any* $i \in A, j \in B$, $D(i,j) \leq \Delta$ is $\leq n^2\Theta\left(e^{-\frac{\mathcal{O}(1)}{\Delta}}\right)$.
Thus the probability that $A^{(0)}, B^{(0)}$ are Δ-separated is $\geq 1 - n^2\Theta\left(e^{-\frac{\mathcal{O}(1)}{\Delta}}\right) = 1 - \mathcal{O}(1)$ when $\Delta = \mathcal{O}\left(\frac{1}{\log n}\right)$.
Consequently with some positive probability $A^{(0)}, B^{(0)}$ are well-separated, and by carrying out this sampling procedure $\mathrm{poly}(n)$ many times, we will find, with very high probability, a particular pair of well-separated sets. ■

[15] recall Lemma 1.6

We shall now progress towards proving the *strong* Global Structure Theorem, which says that if the hypotheses of Theorem 5.4 hold, then, in fact, we have a pair of $\Omega(1/\sqrt{\log n})$-separated sets.
The way we go about showing it is as follows: We first construct $A^{(0)}, B^{(0)}$ as in the proof of Theorem 5.4. We then construct a directed bipartite graph 'H', between $A^{(0)}, B^{(0)}$, where two vertices are connected if the distance between them is $\leq \Delta$ [16]. We then construct a *maximal matching* M in H such that when M is deleted from H, we don't lose too many vertices, yet all the vertices close to each other are gone, and we get our desired well-separated set.
It is now time to fill in the details.

5.2.1. The Details

Construct $A^{(0)}, B^{(0)}$ as in the proof of Theorem 5.4.
Let H be a graph on $[n]$, where $E(H) := \{\{i,j\} : D(i,j) \leq \Delta\}$. Consider the subgraph of H induced by $A^{(0)} \cup B^{(0)}$, and orient each edge in that subgraph to be going from $A^{(0)}$ to $B^{(0)}$. Further, order those edges lexicographically. Generate a maximal matching M on that subgraph by choosing the (lexicographically) smallest edge, adding it to M, removing the endpoints of the said edge, and so on. Note that the construction of M is completely deterministic once we have fixed the subgraph, ie:- $H \cap (A^{(0)} \times B^{(0)})$. Furthermore, note that when we delete the vertices in M from $H \cap (A^{(0)} \times B^{(0)})$, all the remaining edges have length $\geq \Delta$, because otherwise we could have extended M.
We have to now prove that M is not too large, and thus deleting the edges and vertices of M doesn't cause $|A^{(0)} \setminus V(M)| \cdot |B^{(0)} \setminus V(M)|$ to become $o(n^2)$.
Let $H^k(i)$ be the set of vertices that are at most k steps away from i in H. Define $\gamma_i^{(k)} := \max_{j \in H^k(i)}(g_j - g_i)$, and define $\phi_k := \sum_{i \in [n]} \mathbb{E}[\gamma_i^{(k)}]$. Then

Lemma 5.5. For any $k \in \mathbb{N}$,

$$\phi_{k+1} - \phi_k \geq 2\mathbb{E}[|M|] - \mathcal{O}(n) \max_{\substack{i \in [n] \\ j \in H^{k+1}(i)}} \sqrt{\mathbb{E}[(g_i - g_j)^2]}$$

Proof. Note that if $(i,j) \in E(H)$, then $H^k(j) \subseteq H^{k+1}(i)$. Consequently, $\gamma_i^{(k+1)} \geq \gamma_j^{(k)} + (g_j - g_i) \geq \gamma_j^{(k)} + 2$, where recall that $g_j \geq 1, -1 \geq g_i$ since $j \in B^{(0)}, i \in A^{(0)}$.
Now, for all $i \in [n]$, define the variables L_i, R_i, where

$$L_i := \begin{cases} 1, i \text{ has an outgoing edge in } M \\ 0, i \text{ has an incoming edge in } M \\ \frac{1}{2}, \text{otherwise} \end{cases}$$

$$R_i := 1 - L_i$$

Finally, note that since $H^k(\cdot) \subseteq H^{k+1}(\cdot)$, $\gamma_\cdot^{(k+1)} \geq \gamma_\cdot^{(k)}$. Consequently, if $(i,j) \notin E(M)$, then write the inequality

$$\gamma_i^{(k+1)} + \gamma_j^{(k+1)} \geq \gamma_i^{(k)} + \gamma_j^{(k)}$$

and if $(i,j) \in E(M)$, then write the inequality $\gamma_i^{(k+1)} \geq \gamma_j^{(k)} + 2$. Summing over all these inequalities for all possible $i,j \in [n]$ yields

$$(n-1)\left(\sum_{i \in [n]} L_i \gamma_i^{(k+1)}\right) \geq (n-1)\left(\sum_{i \in [n]} R_i \gamma_i^{(k)} + 2|M|\right) \implies \sum_{i \in [n]} L_i \gamma_i^{(k+1)} \geq \sum_{i \in [n]} R_i \gamma_i^{(k)} + 2|M|$$

$$\implies \sum_{i \in [n]} \mathbb{E}\left[L_i \gamma_i^{(k+1)}\right] \geq \sum_{i \in [n]} \mathbb{E}\left[R_i \gamma_i^{(k)}\right] + 2\mathbb{E}[|M|]$$

[16] we shall later fix Δ to be $\mathcal{O}(\sqrt{\log n})$, but for now let it remain indeterminate

Now, note that for some $g \sim \mathcal{N}(0, \widetilde{\mathbb{E}}_\mu[xx^\intercal])$, $g_i \geq 1$ and $g_i \leq -1$ are equiprobable events. Consequently, for any $i \in [n]$, i lies in either $A^{(0)}$ or $B^{(0)}$ with equal probability, and consequently, $\mathbb{E}[L_i] = \mathbb{E}[R_i] = \frac{1}{2}$ for all $i \in [n]$.
Now, we must extricate L_i, R_i from within the expectation operator somehow: We prove that that doesn't cost us too much. Indeed,

$$\left|\mathbb{E}\left[L_i\gamma_i^{(k+1)}\right] - \mathbb{E}\left[L_i\right]\cdot\mathbb{E}\left[\gamma_i^{(k+1)}\right]\right| = \left|\mathbb{E}\left[\left(L_i - \mathbb{E}[L_i]\right)\left(\gamma_i^{(k+1)} - \mathbb{E}\left[\gamma_i^{(k+1)}\right]\right)\right]\right| = \text{Corr}\left(L_i, \gamma_i^{(k+1)}\right)$$

$$\leq \underbrace{\sqrt{\text{Var}(L_i)}}_{=\mathcal{O}(1)}\sqrt{\text{Var}\left(\gamma_i^{(k+1)}\right)} \overset{\text{Lemma 0.19}}{\leq} \mathcal{O}(1)\sqrt{\max_{j\in H^{k+1}(i)}\mathbb{E}[(g_j - g_i)^2]}$$

The invocation of Lemma 0.19 is justified by the fact that $\gamma_i^{(k+1)}$ is the maximum of the Gaussian RVs $(g_j - g_i)$ for $j \in H^{k+1}(i)$.
Thus

$$\phi_{k+1} = \sum_{i\in[n]}\mathbb{E}[\gamma_i^{(k+1)}] = 2\sum_{i\in[n]}\mathbb{E}[L_i]\mathbb{E}[\gamma_i^{(k+1)}] \geq 2\sum_{i\in[n]}\mathbb{E}[L_i\gamma_i^{(k+1)}] - \mathcal{O}(1)\sum_{i\in[n]}\sqrt{\max_{j\in H^{k+1}(i)}\mathbb{E}[(g_j - g_i)^2]}$$

$$\geq 2\sum_{i\in[n]}\mathbb{E}[R_i\gamma_i^{(k)}] + 2\mathbb{E}[|M|] - \mathcal{O}(1)\sum_{i\in[n]}\sqrt{\max_{j\in H^{k+1}(i)}\mathbb{E}[(g_j - g_i)^2]}$$

$$\geq 2\sum_{i\in[n]}\mathbb{E}[R_i]\mathbb{E}[\gamma_i^{(k)}] + 2\mathbb{E}[|M|] - \mathcal{O}(1)\sum_{i\in[n]}\sqrt{\max_{j\in H^{k+1}(i)}\mathbb{E}[(g_j - g_i)^2]}$$

$$= \phi_k + 2\mathbb{E}[|M|] - \mathcal{O}(n)\sqrt{\max_{\substack{i\in[n]\\ j\in H^{k+1}(i)}}\mathbb{E}[(g_j - g_i)^2]} = \phi_k + 2\mathbb{E}[|M|] - \mathcal{O}(n)\max_{\substack{i\in[n]\\ j\in H^{k+1}(i)}}\sqrt{\mathbb{E}[(g_j - g_i)^2]}$$

■

Given Lemma 5.5, the rest is easy: Note that if i and j are k steps apart in H, then $\mathbb{E}[(g_i - g_j)^2] = \widetilde{\mathbb{E}}_\mu[(x_i - x_j)^2] \leq k\Delta$ by the Squared Triangle Inequality. Consequently, by Lemma 5.5, $\phi_{k+1} - \phi_k \geq 2\mathbb{E}[|M|] - \mathcal{O}(n)\sqrt{k\Delta}$. Set $k_0 = \frac{c}{\Delta}\left(\frac{\mathbb{E}[|M|]}{n}\right)^2$, where c is such that the last inequality in the following chain of inequalities holds, for $k \leq k_0$:

$$\phi_{k+1} - \phi_k \geq 2\mathbb{E}[|M|] - \mathcal{O}(n)\sqrt{\frac{c}{\Delta}\left(\frac{\mathbb{E}[|M|]}{n}\right)^2\Delta} \geq \mathbb{E}[|M|]$$

Then $\phi_{k_0} \geq k_0\mathbb{E}[|M|]$. Finally, note that $\max_{i,j\in[n]}(g_j - g_i) \geq \frac{\phi_{k_0}}{k_0} \geq \frac{\phi_{k_0}}{n}$, and thus

$$\max_{i,j\in[n]}(g_j - g_i) = \Omega\left(\frac{k_0}{n}\mathbb{E}[|M|]\right) = \frac{\Omega(1)}{\Delta}\left(\frac{\mathbb{E}[|M|]}{n}\right)^3 \implies \mathbb{E}\left[\max_{i,j\in[n]}(g_j - g_i)\right] = \frac{\Omega(1)}{\Delta}\left(\frac{\mathbb{E}[|M|]}{n}\right)^3$$

On the other hand, note that $g_j - g_i$ are all Gaussian RVs with variance at most $\mathcal{O}(1)$ [17]. It is then standard to show [18] that $\mathbb{E}\left[\max_{i,j\in[n]}(g_j - g_i)\right] \leq \mathcal{O}(\sqrt{\log n})$.
Thus

$$\frac{\Omega(1)}{\Delta}\left(\frac{\mathbb{E}[|M|]}{n}\right)^3 \leq \mathcal{O}(\sqrt{\log n})$$

Setting $\Delta = \Theta\left(\frac{1}{\sqrt{\log n}}\right)$ yields that $\mathbb{E}[|M|] = \mathcal{O}(n)$, which implies that deleting the vertices of M from $A^{(0)}, B^{(0)}$ still keeps $|A^{(0)} \setminus V(M)| \cdot |B^{(0)} \setminus V(M)| = \Omega(n^2)$, as desired.
We have thus proved the strong Global Structure Theorem, which we formally state below.

[17] note that the covariance matrix of g_i, g_j has 1s on its diagonal, and thus the covariance of g_i, g_j must be at most 1, from which it follows that $g_j - g_i$ has bounded variance
[18] see, for example, this stackexchange answer

Theorem 5.6 (Strong Global Structure Theorem)**.** Let G be a d-regular graph such that $\sum_{i,j} D(i,j) = \Omega(n^2)$. Then one can find, in $\mathrm{poly}(n)$ time, sets A and B which are $\Omega(1/\sqrt{\log n})$-separated.

5.3. Arora-Rao-Vazirani Algorithm

Theorem 5.7. Let G be a d-regular graph with n vertices. Then there is a degree 4 SoS certificate for

$$f_G(x) - \frac{\Phi_G}{\Theta(\sqrt{\log n})} \cdot \frac{d}{n} f_{K_n}(x)$$

Proof. As in all approximation algorithm analyses before, let μ be (close to) the optimal pseudo-distribution maximizing the pseudo-expectation of $f_G(x) - \Phi_G \cdot \frac{d}{n} f_{K_n}(x)$.

First, invoke Theorem 5.6 [19] to find two separated sets A, B such that $D(A,B) \geq \Omega\left(\frac{1}{\sqrt{\log n}}\right)$.

Now, we mimic the analyses of the min-cut algorithm: Recall how we constructed a distribution by making a 'cut' in the 'line map' $\ell : [n] \mapsto [0,1] : i \mapsto D(i,1)$. We now consider the map $\ell' : [n] \mapsto [0,1] : i \mapsto D(i,A)$, where $D(i,A) := \min_{a \in A} D(i,a)$. It is not too difficult to see that $D(\cdot, A)$ is a bounded metric on $[n]$ too, and thus ℓ' is a "valid" line map. Once again, let μ' be the distribution of the cut of the line map ℓ'. As in the min-cut case, we once again have $\mathbb{E}_{\mu'}[f_G(x)] \leq \widetilde{\mathbb{E}}_{\mu}[f_G(x)]$ [20]. But also note that

$$\mathbb{E}_{\mu'}[f_{K_n}(x)] = \frac{1}{4}\sum_{i,j} \mathbb{E}_{\mu'}[(x_i - x_j)^2] \geq \frac{1}{4}\sum_{j \in B} D(j,A) \geq \frac{\Delta}{4}|A| \cdot |B| \geq \Omega\left(\frac{1}{\sqrt{\log n}}\right) n^2 \geq \Omega\left(\frac{1}{\sqrt{\log n}}\right) \widetilde{\mathbb{E}}_{\mu}[f_{K_n}(x)]$$

where the last inequality follows from the fact that $\widetilde{\mathbb{E}}_{\mu}[f_{K_n}(x)] = \sum_{i,j} D(i,j) \leq \binom{n}{2} = \mathcal{O}(n^2)$. ∎

[19] Note that one of the hypotheses in Theorem 5.6 was that $\sum_{i,j} D(i,j) = \mathcal{O}(n^2)$. We shall omit the proof of the fact that μ indeed satisfies this property

[20] in fact, one might note that the min-cut algorithm described above works verbatim with the metric $D(\cdot, 1)$ replaced by the metric $D(\cdot, A)$.

§6. Unique Games Conjecture

We have referenced the Unique Games Conjecture (UGC) multiple times before: For example, we said that conditional on the UGC, an $(\alpha_{GW}+\varepsilon)$-approximation algorithm for MAX-CUT is NP-hard; or, conditional on the UGC, any constant factor approximation algorithm for the unique sparsest cut is NP-hard.
We thus take some time out to understand what the Unique Games Conjecture is all about.

Definition 6.1 (2-Constraint Satisfaction Problem (2-CSP))**.** Suppose we have n variables, $x_1,\dots,x_n$, which take values in some alphabet of size q (WLOG considered to be $[q]$). We also have m *constraints* $(C_1,S_1),\dots,(C_m,S_m)$, where each C_i is a pair of variables (x_{i_1},x_{i_2}), and $S_i\subseteq[q]^2$.
A constraint (C_i,S_i) is said to be *satisfied* by an assignment $\nu:\{x_1,\dots,x_n\}\mapsto[q]$ if $(\nu(x_{i_1}),\nu(x_{i_2}))\in S_i$.
The algorithmic goal of a 2-CSP is to find an assignment that maximizes the number of constraints satisfied.

Example (Max-Cut is a 2-CSP)**.** For any graph $G(V,E)$, with $|V|=n,|E|=m$, let our alphabet be $\{0,1\}$ (ie:- $q=2$), and let our variables be $x_1,\dots,x_n$. Finally, for every $\{i,j\}\in E$, we have the constraint $((x_i,x_j),\{(0,1),(1,0)\})$.
It is easy to see that this CSP encodes the Max-Cut problem.

Example (Max-3-coloring is a 2-CSP)**.** The problem of Max-3-coloring asks for a 3-coloring of a given graph such that the number of edges having both endpoints of the same color is minimized.
For any graph $G(V,E)$, with $|V|=n,|E|=m$, let our alphabet be $\{1,2,3\}$ (ie:- $q=3$), and let our variables be $x_1,\dots,x_n$. Finally, for every $\{i,j\}\in E$, we have the constraint $((x_i,x_j),\{(\alpha,\beta):\alpha,\beta\in\{1,2,3\},\alpha\neq\beta\})$.
It is easy to see that this CSP encodes the Max-3-coloring problem.

Definition 6.2 (Promise Problem)**.** For $0\le s\le c\le 1$, the (c,s)-promise problem takes as input a 2-CSP instance, and the goal is to decide whether:

1. There exists an assignment which satisfies $\ge c$ fraction of constraints, or
2. Every assignment satisfies $\le s$ fraction of constraints.

Remark. A few remarks are in order:

1. If we have a $\left(\frac{s}{c}\right)$-approximation algorithm for a CSP, then that same algorithm can also decide the (c,s)-promise problem of that CSP.
2. A $\left(1,1-\frac{1}{m}\right)$-2CSP promise problem can be used to check the satisfiability of a CNF. Consequently, $\left(1,1-\frac{1}{m}\right)$-2CSP promise problems are NP-complete for $q\ge 3$.

Example. By Lemma 3.4, $(1-\varepsilon,1-\sqrt{\varepsilon})$-Max-Cut is in P.

Definition 6.3 (Unique 2-CSP)**.** A constraint $(C=(x,y),S)$ is called unique, if for every assignment of x, there is a unique assignment of y such that the constraint (C,S) is satisfied.
A 2-CSP is said to be unique if every constraint in it is unique.
A unique 2-CSP is also known as a *unique game*.

Remark. If (C,S) is a unique constraint, then $S=\{(i,\pi(i)):i\in[q]\}$, where π is a permutation of $[q]$.

Example. The following are some (non) examples of unique games:

1. Max-Cut is a unique game.

2. Max-2SAT asks for the assignment satisfying the maximum number of clauses in some given instance of a 2-SAT problem (in CNF form). Max-2SAT is *not* a unique game.

3. Consider the Max-2LIN problem, which gives us a prime p, and some linear equations of the form $x_i + x_j \equiv a_{ij} \bmod p$ or $x_i - x_j \equiv a_{ij} \bmod p$, and asks us to find an assignment $\{x_1, \dots, x_n\} \mapsto \mathbb{Z}/p\mathbb{Z}$ which maximizes the number of linear equations satisfied. Max-2LIN is a unique game.

Even though $(1, 1 - \frac{1}{m})$-2CSPs are NP-hard, $(1, 1 - \frac{1}{m})$ Unique Games are polynomial time decidable [21].

Theorem 6.1. It can be decided in polynomial time if a unique game with m constraints is satisfiable or not. Equivalently stated, $(1, 1 - \frac{1}{m})$-Unique Games are polynomial time decidable.

Proof. Form a graph G from all pairs that appear in constraints. For every connected component of G, apply the following algorithm:

1. Pick an arbitrary vertex u, and assign it some alphabet $\sigma \in [q]$. By the uniqueness constraint, this assignment of u forces an assignment of every vertex in the same connected component as u. If some vertex can't be consistently assigned, then there doesn't exist any satisfying assignment assigning u to σ. In that case, repeat the process, picking some $\sigma' \in [q] \setminus \{\sigma\}$.

Thus, in polynomial time, we can decide the satisfiability of a unique game. ■

We can finally state the Unique Games Conjecture.

Problem (Unique Games Conjecture)**.** For every $\varepsilon > 0$, there exists a $q = q(\varepsilon) \in \mathbb{N}$ such that $(1-\varepsilon, \varepsilon)$-UG is hard, ie:- it is hard to decide if $\geq 1 - \varepsilon$ fraction of constraints of some given UG (on an alphabet of size $\geq q(\varepsilon)$) are satisfiable, or if $\leq \varepsilon$ fraction are satisfiable.

However, note that the UGC is by no means the only way of approaching hardness-of-approximation results: Indeed, one of the biggest results in recent times is the PCP theorem, (a strong version of which, proven by Håstad in [Hås01]), states that $\left(1, \frac{7}{8} + \varepsilon\right)$-3SAT is NP-hard for every $\varepsilon > 0$ [22].

6.1. A History of the Unique Games Conjecture

All hardness of approximation results stated in this section hold provided the UGC holds.
The Unique Games Conjecture was proposed by Subhash Khot in [Kho02], who showed that $(1 - \varepsilon, 1 - \varepsilon^t)$-2LIN is NP-hard for all $\varepsilon > 0$ and $t \in \left[\frac{1}{2}, 1\right]$. Shortly after, Khot and Regev, in [KR03] proved that Vertex-Cover was hard to approximate within a factor of $(2 - \varepsilon)$ for any $\varepsilon > 0$ [23]. Shortly after, Khot et. al., in [KKMO04] showed that the Max-Cut problem is hard to approximate within a factor of $(\alpha_{GW} + \varepsilon)$ for any $\varepsilon > 0$.
The hardness of approximation for Max-Cut unveiled some interesting connections between 2-CSPs and Gaussian roundings. Work in this line culminated in Raghavendra's work ([Rag08]), which showed that for every CSP, the best possible approximation ratio is given by a corresponding SDP. This unified hardness of approximation results for many problems where the best approximation algorithms had been obtained by SDPs.
We now turn our attention to research gone into proving the UGC itself: In his original paper ([Kho02]), Khot proved

[21] this difference between CSPs and UGs is because of the extra knowledge of uniqueness that we have in the case of a UG, which allows for a polynomial time algorithm deciding the UG

[22] there is a trivial $\frac{7}{8}$-approximation algorithm for 3-SAT: Indeed, a random assignment of variables satisfies any clause of size 3 with probability $\frac{7}{8}$, and thus in expectation satisfies $\frac{7}{8}$-fraction of clauses. Thus Håstad's result is optimal

[23] A 2-approximation for vertex cover is very well known: Indeed, greedily construct a maximal matching on the given graph. The vertices of the matching form a vertex cover of size at most twice that of the smallest possible vertex cover

that $\left(1-\varepsilon, 1-\mathcal{O}\left(q^2\varepsilon^{\frac{1}{5}}\sqrt{\log\frac{1}{\varepsilon}}\right)\right)$-UG lies in P. Thereafter, this result was improved by a long line of publications, and finally in 2006, it was shown by Charikar-Makarychev-Makarychev [CMM06] that $\left(1-\varepsilon, 1-\mathcal{O}\left(\sqrt{\varepsilon\log q}\right)\right)$-UG lied in P, and furthermore, they also showed that, if the UGC was true, then their result couldn't be improved upon. [KKMO04] corroborated this in 2007, when they proved that $\left(1-\varepsilon, 1-\sqrt{\frac{2}{\pi}}\sqrt{\varepsilon\log q}+\varepsilon\right)$-UG was NP-hard. Interestingly, the [CMM06] result just used a (variant of) degree 2 Sum of Squares algorithm.

Now, the UGC posits that $(1-\varepsilon, \varepsilon)$-UG is NP-hard, but it doesn't say anything about the exponential complexity of the problem: Progress was made in this line by Arora, Barak, and Steurer [ABS15], who showed that $\left(1-\varepsilon, \frac{1}{2}\right)$-UG could be solved in $2^{q^2 n^{\mathcal{O}(\varepsilon^{1/3})}}$ time, which is better than the naïve $2^{\mathcal{O}(n)}$ time algorithm.

The [ABS15] result is interesting when one takes a different perspective: Consider the Exponential Time Hypothesis propounded by Impagliazzo, Kabanets, and Wigderson [IKW01], which hypothesizes that there is no $2^{o(n)}$ algorithm for solving 3-SAT.

Now, it can be shown that any 3-SAT instance can be reduced to a $\left(1-\varepsilon, \frac{1}{2}\right)$-UG instance in $\mathrm{poly}(n)$ time: Thus, if UG were solvable in $2^{n^{o(1)}}$ time, then 3-SAT would be solvable in $2^{n^{o(1)}}$ time too, contradicting the Exponential Time Hypothesis.

Consequently, there are good reasons to believe that the [ABS15] results are close to optimal for $\left(1-\varepsilon, \frac{1}{2}\right)$-UG.

A breakthrough regarding the UGC was made in 2018 when Dinur, Khot, Kindler, Minzer, and Safra [DKK+18] settled the *2-to-2 Games conjecture*, which deals with constraints where fixing one variable leaves us with 2 choices for the other variables (as opposed to a unique choice in case of a unique constraint). One consequence of their result is that $\left(\frac{1}{2}-\varepsilon, \varepsilon\right)$-UG is NP-hard.

§7. Lower Bounds Through Sum of Squares

Recall, in Section 3, how we showed that our degree 2 SoS analysis of the Max-Cut problem was optimal, by presenting the cycle as a barrier to any improvement through degree 2 Sum-of-Squares alone.
Continuing a similar line of thought, we shall now see a class of problems, which are provably difficult to deal with the Sum-of-Squares hierarchy, ie:- we'll delineate some limitations of the Sum-of-Squares hierarchy.

7.1. k-XOR is hard using SoS

Problem (k-XOR Problem)**.** Suppose we have n variables $z_1, \ldots, z_n \in \mathbb{F}_2$, and m equations on these variables, where every equation is of the form $z_{i_1} + z_{i_2} + \ldots + z_{i_k} = 0$ or 1. Thus every equation involves exactly k variables.
The algorithmic goal of this problem is to maximize the number of equations satisfied.

If we set $x_i = (-1)^{z_i}$, then our equations take the form $x_{i_1} x_{i_2} \cdots x_{i_k} = \pm 1$. Thus, an instance of k-XOR is given by m sets $C_1, C_2, \ldots, C_m \subseteq [n]$, with $|C_j| = k$ for every $j \in [m]$, and for every set C_j, we have the corresponding equation $\prod_{i \in C_j} x_i = b_i \in \{-1, 1\} \iff b_i \prod_{i \in C_j} x_i = 1$.
We denote the constraints $(C_j, b_j)_{j \in [m]}$ by $\mathcal{I}$. For $x \in \mathbb{R}^n$, define

$$\mathcal{I}(x) := \frac{1}{m} \sum_{j=1}^{m} b_j \prod_{i \in C_j} x_i = \frac{1}{m} \sum_{j=1}^{m} b_j x_{C_j}$$

Note that if $x \in \{-1, 1\}^n \subseteq \mathbb{R}^n$, then

$$\mathrm{Val}_{\mathcal{I}}(x) := \frac{1 + \mathcal{I}(x)}{2}$$

gives us the fraction of equations satisfied by the assignment x. Thus the goal of k-XOR problem can be said to be approximating $\mathrm{opt}(\mathcal{I}) := \max_{x \in \{-1,1\}^n} \mathrm{Val}_{\mathcal{I}}(x)$.
Note that if $\mathrm{opt}(\mathcal{I}) = 1$, then Gaussian elimination also yields x for which the optimum is attained.
Also note that for any $\mathcal{I}$, a random assignment of variables satisfies half of the equations in expectation, and thus we have a trivial $\frac{1}{2}$-approximation algorithm[24]. However, unlike the Max-Cut problem, we run out of luck when trying to improve this approximation algorithm, due to the following theorem of Håstad ([Hås01]):

Theorem 7.1. For any $\varepsilon > 0$ and any $k \geq 3$, it is NP-hard to decide if $\mathrm{opt}(\mathcal{I}) \geq 1 - \varepsilon$ or $\mathrm{opt}(\mathcal{I}) \leq \frac{1}{2} + \varepsilon$ for some given k-XOR instance $\mathcal{I}$.

Stated differently, assuming P $\neq$ NP, there is no polynomial time algorithm to find a $\left(\frac{1}{2} + 2\varepsilon\right)$-satisfying assignment for some $\mathcal{I}$ such that $\mathrm{opt}(\mathcal{I}) = 1 - \varepsilon$. Equivalently, for $k \geq 3$, a $\frac{1}{2}$-approximation algorithm is the best we can get for k-XOR. Problems such as k-XOR are thus called *approximation resistant*.
Now, the reader may be excused for feeling cheated at this juncture: We promised that the k-XOR problem would underline some fundamental limitation of the Sum-of-Squares hierarchy, but that hardly seems to be the case, because (a $> \frac{1}{2}$-approximation algorithm for) k-XOR is resistant against *all* polynomial time schemes, not just SoS, and thus it does seem a bit unfair to give k-XOR as an example of the limitations of SoS.
Nevertheless, there is a more philosophical way of interpreting the results below: Given how successful the SoS hierarchy has been, in giving approximation algorithms and hardness of approximation results (which is not surprising in light of Raghavendra's result [Rag08]), if some problem can't be attacked using SoS, then that gives us a good indication that there may be no other ways to attack the problem. So, at the very least, hardness results of SoS develop our intuition about problems whose difficulty is yet unknown.
We shall now finally state our hardness result without any ado.

[24] if this approximation algorithm sounds awfully similar to the $\frac{1}{2}$-approximation algorithm for Max-Cut, that is because Max-Cut is just 2-XOR with all $b_j = -1$.

Theorem 7.2 (Grigoriev's Theorem)**.** For any $k \geq 3$, $c < 2$, there exists a constant $c' = c'(k)$ and a family of k-XOR instances $(\mathcal{I}_n)_{n\in\mathbb{N}}$ such that

$$c \cdot \text{opt}(\mathcal{I}_n) - \text{Val}_{\mathcal{I}_n}(x)$$

has no degree $(c'n)$-SoS certificate for large enough n.

Thus a $\frac{1}{c}$-approximation algorithm, obtained through SoS, for $c < 2$, yields a $\text{poly}(n^r)$ time algorithm, where $r = \mathcal{O}(n)$ by the above theorem. Thus any SoS algorithm for k-XOR, which approximates better than a factor of 2 must take $\text{poly}(n^{\mathcal{O}(n)})$ time.
We'll prove the above theorem by the probabilistic method. Let Δ be some parameter to be set later. Every k-sized subset of $[n]$ is taken to be a constraint of $\mathcal{I}_n$ with probability $\frac{n\Delta}{\binom{n}{k}}$ [25]. The corresponding bit 'b' of the constraint is taken to be ± 1 with equal probability.
We then prove Theorem 7.2 by demonstrating a degree $(c'n)$ pseudo-distribution μ such that

$$\widetilde{\mathbb{E}}_\mu[c \cdot \text{opt}(\mathcal{I}_n) - \text{Val}_{\mathcal{I}_n}(x)] < 0 \iff \widetilde{\mathbb{E}}_\mu[\text{Val}_{\mathcal{I}_n}(x)] > c \cdot \text{opt}(\mathcal{I}_n) \tag{7.1}$$

Our strategy to demonstrate such a μ goes as follows: We actually construct a μ such that $\widetilde{\mathbb{E}}_\mu[\text{Val}_{\mathcal{I}_n}(x)] = 1 \iff \widetilde{\mathbb{E}}_\mu[\mathcal{I}_n(x)] = 1$. The following lemma then shows that such a μ, with high probability, satisfies Eq. (7.1).

Lemma 7.3. There exists a constant $D > 0$, such that if $\Delta \geq \frac{D}{\varepsilon^2}$ and $\widetilde{\mathbb{E}}_\mu[\text{Val}_{\mathcal{I}_n}(x)] = 1$, then $\text{opt}(\mathcal{I}_n(x)) \leq \frac{1}{2} + \varepsilon$ with probability ≥ 0.99.

Proof Sketch. Fix any assignment $y \in \{-1, 1\}^n$. Since each bit value in $\mathcal{I}_n$ was sampled i.i.d with probability $\frac{1}{2}$, each constraint in $\mathcal{I}_n$ is satisfied with probability $\frac{1}{2}$, and furthermore, one constraint being satisfied is independent of some other constraint being satisfied.
Thus the satisfaction of constraints in $\mathcal{I}_n$ are i.i.d Bernoulli random variables with parameter $\frac{1}{2}$. Then the lemma follows by a routine application of Chernoff bounds. ■

Now, by Lemma 1.9, WLOG we can assume μ to be $(c'n)$-degree multilinear polynomial. Furthermore, the proof technique of Lemma 1.9 also shows that to calculate $\widetilde{\mathbb{E}}_\mu[f]$ for any function f (expressed as a multilinear polynomial), it suffices to consider only the degree $\leq c'n$ terms in f, ie:- to describe $\widetilde{\mathbb{E}}_\mu[\cdot]$, it suffices to describe $\widetilde{\mathbb{E}}_\mu[x_S]$ for $|S| \leq c'n$.
Now, note that

$$\widetilde{\mathbb{E}}_\mu[\mathcal{I}(x)] = \frac{1}{m}\sum_{j=1}^{m} b_j \widetilde{\mathbb{E}}_\mu[x_{C_j}] \implies |\widetilde{\mathbb{E}}_\mu[\mathcal{I}(x)]| \leq \frac{1}{m}\sum_{j=1}^{m} |b_j| \cdot \underbrace{|\widetilde{\mathbb{E}}_\mu[x_{C_j}]|}_{\leq 1 \text{ by Lemma 1.6}} \leq 1$$

Thus, if we are to have $\widetilde{\mathbb{E}}_\mu[\mathcal{I}(x)] = 1$, then $\widetilde{\mathbb{E}}_\mu[x_{C_j}] = b_j$ for every $j \in [m]$.
Similarly,

$$1 = \widetilde{\mathbb{E}}_\mu[\mathcal{I}(x)]^2 = \left(\frac{1}{m}\sum_{j=1}^{m} b_j \widetilde{\mathbb{E}}_\mu[x_{C_j}]\right)^2 = \frac{1}{m^2}\left(\widetilde{\mathbb{E}}_\mu\left[\sum_{j=1}^{m} b_j x_{C_j}\right]\right)^2 \overset{\text{Jensen's Inequality}}{\leq} \frac{1}{m^2}\widetilde{\mathbb{E}}_\mu\left[\left(\sum_{j=1}^{m} b_j x_{C_j}\right)^2\right] \tag{7.2}$$

$$= \frac{1}{m^2}\sum_{j,\ell\in[m]} b_j b_\ell \widetilde{\mathbb{E}}_\mu\left[x_{C_j} x_{C_\ell}\right] \leq \frac{1}{m^2}\sum_{j,\ell\in[m]} |b_j b_\ell| \cdot \left|\widetilde{\mathbb{E}}_\mu\left[x_{C_j} x_{C_\ell}\right]\right| \leq 1 \tag{7.3}$$

Thus, if we are to have $\widetilde{\mathbb{E}}_\mu[\mathcal{I}(x)] = 1$, then we must also have $\widetilde{\mathbb{E}}_\mu[x_{C_j} x_{C_\ell}] = b_j b_\ell$ for every $j, \ell \in [m]$.
We now analyze the so-called *degree d derivation* to obtain our desired μ.

[25] consequently, in expectation, there are $n\Delta$ constraints in $\mathcal{I}_n$

7.2. Degree d Derivations

Set $d = c'n$ for convenience.
Define Der_d to be the output of the following process:

1. Set $\mathrm{Der}_d \leftarrow \varnothing$. Der_d should be imagined as a set of equations.
2. For each $j \in [m]$, add $x_{C_j} = b_j$ to Der_d.
3. Traverse through all monomials $x_S, |S| \leq d$ in some fixed order, where monomials of lower degrees are processed before monomials of higher degrees. For each x_U in this traversal, if $x_S = b_S$ and $x_T = b_T$ belong to Der_d, such that $S \oplus T = U$ [26], then we add $x_U = b_S b_T =: b_U$ to Der_d.
4. Finally, for any $S \subseteq [n], |S| \leq d$, set $\widetilde{\mathbb{E}}_\mu[x_S] = b_S$.

Now, note that there can be potential "conflicts" in the process described above: Indeed, consider an example where $U = \{1,2,3,4\}, S_1 = \{1,2\}, T_1 = \{3,4\}, S_2 = \{1,3\}, T_2 = \{2,4\}, b_{S_1} = b_{T_1} = b_{S_2} = 1 = -b_{T_2}$. Clearly, $S_1 \oplus T_1 = U = S_2 \oplus T_2$, yet $b_{S_1} b_{T_1} \neq b_{S_2} b_{T_2}$.
We show that with high probability, conflicts mentioned above don't occur [27].

7.2.1. Conflicts don't happen in a degree d derivation

Definition 7.1 (Uniform Hypergraphs)**.** A k-uniform hypergraph $\mathcal{H}$ on the vertex set V is a collection of hyperedges, where every hyperedge is a subset of V of size k.

Example. A 3-uniform hypergraph on the vertex set $[6]$: $\{\{1,2,3\},\{2,4,5\},\{3,4,6\},\{1,5,6\}\}$.

Definition 7.2 (Hypergraph Expansion)**.** A k-uniform hypergraph on $[n]$ is said to be (t,β)-expanding if for every subset $\mathcal{C}$ of at most t hyperedges,

$$\left|\bigcup_{e \in \mathcal{C}} e\right| \geq \beta |\mathcal{C}|$$

The constraints $C_j, j \in [m]$ of a k-XOR instance $\mathcal{I}_n$ form a k-uniform hypergraph with m edges. We call a k-XOR instance (t,β)-expanding if its underlying hypergraph is.
We shall now state a lemma (without proof), which essentially says that randomly generated k-XOR instances are very good expanders.

Lemma 7.4. Let $\mathcal{I}_n$ be the random k-XOR instance as constructed in the previous section. Then for all $\delta > 0$, there exists $\eta = \eta(\delta, \Delta) > 0$, such that with probability ≥ 0.99, $\mathcal{I}_n$ is $(\eta n, k - 1 - \delta)$-expanding.

We will now use the above lemma to show that there are no conflicts.

Lemma 7.5. Suppose $\mathcal{I}_n$ is $(\eta n, \alpha)$-expanding, where $\alpha \in \left(\frac{k}{2} + \frac{1}{10}, k - 1\right)$. Then, for $d < \frac{n\eta}{100}$, there don't exist S_1, T_1, S_2, T_2 in Der_d such that $S_1 \oplus T_1 = S_2 \oplus T_2$. Consequently, Der_d doesn't have any conflicts.

[26] recall that $X \oplus Y$ denotes the symmetric difference of the sets X and Y. If $X \oplus Y = Z$, then $x_X \cdot x_Y = x_Z$.
[27] recall that C_j, b_j are random quantities, and thus our statement holds only with high probability

Proof. Define $\mathcal{D} := \{U \subseteq [n] : (x_U = b_U) \in \mathrm{Der}_d\}$.
Assume for the sake of contradiction we have S_1, T_1, S_2, T_2 such that $S_1 \oplus T_1 = S_2 \oplus T_2$.
Note that every set in $\mathcal{D}$ is composed by taking symmetric differences of the constraint sets $C_j, j \in [m]$. Thus, for $S \in \{S_1, T_1, S_2, T_2\}$, we define $U_S := \{C_\ell : \ell \in [m]\}$ where $S = \bigoplus_{C \in U_S} C$.
Now, consider

$$U := \bigoplus_{S \in \{S_1, T_1, S_2, T_2\}} U_S$$

where $U_X \oplus U_Y$ has the usual meaning (one should always keep in mind that the $\oplus$ operator on sets is really taking the product of variables in their multiset union and performing a multilinear reduction on them).
Now, since $S_1 \oplus T_1 = S_2 \oplus T_2$, each variable occurs an even number of times across all constraints in the U_*'s ($* \in \{S_1, T_1, S_2, T_2\}$). ■

Thus we can successfully carry out the degree d derivation without any conflicts, which means that for any $S \subseteq [n], |S| \leq d$, for which $S \in \mathcal{D}$, we can consistently set $\widetilde{\mathbb{E}}_\mu[x_S] = b_S$. For $S \notin \mathcal{D}$, we set $\widetilde{\mathbb{E}}_\mu[x_S] = 0$ [28]. The only thing remaining to be shown is that μ is in fact a degree d pseudo-distribution.

Theorem 7.6. Let $\mu \in \mathbb{R}^{\{-1,1\}^n}$ be a function such that $\widetilde{\mathbb{E}}_\mu[x_S] = b_S$ for all $S \in \mathcal{D}$, and $\widetilde{\mathbb{E}}_\mu[x_S] = 0$ for $S \subseteq [n], |S| \leq d, S \notin \mathcal{D}$. Then μ is in fact a degree d pseudo-distribution such that $\widetilde{\mathbb{E}}_\mu[\mathrm{Val}_{\mathcal{I}_n}(x)] = 1$.

Remark. Note that *a priori*, we don't know if μ is a pseudo-distribution or not. Nevertheless, we'll continue to use the notation $\widetilde{\mathbb{E}}_\mu[\cdot]$ to denote the formal expectation operator w.r.t μ.

Proof. Note that by its very definition, $\widetilde{\mathbb{E}}_\mu[\mathrm{Val}_{\mathcal{I}_n}(x)] = \widetilde{\mathbb{E}}_\mu[\mathcal{I}_n(x)] = 1$, as desired.
Furthermore, $|\widetilde{\mathbb{E}}_\mu[x_S]| \leq 1$ for $S \subseteq [n], |S| \leq d$. Consequently, we can mimic Eq. (7.2) and get that $\widetilde{\mathbb{E}}_\mu[x_{C_j} x_{C_\ell}] = 1$ for every $j, \ell \in [m]$. Setting $j = \ell$ yields that $\widetilde{\mathbb{E}}_\mu[x_{C_j}^2] = 1 \implies \widetilde{\mathbb{E}}_\mu[1] = 1$.
Now, let p be a multilinear polynomial of degree $\leq \frac{d}{2}$. Then we must show that $\widetilde{\mathbb{E}}_\mu[p^2] \geq 0$.
Consider the relation $\sim$ on $[n]_{\frac{d}{2}} := \{T \subseteq [n] : |T| \leq \frac{d}{2}\}$, where $T_1 \sim T_2$ if $\widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_2}] \neq 0$. We claim that $\sim$ is an equivalence relation. The symmetry of $\sim$ is obvious, and the reflexivity of $\sim$ follows from the fact that $\widetilde{\mathbb{E}}_\mu[1] = 1$. Finally, if $T_1 \sim T_2$ and $T_2 \sim T_3$, then

$$\widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_3}] = \widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_2} x_{T_2} x_{T_3}] = \underbrace{\widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_2}]}_{\neq 0} \cdot \underbrace{\widetilde{\mathbb{E}}_\mu[x_{T_2} x_{T_3}]}_{\neq 0} \neq 0$$

where the second equality follows from the definition of Der_d.
Let the equivalence classes of $[n]_{\frac{d}{2}}$ under $\sim$ be $Q_1, \ldots, Q_r$. Then, decompose p as

$$p(x) = \sum_{i \in [r]} \sum_{S \in Q_i} p_S x_S = \sum_{i \in [r]} p_i(x)$$

Thus

$$\widetilde{\mathbb{E}}_\mu[p^2] = \sum_{i=1}^{r} \widetilde{\mathbb{E}}_\mu[p_i^2] + \sum_{i,j \in [r]} \widetilde{\mathbb{E}}_\mu[p_i p_j] = \sum_{i=1}^{r} \widetilde{\mathbb{E}}_\mu[p_i^2]$$

where the second equality follows from the fact that $\widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_2}] = 0$ if T_1, T_2 don't belong to the same equivalence class. Now, fix any $i \in [r]$, and let $\mathcal{T}$ be an arbitrary member of Q_i. Then note that

$$\widetilde{\mathbb{E}}_\mu[p_i^2] = \widetilde{\mathbb{E}}_\mu\left[\left(\sum_{T \in Q_i} p_T x_T\right)^2\right] = \sum_{T_1, T_2 \in Q_i} p_{T_1} p_{T_2} \widetilde{\mathbb{E}}_\mu[x_{T_1} x_{T_2}] = \sum_{T_1, T_2 \in Q_i} p_{T_1} p_{T_2} \widetilde{\mathbb{E}}_\mu[x_{T_1} x_{\mathcal{T}} x_{\mathcal{T}} x_{T_2}]$$

[28] This choice is essentially borne out of laziness. Since we don't know anything about the pseudo-expectation of μ over S, we may as well assume that μ is 'unbiased' for x_S

$$= \sum_{T_1, T_2 \in Q_i} p_{T_1} p_{T_2} \widetilde{\mathbb{E}}_\mu[x_{T_1} x_{\mathcal{T}}] \widetilde{\mathbb{E}}_\mu[x_{\mathcal{T}} x_{T_2}] = \left(\sum_{T \in Q_i} p_T \widetilde{\mathbb{E}}_\mu[x_T x_{\mathcal{T}}] \right)^2 \geq 0$$

as desired. ■

§8. SoS vs. Spectral Algorithms

As we noted in the proof of Grigoriev's theorem, small degree Sum-of-Squares is unable to solve even average case instances of the k-XOR problem. Stated differently, even the *average case complexity* of the k-XOR problem is hard for SoS.
We now give some motivation for the notion of *average case hardness*: Traditional complexity analysis of an algorithm is *worst case analysis*. We analyze what the worst possible performance of our algorithm could be under some (adversarially chosen) input, and based on that worst-case performance, we rate the efficiency of the algorithm in general. It is clear that the above measure of judging an algorithm might be too pessimistic: Indeed, the worst case complexity of quick sort is $\mathcal{O}(n^2)$, yet, *in practice*, quick sort is one of the fastest sorting algorithms known.
Thus to remedy this situation, the notion of average case analysis was introduced in computer science: In average case analysis, we judge the performance of an algorithm based on the *expected* performance of the algorithm on some input chosen from some distribution $\mathcal{D}$, where $\mathcal{D}$ is designed so as to represent some "natural/common distribution of inputs".
Average case algorithms may be further divided into 3 classes based on the nature of the distribution $\mathcal{D}$:

1. **Refutation Problems**: In this class of problems, we have to produce a *certificate* for the non-existence of some entity: For example, refuting random CSPs entails producing a certificate of unsatisfiability. In this light, Grigoriev's theorem can be interpreted as saying that SoS can't be used for (efficiently) refuting random CSPs. Furthermore, it is usually assumed that with high probability, an input instance from $\mathcal{D}$ is "unsatisfiable". Such distributions are usually denoted as "$\mathcal{D}_{\text{null}}$".

2. **Planted Problems**: In this class of problems, we have to find some particular entity in the input, provided it is known that with high probability, the input indeed possesses the particular entity. For example, suppose we sample graphs from some probability distribution, where with high probability, every graph has a clique of size $\mathcal{O}(\log n)$ within it. Then the planted problem in this context would be to find a $\mathcal{O}(\log n)$-sized clique in some graph sampled from the said distribution.

3. **Distinguishing Problems**: Suppose we have two distributions, $\mathcal{D}_{\text{null}}$ and $\mathcal{D}_{\text{planted}}$. For example, consider two distributions, the first one (which we call $\mathcal{D}_{\text{null}}$) containing graphs, which, with high probability *don't* contain a clique of size $\mathcal{O}(\log n)$. The second distribution, which we call $\mathcal{D}_{\text{planted}}$, contains, with high probability, graphs that have cliques of size $\mathcal{O}(\log n)$ within them.
   Given two graphs G_1, G_2, one sampled from $\mathcal{D}_{\text{null}}$ and the other sampled from $\mathcal{D}_{\text{planted}}$, we have to tell which graph was sampled from which distribution.

In this chapter, we shall look at the 3 different "flavors" of some average-case problems, and we shall see how SoS fares in them. In the process, we shall also introduce *spectral algorithms*. So without ado, let's begin!

8.1. The Max-Clique Problem

Consider the Erdős-Rényi graph $G = G\left(n, \frac{1}{2}\right)$, where there is an edge between any two vertices with probability $\frac{1}{2}$. A folklore result of Bollobás about the size of the maximum clique in G goes as follows.

Theorem 8.1. With probability $1 - o(1)$, the size of the maximum clique in G is $\lfloor 2\log_2(n) \rfloor$ or $\lfloor 2\log_2(n) + 1 \rfloor$.

Thus, in light of Bollobás's theorem, a refutation problem may be framed as follows: Let $\omega \in \mathbb{N}$ be greater than $\lfloor 2\log_2(n) + 1 \rfloor$. Given an Erdős-Rényi random graph, produce a certificate for the non-existence of a clique of size ω. A brute force algorithm for this problem takes $\mathcal{O}(n^\omega) = \Omega(n^{\log n})$ time, which is clearly unacceptable.
We will now study our first *spectral algorithm* for this refutation problem.
Let G be the input graph, and define $A \in \mathbb{R}^{n \times n}$ as follows:

$$A_{ij} := \begin{cases} 1, i \text{ and } j \text{ adjacent in } G \\ -1, \text{otherwise} \end{cases}$$

Let $x \in \{0,1\}^n$ be the indicator vector of some clique $C \subseteq [n]$ in G. Then note that

$$x^\mathsf{T} A x = \left(\sum_{i=1}^n x_i\right)^2 - 2\sum_{i=1}^n x_i^2 = \left(\sum_{i=1}^n x_i\right)^2 - 2\sum_{i=1}^n x_i = |C|^2 - 2|C|$$

But by the definition of spectral norms, we also have that

$$x^\mathsf{T} A x \leq \|x\|_2^2 \cdot \|A\|_2 = |C| \cdot \|A\|_2$$

Thus

$$|C|^2 - 2|C| \leq |C| \cdot \|A\|_2 \implies |C| \leq 2 + \|A\|_2$$

Thus our spectral refutation algorithm goes as follows: Calculate $\|A\|_2$ [29], and return $\mathcal{C} := \|A\|_2 + 2$. $\mathcal{C}$ is an upper bound for the maximum clique size of G.
Now, it is a classical result in random matrix theory (see [AGZ09]. Also see [Tao11], [TV08]), that the spectral norm of a symmetric Rademacher random matrix, such as A, converges to the so-called "semi-circular" distribution over an interval of size $\Theta(\sqrt{n})$.
Thus, if $\omega \gg \sqrt{n}$, then the spectral norm of A serves as a refutation of the hypothesis that there is some clique of size ω. Conversely, if $\omega = o(\sqrt{n})$, then for a non-negligible fraction [30] of Erdős-Rényi graphs, $\|A\|_2 + 2$ will fail to refute the existence of some clique of size ω.
What is surprising is that the above $\sqrt{n}$ bound is essentially the best known (asymptotically) refutation algorithm for the Max-Clique problem. Notice the huge gap between the reality (which says that there are no cliques of size $\geq 2\log_2(n) + 2$), and what we can actually certify in polynomial time ($\sqrt{n}$). In fact, an algorithm for refuting even clique sizes of $\omega = n^{0.49}$ is not known.
Such a gap between the size of some quantity (in this case the size of the Max-Clique), and what we can actually compute/certify in polynomial time, is known as an *information computation gap*.
Thus we saw a spectral algorithm, and saw how it (asymptotically) establishes the best bounds for the Max-Clique refutation algorithm. As usual, we now wish to subsume the class of spectral algorithms under the Sum-of-Squares hierarchy.

8.2. Sum of Squares Derivations in $\{0,1\}^n$

Since we are working with characteristic vectors of sets, we shift our attention to Sum-of-Squares proofs on $\{0,1\}^n$, instead of our usual boolean hypercube.

Lemma 8.2. Let μ be a degree 2 pseudo-distribution on $\{0,1\}^n$. Then $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T} M x] \leq \|M\|_2 \cdot \widetilde{\mathbb{E}}_\mu[\|x\|_2^2]$.

Proof. We have to prove that $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}(\|M\|_2 - M)x] \geq 0$. Now, note that $(\|M\|_2 - M)$ is a PSD matrix, and thus, by Lemma 0.3, $\|M\|_2 - M = B^\mathsf{T} B$ for some matrix B. But then $x^\mathsf{T}(\|M\|_2 - M)x = \|Bx\|_2^2$, which is the sum of squares of linear polynomials, and thus has non-negative pseudo-expectation. ■

Lemma 8.3. Let μ be a degree 2 pseudo-distribution on $\{0,1\}^n$ such that $\widetilde{\mathbb{E}}_\mu[x_i x_j] = 0$ for every $i \neq j$ which are not adjacent in our graph G (we denote adjacency as $\sim$).
Then $\widetilde{\mathbb{E}}_\mu\left[\|x\|_2^4\right] \leq \widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2(2 + \|A\|_2)\right]$.

[29] the spectral norm of A can be calculated in polynomial time (see power iteration), so our refutation algorithm is indeed polynomial time
[30] more precisely, a $1 - \mathcal{O}\left(\frac{\omega^2}{n}\right)$ fraction

Proof. Note that

$$\widetilde{\mathbb{E}}_\mu[\|x\|_2^4] = \widetilde{\mathbb{E}}_\mu\left[\left(\sum_{i=1}^n x_i^2\right)^2\right] = \widetilde{\mathbb{E}}_\mu\left[\left(\sum_{i=1}^n x_i\right)^2\right]$$

$$\widetilde{\mathbb{E}}_\mu\left[\left(\sum_{i=1}^n x_i\right)^2\right] = \widetilde{\mathbb{E}}_\mu\left[\sum_{i=1}^n x_i^2\right] + 2\widetilde{\mathbb{E}}_\mu\left[\sum_{i\sim j} x_i x_j\right] + 2\widetilde{\mathbb{E}}_\mu\left[\sum_{\substack{i\not\sim j\\ i\neq j}} x_i x_j\right] = \widetilde{\mathbb{E}}_\mu\left[\sum_{i=1}^n x_i^2\right] + 2\widetilde{\mathbb{E}}_\mu\left[\sum_{i\sim j} x_i x_j\right]$$

$$= \widetilde{\mathbb{E}}_\mu\left[\sum_{i=1}^n x_i^2\right] + 2\widetilde{\mathbb{E}}_\mu\left[\sum_{i\sim j} x_i x_j\right] - 2\widetilde{\mathbb{E}}_\mu\left[\sum_{\substack{i\not\sim j\\ i\neq j}} x_i x_j\right] = 2\widetilde{\mathbb{E}}_\mu\left[\sum_{i=1}^n x_i^2\right] + \widetilde{\mathbb{E}}_\mu\left[x^\mathsf{T} A x\right]$$

where $A \in \{-1,1\}^{n\times n}$ is the matrix we defined earlier.
Since μ is a degree 2 pseudo-distribution, by Lemma 8.2, $\widetilde{\mathbb{E}}_\mu[x^\mathsf{T}Ax] \leq \widetilde{\mathbb{E}}_\mu[\|A\|\cdot\|x\|_2^2]$, and thus we have our desired result. ■

Finally, we get rid of the 4^{th} power by the pseudo-distribution Cauchy-Schwarz inequality.

Lemma 8.4 (Cauchy-Schwarz Inequality)**.** Let μ be a pseudo-distribution of degree d, and let p, q be polynomials of degree $\leq \frac{d}{2}$ on $\{0,1\}^n$. Then

$$\widetilde{\mathbb{E}}_\mu[pq]^2 \leq \widetilde{\mathbb{E}}_\mu[p^2]\cdot\widetilde{\mathbb{E}}_\mu[q^2]$$

Proof. It can be easily seen that $\widetilde{\mathbb{E}}_\mu[pq]^2 \leq \widetilde{\mathbb{E}}_\mu[p^2]\cdot\widetilde{\mathbb{E}}_\mu[q^2]$ is equivalent to

$$\sum_{x,y\in\{0,1\}^n} \mu(x)\mu(y)(p(x)q(y)-p(y)q(x))^2 \geq 0 \iff \widetilde{\mathbb{E}}_\mu\left[(p(x)q(y)-p(y)q(x))^2\right] \geq 0$$

Now, note that the both the x-degree and the y-degree of the polynomial $p(x)q(y)-p(y)q(x)$ is bounded by $\frac{d}{2}$. Also note that

$$\widetilde{\mathbb{E}}_\mu\left[(1,x,1,y)^{\otimes\frac{d}{2}}((1,x,1,y)^{\otimes\frac{d}{2}})^\mathsf{T}\right] = \left(\widetilde{\mathbb{E}}_\mu\left[(1,x)^{\otimes\frac{d}{2}}((1,x)^{\otimes\frac{d}{2}})^\mathsf{T}\right]\right)\bigotimes\left(\widetilde{\mathbb{E}}_\mu\left[(1,y)^{\otimes\frac{d}{2}}((1,y)^{\otimes\frac{d}{2}})^\mathsf{T}\right]\right)$$

Since the tensor product of PSD matrices is PSD, it follows that $\widetilde{\mathbb{E}}_\mu\left[(p(x)q(y)-p(y)q(x))^2\right] \geq 0$, as desired. ■

Theorem 8.5. Let μ be a degree 4 pseudo-distribution on $\{0,1\}^n$ such that $\widetilde{\mathbb{E}}_\mu[x_ix_j] = 0$ for every $i \neq j$ which are not adjacent in our graph G (we denote adjacency as $\sim$).
Then $\widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2\right] \leq (2+\|A\|_2)$.

Proof. By Cauchy-Schwarz, $\widetilde{\mathbb{E}}_\mu[\|x\|_2^4] \geq \widetilde{\mathbb{E}}_\mu[\|x\|_2^2]^2$. Thus $\widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2(2+\|A\|_2)\right] \geq \widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2\right]^2 \implies \widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2\right] \leq (2+\|A\|_2)$, as desired. ■

Consequently, if we can find a degree 4 pseudo-distribution μ maximizing the quantity $\widetilde{\mathbb{E}}_\mu[\|x\|_2^2]$ under the constraints $\widetilde{\mathbb{E}}_\mu[x_ix_j] = 0$ for $i \not\sim j, i \neq j$ [31], then we have a SoS refutation proof in our hands: Indeed, note that if $x \neq 0$ is such that $\mathbb{E}[x_ix_j] = 0$ for non-adjacent i, j, then the maximum value of $\mathbb{E}_\nu[\|x\|_2^2]$ for *actual* distributions ν is the

[31] pseudo-expectations with these additional constraints can also be maximized in polynomial time, à la Theorem 2.6

size of the maximum clique in G (and the corresponding maximizer is the characteristic vector of the maximum clique). Since actual distributions are also pseudo-distributions, the maximum value of $\widetilde{\mathbb{E}}_\mu[\|x\|_2^2]$ exceeds the maximum clique size, and thus serves a refutatory purpose. Furthermore, the inequality $\widetilde{\mathbb{E}}_\mu\left[\|x\|_2^2\right] \leq (2 + \|A\|_2)$ also establishes that our SoS algorithm is atleast as good as the spectral algorithm described above.
Note that this exact algorithm also helps us solve the planted version of the max-clique problem.
The planted version goes as follows.

Problem (Planted Version of Max-Clique)**.** Let G be a randomly sampled Erdős-Rényi graph, and let $\omega \gg 2\log_2(n)$. We randomly select a subset $S \subseteq [n]$ such that $|S| = \omega$, and we add all the (missing) edges in S to make S a clique. We then gives this modified graph as an input to our algorithm, which must find out this "planted" clique.

It is quite clear how our SoS algorithm solves this planted problem: as we noted earlier, the maximizer of $\widetilde{\mathbb{E}}_\mu[\|x\|_2^2]$ is the indicator vector of the largest clique, which in the case of the planted problem, is S (with very high probability). Thus the maximizer of the SDP program $\widetilde{\mathbb{E}}_\mu[\|x\|_2^2]$ is the required planted clique.
Similarly, we may also use either the SoS algorithm or the spectral algorithm to solve the so-called distinguishing problem.

Problem (Distinguishing Version of Max-Clique)**.** Let G_1 be a randomly sampled Erdős-Rényi graph.
Sample G_2 as an Erdős-Rényi graph, independently of G_1, and let G_2' be the planted version of G_2 for some $\omega \gg \sqrt{n}$.
Given G_1, G_2', we have to tell which graph is the planted graph.

Once again, if we calculate $2 + \|A_1\|$ and $2 + \|A_2'\|$, then with very high probability, one of these two values will exceed $\sqrt{n}$, which tells us which graph is the planted one.
Note that gap between the ω for which the Planted problem can be solved, vs. the ω for which the distinguishing problem can be solved.

References

[ABS15] Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. *J. ACM*, 62(5), nov 2015. doi:10.1145/2775105.

[AGZ09] Greg W. Anderson, Alice Guionnet, and Ofer Zeitouni. *An Introduction to Random Matrices*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2009. doi:10.1017/CBO9780511801334.

[AM85] N Alon and V.D Milman. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38(1):73–88, 1985. URL: https://www.sciencedirect.com/science/article/pii/0095895685900929, doi:10.1016/0095-8956(85)90092-9.

[AMMN06] Noga Alon, Konstantin Makarychev, Yury Makarychev, and Assaf Naor. Quadratic forms on graphs. *Inventiones Mathematicae*, 163(3):499–522, March 2006. doi:10.1007/s00222-005-0465-9.

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. URL: https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf.

[CKK+05] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. In *20th Annual IEEE Conference on Computational Complexity (CCC'05)*, pages 144–153, 2005. doi:10.1109/CCC.2005.20.

[CMM06] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Near-optimal algorithms for unique games. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '06, pages 205–214, New York, NY, USA, 2006. Association for Computing Machinery. doi:10.1145/1132516.1132547.

[DKK+18] Irit Dinur, Subhash Khot, Guy Kindler, Dor Minzer, and Muli Safra. Towards a proof of the 2-to-1 games conjecture? In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018, pages 376–389, New York, NY, USA, 2018. Association for Computing Machinery. doi:10.1145/3188745.3188804.

[GLS88] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.

[Gro23] Grothendieck inequality. Grothendieck inequality, 2023. URL: https://en.wikipedia.org/wiki/Grothendieck_inequality.

[GW93] M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6), 1993.

[Hås01] Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001. URL: http://portal.acm.org/citation.cfm?doid=502090.502098, doi:http://doi.acm.org/10.1145/502090.502098.

[IKW01] R. Impagliazzo, V. Kabanets, and A. Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. In *Proceedings 16th Annual IEEE Conference on Computational Complexity*, pages 2–12, 2001. doi:10.1109/CCC.2001.933865.

[Kho02] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the 17th Annual IEEE Conference on Computational Complexity, Montréal, Québec, Canada, May 21-24, 2002*, page 25. IEEE Computer Society, 2002. doi:10.1109/CCC.2002.1004334.

[KKMO04] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for max-cut and other 2-variable csps? In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 146–154, 2004. doi:10.1109/FOCS.2004.49.

[KR03] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-\varepsilon. In *18th Annual IEEE Conference on Computational Complexity (Complexity 2003), 7-10 July 2003, Aarhus, Denmark*, page 379. IEEE Computer Society, 2003. doi:10.1109/CCC.2003.1214437.

[Max23] Maximum Cut. Maximum cut, 2023. URL: https://en.wikipedia.org/wiki/Maximum_cut.

[Rag08] Prasad Raghavendra. Optimal algorithms and inapproximability results for every csp? In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 245–254, New York, NY, USA, 2008. Association for Computing Machinery. doi:10.1145/1374376.1374414.

[Sch13] Claus Scheiderer. Sums of squares of polynomials with rational coefficients, 2013. arXiv:1209.2976.

[Tao11] Terence Tao. Topics in random matrix theory, 2011. Accessed on June 20, 2023. URL: https://terrytao.wordpress.com/category/teaching/254a-random-matrices/.

[TV08] Terence Tao and Van Vu. Random matrices: The circular law, 2008. arXiv:0708.2895.