Algorithmic Thresholds for Refuting Random Polynomial Systems

Jun-Ting (Tim) Hsieh Pravesh K. Kothari

March 24, 2022

CMU

Random polynomial systems

- **Input:** m polynomial equations $p_i(x) = b_i$ in n variables:

$$p_1(x) = b_1,$$

$$p_2(x) = b_2,$$

$$\dots$$

$$p_m(x) = b_m.$$

  - Each p_i is a homogeneous polynomial of degree k with i.i.d. Gaussian coefficients,
  - Each b_i is i.i.d. Gaussian.

Solutions?

- Is there a solution?
  - YES: can we find one?
  - NO: can we prove that it's unsatisfiable?
- Analogy: 3SAT formula $(x_1 \vee \neg x_3 \vee x_5) \wedge \cdots \wedge (x_2 \vee x_3 \vee x_5)$
  - Satisfiable: find a satisfying solution
  - Unsatisfiable: find a refutation

Solutions?

- Is there a solution?
  - YES: can we find one?
  - NO: can we prove that it's unsatisfiable?
- Analogy: 3SAT formula $(x_1 \vee \neg x_3 \vee x_5) \wedge \cdots \wedge (x_2 \vee x_3 \vee x_5)$
  - Satisfiable: find a satisfying solution -> *verifiable* proof of satisfiability (NP).
  - Unsatisfiable: find a refutation -> *verifiable* proof of unsatisfiability (coNP).

Refutation algorithm

- What are refutation algorithms?
- Given a system of equations, an algorithm that either outputs a "proof of unsatisfiability" or returns "don't know".
- We want an efficient refutation algorithm that
  - Takes the system $\{p_i(x) = b_i\}_{i \leq m}$ as input.
  - With probability $1 - o(1)$ over the randomness of the input equations, outputs a refutation.

Motivation

- **Algebraic geometry:** solution geometry of polynomial systems [Beltran-Shub'08, Burgisser-Cucker'11, Lairez'17].
- **Combinatorial optimization:** random 3SAT/3XOR/CSP refutation [Feige'02, Coja-Oghlan-Goerdt-Lanka'07, Allen-O'Donnell-Witmer15, Kothari-Mori-O'Donnell-Witmer'17].
- **Statistical Learning:** *matrix sensing* problem with "random Gaussian measurements" [Barak-Moitra'16, Potechin-Steurer'17, d'Orsi-Kothari-Novikov-Steurer'20].
- **Cryptography:** candidate PRGs based on hardness of solving random polynomial systems [Lombardi-Vaikuntanathan'17, Barak-Hopkins-Jain-Kothari-Sahai'19].

Why study refutations?

- If you can't find a solution, it would be nice to give a proof of unsatisfiability.
  - SAT solvers: output "sat" x, or "unsat" proof.
- Hardness of refutation ⇒ hardness of learning [Daniely-Linial-Shalev-Shwartz'14].
- Hardness of random "noisy" 3XOR ⇒ public-key encryption [Applebaum-Barak-Wigderson'09].

Random polynomial systems

- At what m is the input system $\{p_i(x) = b_i\}_{i \leq m}$ unsatisfiable?
- $m = n$: information-theoretic threshold.
  - Bezout's theorem: the number of common zeros of n "generic" degree-k polynomials is at most k^n.
  - The probability that the $(n+1)$-th polynomial has a common zero is 0.
- **Question:** What's the smallest m at which efficient algorithms can find refutations (certify that it's unsatisfiable)?
  - Algorithmic threshold.

Algorithmic threshold of refutation

[Raghavendra-Rao-Schramm’17]
For $\delta \in (0,1)$, $\exp(n^{\delta})$ time for $\tilde{O}(n) \cdot n^{(1-\delta)/2}$ clauses

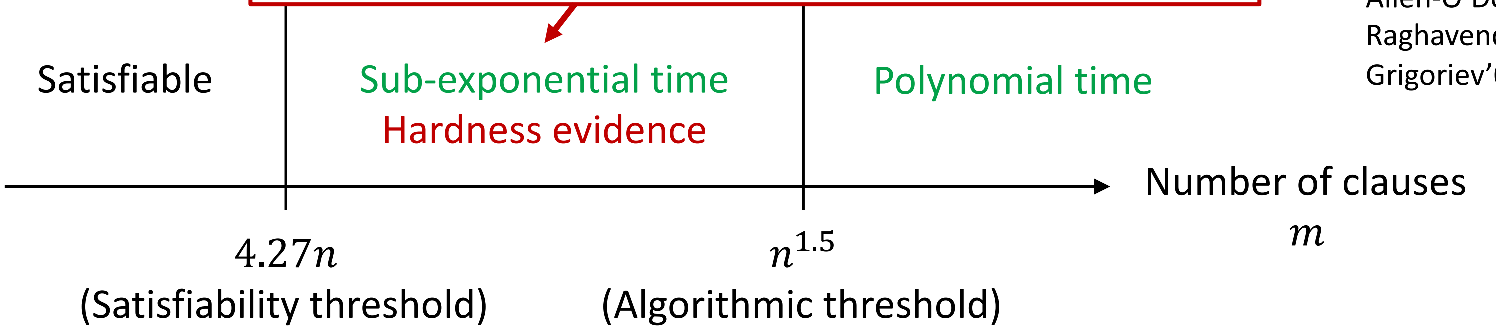
[Feige’02, Feige-Kim-Ofek’06, Friedman-Goerdt-Krivelevich’05, Allen-O’Donnell-Witmer’15, Raghavendra-Rao-Schramm’17, Grigoriev’01, Schoenebeck’08]

Random 3SAT

$(x_1 \vee \neg x_3 \vee x_6)$

$(x_2 \vee x_3 \vee \neg x_5)$

$\vdots$



Algorithmic threshold of refutation

[Raghavendra-Rao-Schramm'17]
For $\delta \in (0,1)$, $\exp(n^{\delta})$ time for $\tilde{O}(n) \cdot n^{(1-\delta)/2}$ clauses

[Feige'02, Feige-Kim-Ofek'06, Friedman-Goerdt-Krivelevich'05, Allen-O'Donnell-Witmer'15, Raghavendra-Rao-Schramm'17, Grigoriev'01, Schoenebeck'08]

Random 3SAT

$(x_1 \vee \neg x_3 \vee x_6)$
$(x_2 \vee x_3 \vee \neg x_5)$
$\vdots$

Satisfiable

Sub-exponential time
Hardness evidence

Polynomial time

Number of clauses m

$4.27n$
(Satisfiability threshold)

$n^{1.5}$
(Algorithmic threshold)

Our problem

$p_1(x) = b_1$
$p_2(x) = b_2$
$\vdots$

Satisfiable

Sub-exponential time
Hardness evidence

Polynomial time

Number of polynomials m

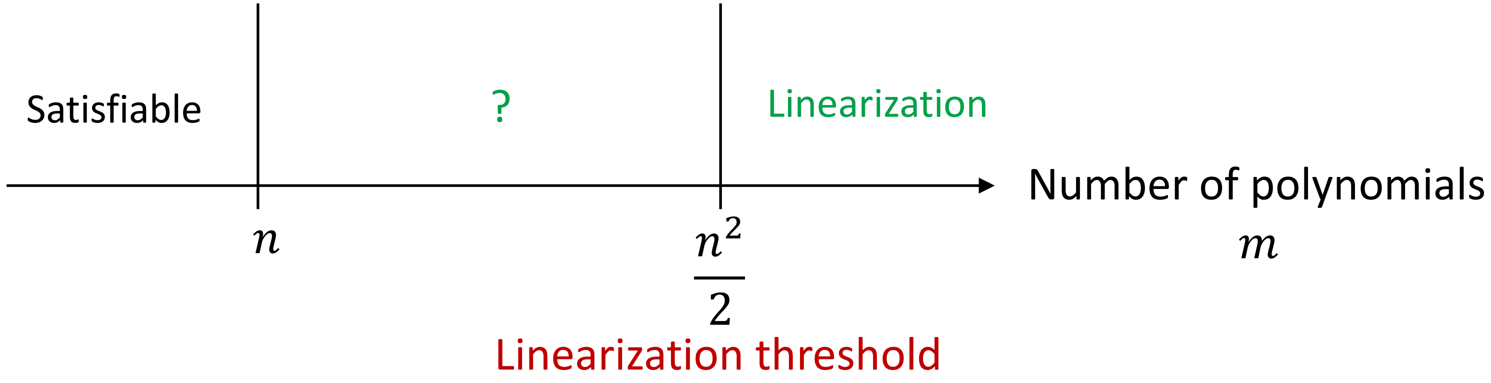
n
(Info-theoretic threshold)

?
(Algorithmic threshold)

Linearization threshold

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

- Let's restrict to the case when p_i's are degree 2 for simplicity.
- When $m \geq \frac{n(n+1)}{2}$, easy.
  - Linearization trick: $x_i x_j \to y_{ij}$. Done via Gaussian elimination.



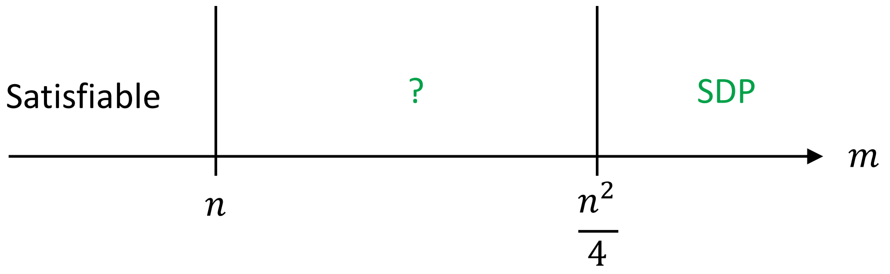
Semidefinite relaxation

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

- Write each $p_i(x) = x^\top G_i x = \langle G_i, xx^\top \rangle$.
- SDP relaxation: replace $xx^\top$ with X:

$$\langle G_i, X \rangle = b_i, \qquad \forall i \in [m],$$
$$X \succcurlyeq 0.$$

  - Infeasible $\Rightarrow$ proof of unsatisfiability.
  - Feasible $\Rightarrow$ don't know.
- Our first result:
  - If $m \ge \frac{n^2}{4} + \tilde{O}(n)$, then the SDP is infeasible w.h.p.
  - If $m \le \frac{n^2}{4} - \tilde{O}(n)$, then the SDP is feasible w.h.p.

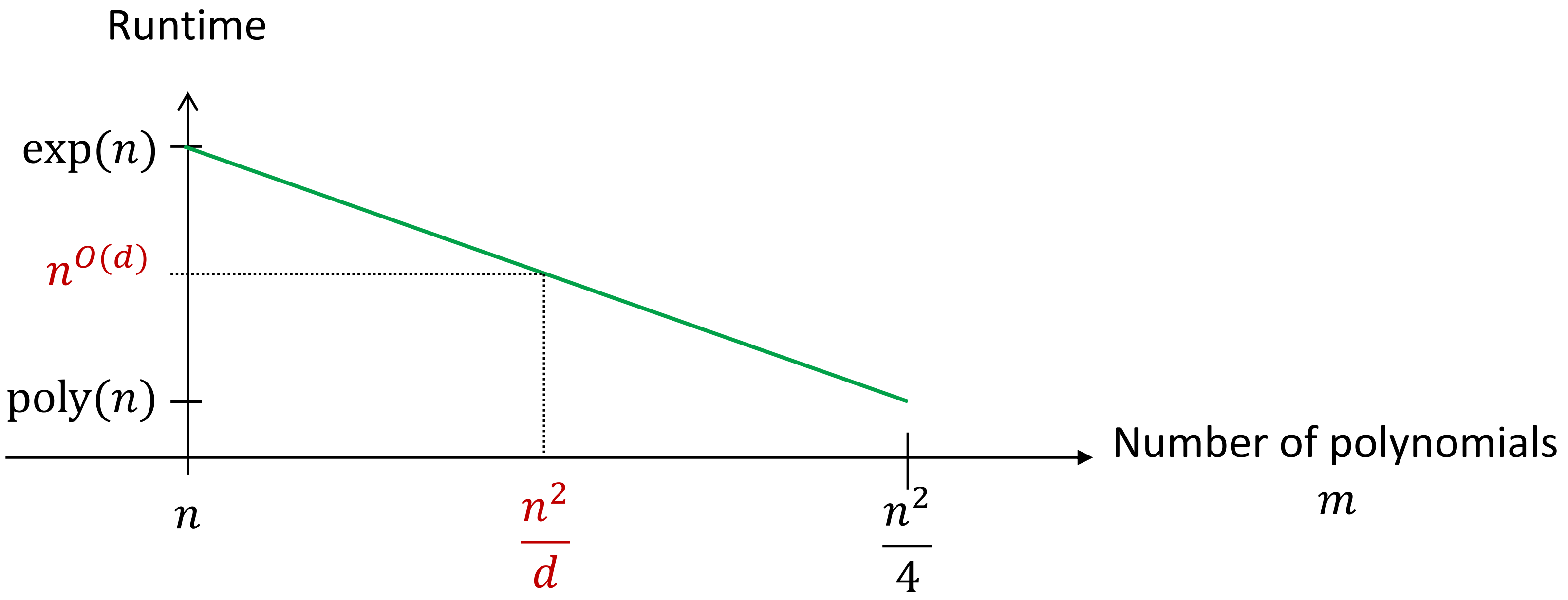


Our results: upper bound

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

Main result: upper bound

- For any $d \in \mathbb{N}$, we give an $n^{O(d)}$-time refutation algorithm that succeeds when $m \ge O\left(\frac{n^2}{d}\right)$.

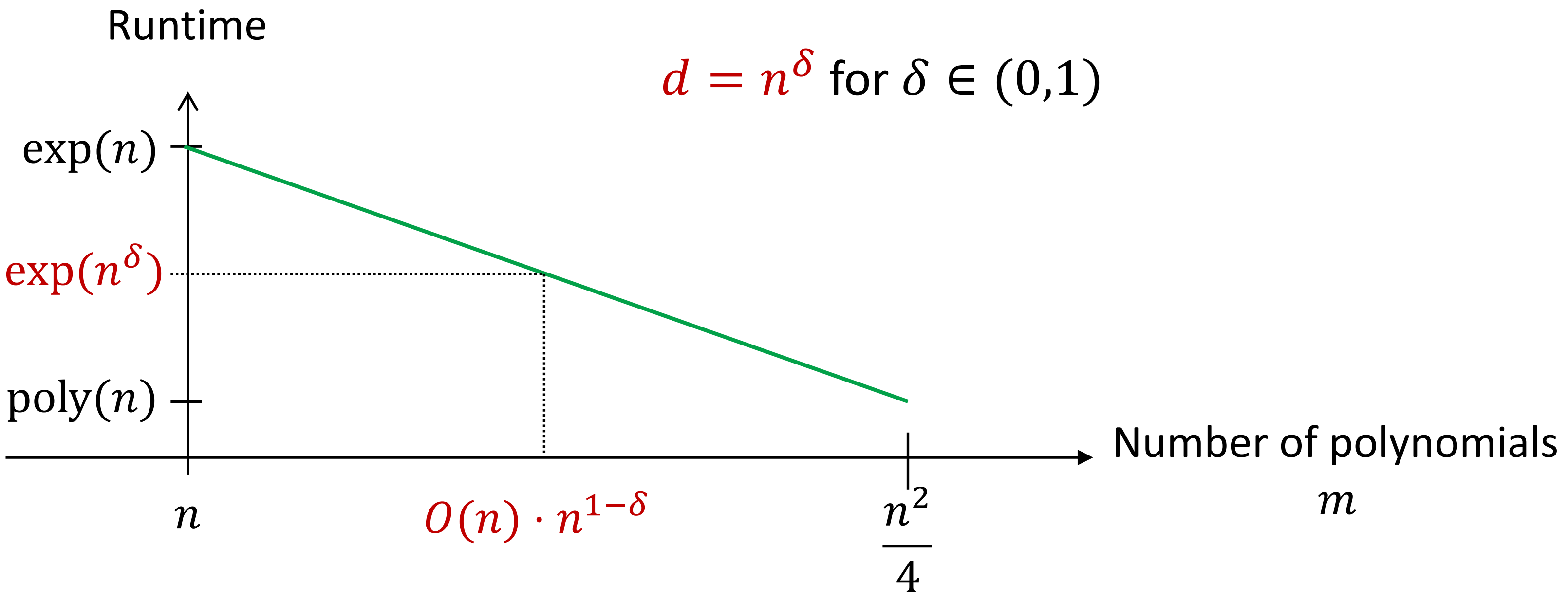


Our results: upper bound

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

Main result: upper bound

- For any $d \in \mathbb{N}$, we give an $n^{O(d)}$-time refutation algorithm that succeeds when $m \geq O\left(\frac{n^2}{d}\right)$.



Our results: lower bound

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

- Is this optimal?
  - No NP-hardness known (not even for random 3SAT).

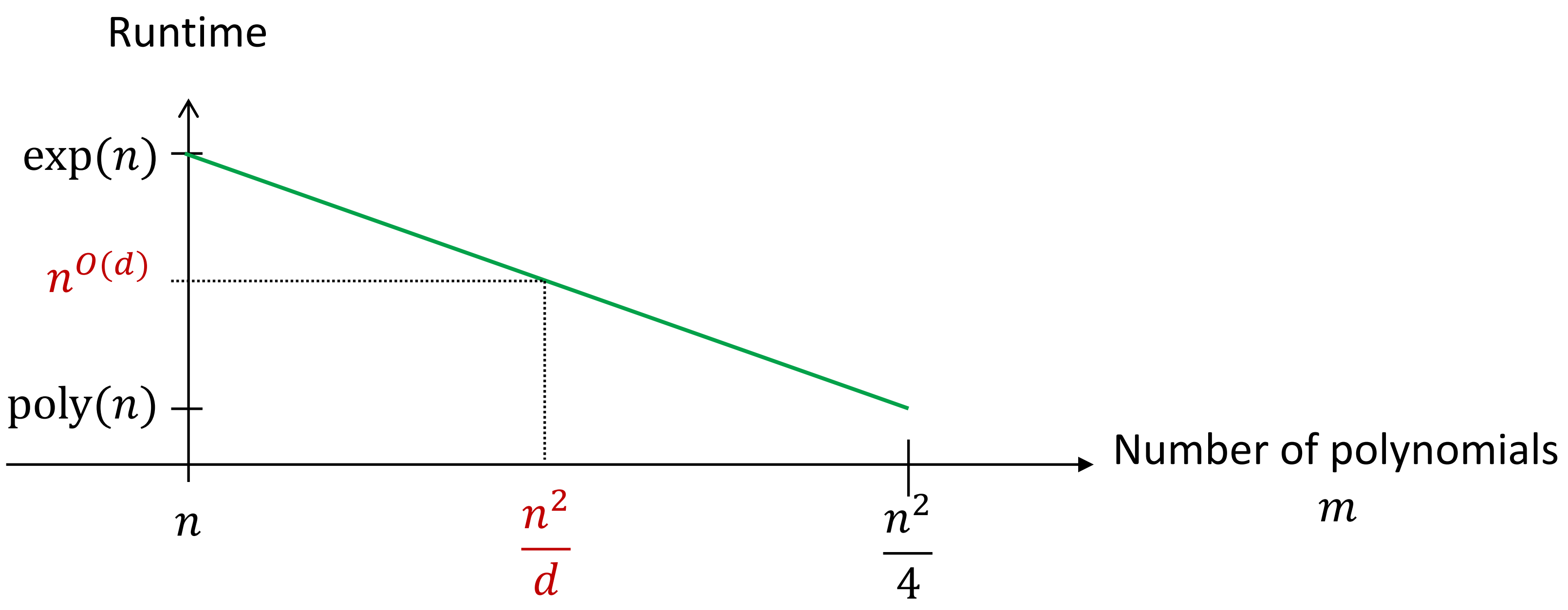
Main result: lower bound

- A subclass of $n^{O(d)}$-time algorithms fail when $m \le O\left(\frac{n^2}{d}\right)$.
  - Fail at *distinguishing*
    - Null model: the random polynomial system,
    - Planted model: a certain distribution over feasible polynomial systems.

Our results: summary

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

- Strongly suggests an algorithmic threshold of $O\left(\frac{n^2}{d}\right)$ for $n^{O(d)}$ time algorithms.

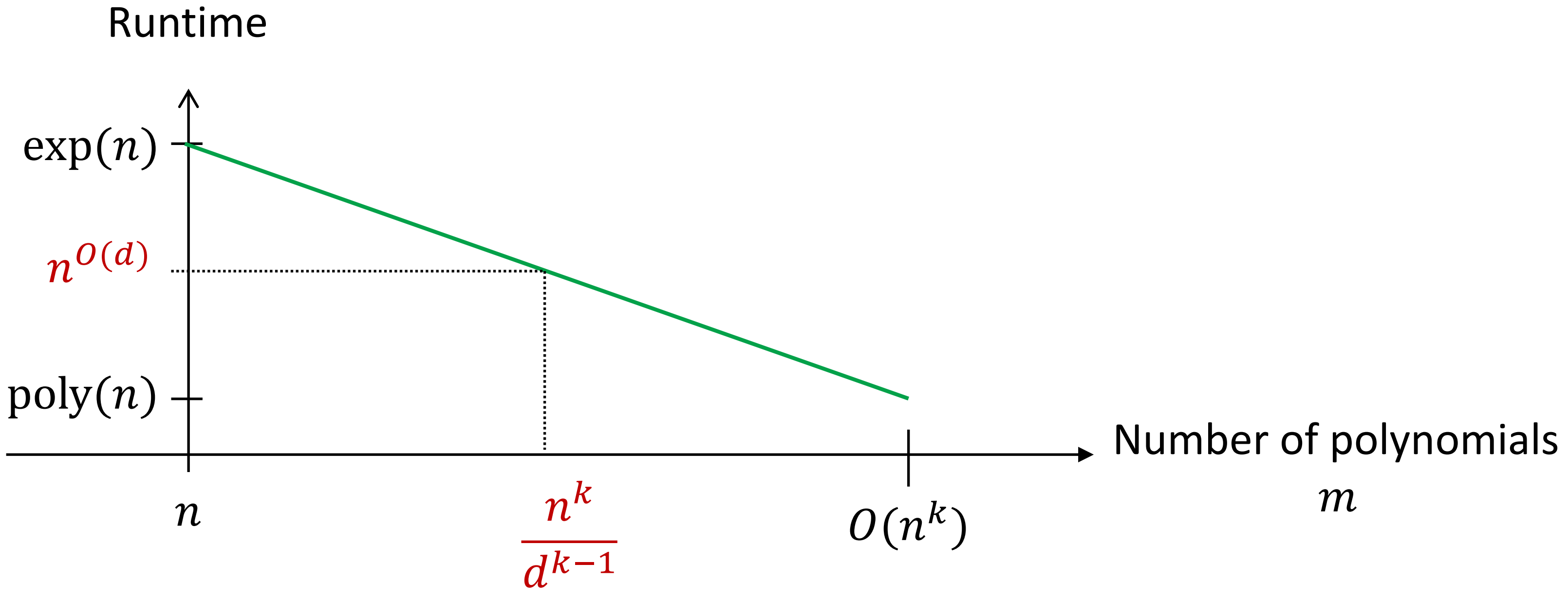


Information-computation gap!

Our results: summary

Refutation problem:
Certify that $\{p_i(x) = b_i\}_{i \le m}$ is infeasible.

- More generally, for degree-k polynomials, the algorithmic threshold is $O\left(\frac{n^k}{d^{k-1}}\right)$ for $n^{O(d)}$ time algorithms.



Information-computation gap!

Background

Background: proof systems

- Most problems in computer science can be represented as a system of polynomial constraints.
  - 3SAT:
    - $x_i \in \{0,1\} \;\rightarrow\; x_i^2 - x_i = 0.$
    - Clause $\left(x_i \vee \neg x_j \vee x_k\right) \;\rightarrow\; (1 - x_i)(x_2)(1 - x_3) = 0.$
  - Max-Cut:
    - $x_i \in \{\pm 1\} \;\rightarrow\; x_i^2 = 1.$
    - For each edge $(i, j) \;\rightarrow\; x_i x_j = -1.$

Background: proof systems

- Toy example: consider the following system

$$x_1^2 - x_1 = 0$$
$$x_2^2 - x_2 = 0$$
$$x_1 + x_2 - 2 = 0$$
$$x_1 x_2 = 0$$

- This is infeasible. How do we prove it? Linear combination of the above:

$$\frac{1}{2} \cdot (x_1^2 - 1) + \frac{1}{2} \cdot (x_2^2 - 1) + \frac{1}{2}(-x_1 - x_2 - 2) \cdot (x_1 + x_2 - 2) + 1 \cdot x_1 x_2 = 1.$$

- We have *derived* $1 = 0$! This is a contradiction!

Background: Nullstellensatz

- Given a system of constraints $\{f_1(x) = \cdots = f_m(x) = 0\}$, we can derive many more high-degree equalities, e.g.

$$f_1(x)x_1x_2 = f_1(x) + f_2(x) = 0$$

$$g(x) \coloneqq \sum_{i=1}^{m} a_i(x)f_i(x) = 0$$

- If $\max_i \deg(a_i f_i) \leq d$, then this derivation can be captured by the degree-d Nullstellensatz proof system.

Background: Nullstellensatz

Given a system of constraints $\{f_1(x) = \cdots = f_m(x) = 0\}$

- Suppose there exist $a_1, \ldots, a_m$ such that $\sum_{i=1}^{m} a_i f_i = 1$, then we get $1 = 0$. If $\max_i \deg(a_i f_i) \leq d$, this is called a degree-d Nullstellensatz refutation.
- **Automatizable:** such a refutation can be computed in $n^{O(d)}$ time.
  - By simply solving a system of linear equations.

Background: Nullstellensatz

- How powerful is Nullstellensatz? Can it capture everything?
- **Completeness**: (weak) Hilbert's Nullstellensatz.
  - For an algebraically closed field $\mathbb{F}$, given a system $\{f_1(x) = \cdots = f_m(x) = 0\}$, if it's unsatisfiable, you can always find $a_1, \dots, a_m \in \mathbb{F}[x]$ such that

$$a_1 f_1 + \cdots + a_m f_m = 1.$$

  - But, we have no guarantees on the degrees of a_i.
    - The runtime $n^{O(d)}$ can be very large [Buss-Pitassi'98, Razborov'98].
    - Intuition: as m increases, d decreases.

Semidefinite Relaxation

Random polynomial systems

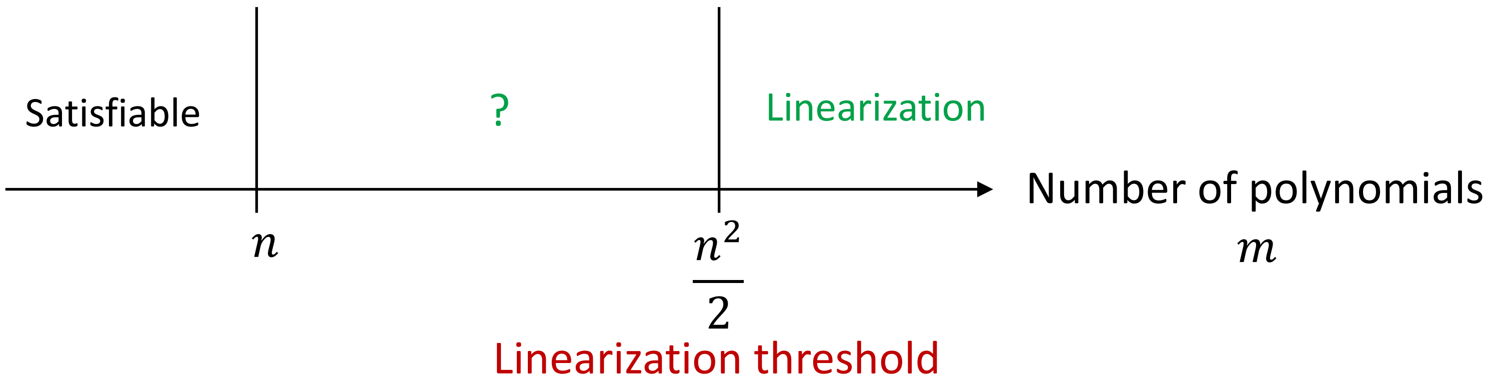
- **Task**: find a refutation for

$$p_1(x) = b_1,$$

$$\dots$$

$$p_m(x) = b_m.$$

  - Coefficients of each p_i and b_i are chosen i.i.d. Gaussian.
  - For simplicity, we assume each p_i is quadratic.



SDP relaxation

- Write each $p_i(x) = x^\top G_i x = \langle G_i, xx^\top \rangle$.
- SDP relaxation: replace $xx^\top$ with X:

$$\langle G_i, X \rangle = b_i, \qquad \forall i \in [m],$$
$$X \succcurlyeq 0.$$

  - Infeasible ⇒ proof of unsatisfiability.
  - Feasible ⇒ don't know.

SDP relaxation

- Write each $p_i(x) = x^\top G_i x = \langle G_i, xx^\top \rangle$.
- SDP relaxation: replace $xx^\top$ with X:

$$\langle G_i, X \rangle = b_i, \qquad \forall i \in [m],$$

$$X \succcurlyeq 0.$$

  - Trivial (solutions $= \{0\}$) $\Rightarrow$ proof of unsatisfiability.
  - Non-trivial (solutions $\neq \{0\}$) $\Rightarrow$ don’t know.
- For simplicity, assume $b_i = 0$.

SDP relaxation

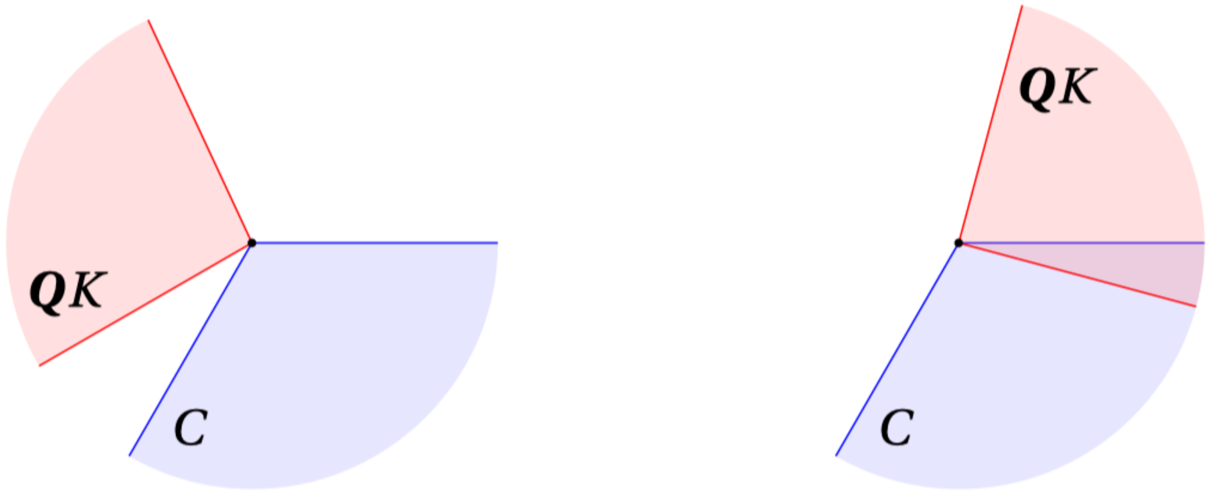
Semidefinite program:
- $\langle G_i, X \rangle = 0, \forall i \in [m]$,
- $X \succcurlyeq 0$.

- We view X (symmetric matrix) as a vector in a space of dimension $\frac{n(n+1)}{2}$.
  - $\langle G_i, X \rangle = 0\ \forall i \in [m]$ define a random linear subspace L of dimension $\frac{n(n+1)}{2} - m$.
  - Let C be the cone of PSD matrices.
  - $C \cap L$ is the set of solutions.
- **Question:** is $C \cap L$ non-trivial?
  - Trivial $\Rightarrow$ SDP has no non-trivial solutions $\Rightarrow$ SDP succeeds at refuting.
  - Non-trivial $\Rightarrow$ SDP has non-trivial solutions $\Rightarrow$ SDP fails at refuting.

SDP relaxation

Semidefinite program:
- $\langle G_i, X\rangle = 0, \forall i \in [m]$,
- $X \succcurlyeq 0$.

- The constraints $\langle G_i, X\rangle = 0\ \forall i \in [m]$ define a random subspace: write as QL where Q is random rotation and L is any fixed subspace.
- What is $\Pr_Q[C \cap QL \neq \{0\}]$?
  - Conic geometry!



SDP relaxation

Semidefinite program:
- $\langle G_i, X\rangle = 0, \forall i \in [m]$,
- $X \succcurlyeq 0$.

- Answer depends on the statistical dimension δ (generalization of dimension).
  - Subspace L: $\delta(L) = \dim(L) = \frac{1}{2}n(n+1) - m$.
  - PSD cone C: $\delta(C) = \frac{1}{4}n(n+1)$.
- **Lemma** [Amelunxen-Lotz-McCoy-Tropp'14]. Let C, K be convex cones in $\mathbb{R}^N$ and let $Q \in \mathbb{R}^{N\times N}$ be a random rotation matrix. Then,

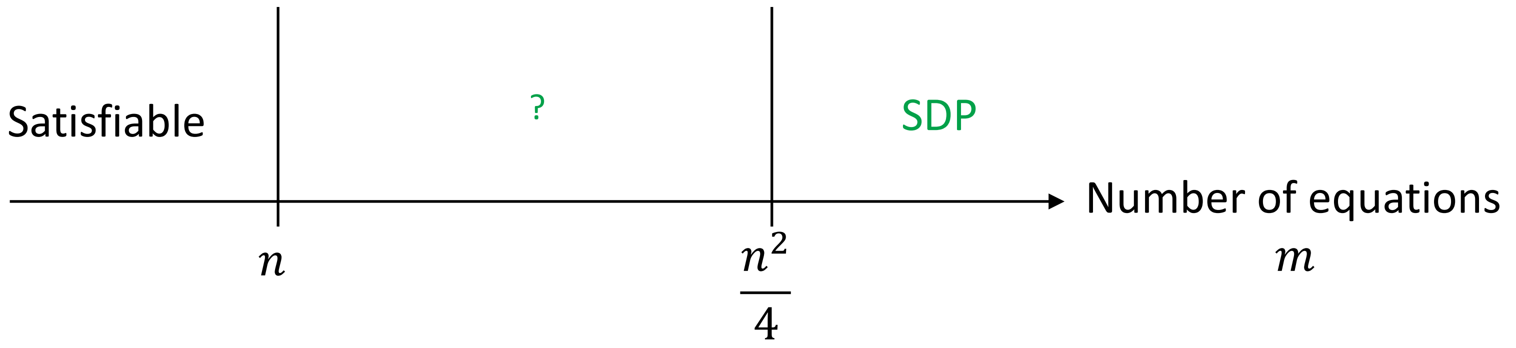
$$\delta(C) + \delta(K) \le N - O\left(\sqrt{N}\log(1/\eta)\right) \Longrightarrow \Pr_Q[C \cap QK \ne \{0\}] \le \eta$$

$$\delta(C) + \delta(K) \ge N + O\left(\sqrt{N}\log(1/\eta)\right) \Longrightarrow \Pr_Q[C \cap QK \ne \{0\}] \ge 1 - \eta$$

SDP relaxation

Semidefinite program:
- $\mathrm{tr}(G_i X) = 0, \forall i \in [m]$,
- $X \succcurlyeq 0$.

- Thus,
  - If $m \geq \frac{n^2}{4} + Cn\log n$, then the SDP is trivial whp $\Rightarrow$ SDP succeeds at refuting.
  - If $m \leq \frac{n^2}{4} - Cn\log n$, then the SDP is non-trivial whp $\Rightarrow$ SDP fails at refuting.
- $\frac{n^2}{4}$ is the threshold for our SDP relaxation.



Upper Bound

Upper bound

- Assume the p_i's are quadratic.
- **Goal:** given system $\{p_i(x) = b_i\}_{i \le m}$, find a degree-$d$ Nullstellensatz refutation:
  - Find polynomials $a_1(x), \ldots, a_m(x)$ of degree $d - 2$ such that

$$\sum_{i=1}^{m} a_i(x)(p_i(x) - b_i) = 1.$$

  - This gives us $1 = 0$!
- Show: whp these polynomials $a_1, \ldots, a_m$ exist!

Upper bound

- We show that the generated ideal at degree d is complete:
  - For every homogeneous polynomial f of degree d, there are polynomials $a_1, \ldots, a_m$ of degree $d-2$ such that $\sum_{i \le m} a_i(x)(p_i(x) - b_i) = f(x)$.
- We have a refutation since
  - We can derive $p_1(x)^{d/2} = b_1^{d/2}$, hence $b_1^{-d/2}\, p_1(x)^{d/2} = 1$.
  - $p_1^{d/2}$ is a homogeneous degree d polynomial, so there exist $a_1, \ldots, a_m$ such that $1 = \sum_{i \le m} a_i(p_i - b_i) = 0$. We have derived $1 = 0$!

Upper bound

Completeness of generated ideal at degree d: For every homogeneous polynomial f of degree d, there are polynomials $a_1, \dots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- Let's gain some intuition. Suppose $p_i(x) = x_{j_1} x_{j_2}$ for $j_1, j_2 \in [n]$ and $b_i = 0$.
- $d = 2$: since a_i's must be constants, we need all monomials:

$$x_1^2, \dots, x_n^2, x_1 x_2, \dots, x_{n-1} x_n$$

- $d = 3$: since a_i's are degree-1, we need fewer polynomials! We can delete $x_1 x_2$:
  - Polynomials like $x_1 x_2 x_3$ is still captured by $x_2 x_3$: $x_1 x_2 x_3 = x_1 \cdot x_2 x_3$.
- $d = 4$: we can even delete $x_1 x_3, x_2 x_3$ because $x_1 x_2 x_3 x_4 = x_1 x_2 \cdot x_3 x_4$.
- As d increases, we need fewer polynomials.

Upper bound

Completeness of generated ideal at degree d:
For every homogeneous polynomial f of degree d, there are polynomials $a_1, \dots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- Consider a graph where $(j_1, j_2) \in E$ if $x_{j_1} x_{j_2}$ is in our set of polynomials.
  - d must be larger than the size of the largest independent set!
  - Suppose we randomly choose $m \approx \frac{n^2}{4}$ monomials, we get $G(n, ½)$.
    - But, largest independent set is $\Theta(\log n)$.
- Fortunately, our polynomials are dense (every monomial appears in every polynomial).

Upper bound

Completeness of generated ideal at degree d:
For every homogeneous polynomial f of degree d, there are polynomials $a_1, \ldots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- Let $f = \sum_{i \le m} a_i(p_i - b_i)$ where $a_1, \ldots, a_m$ are homogeneous polynomials of degree $d - 2$.
- Write out the monomials

$$f(x) = \sum_{|\alpha|=d,d-2} \hat{f}(\alpha) x^\alpha = \sum_{i=1}^{m} \sum_{|\beta|=d-2} \hat{a}_i(\beta) \left(\sum_{|\gamma|=2} \hat{p}_i(\gamma) x^{\beta+\gamma} - b_i x^\beta \right)$$

- Comparing coefficients, we get a linear system

$$\hat{f} = M \cdot \hat{a}$$

- Need: for all $\hat{f}$, there exists $\hat{a}$ such that $\hat{f} = M \cdot \hat{a}$.

Techniques: full rank

Completeness of generated ideal at degree d: For every homogeneous polynomial f of degree d, there are polynomials $a_1, \ldots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- It suffices to show that the matrix M is full row-rank (columns span everything).
  - In many papers on average-case complexity, the problems reduce to proving a certain structured random matrix with highly correlated entries is full rank or PSD.
  - Very non-trivial (sometimes takes 30+ pages)!
- Luckily, we can exploit the structure of our matrix.

Techniques: full rank

Completeness of generated ideal at degree d:
For every homogeneous polynomial f of degree d, there are polynomials $a_1, \ldots, a_m$ such that $\sum_{i \leq m} a_i(p_i - b_i) = f$.

- Decompose the matrix! Find submatrices of M:
  - Each submatrix is full rank,
  - They cover all the rows of M (rows may overlap),
  - They have disjoint columns of M,
  - The diagonal entries of each submatrix are independent of (1) the off-diagonal entries, and (2) the other submatrices.
- How do we “stitch” the submatrices together to prove that M is full row-rank?

Techniques: full rank

Completeness of generated ideal at degree d: For every homogeneous polynomial f of degree d, there are polynomials $a_1, \dots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- **Lemma.** Let $M = \begin{bmatrix} A & C_2 \\ C_1 & B \end{bmatrix}$ such that A is full rank, $B = B' + g \cdot I$, where $g \sim N(0,1)$ independent of the rest. Then M is full rank.
  - This is how we stitch two submatrices. We apply this lemma repeatedly.
  - **Proof**. M is full rank if and only if the Schur complement is full rank:

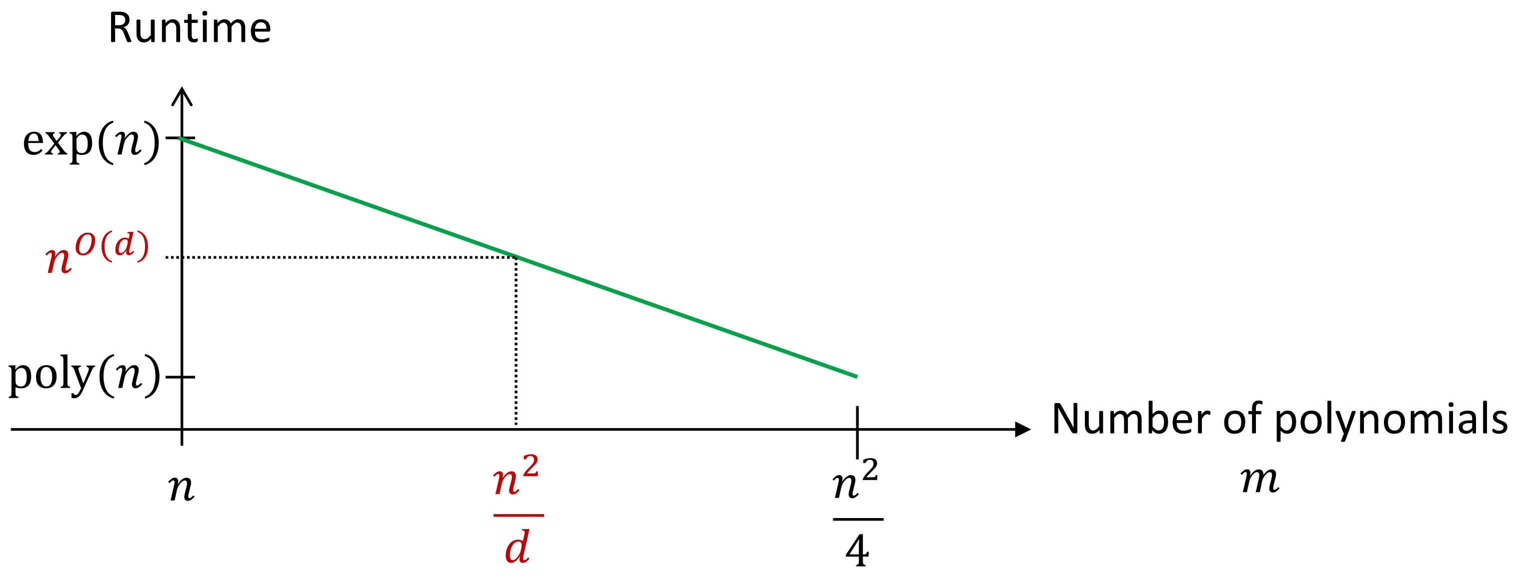
$$B - C_1 A^{-1} C_2 = g \cdot I - (C_1 A^{-1} C_2 - B')$$

  - Suppose not, then g must be an eigenvalue of $C_1 A^{-1} C_2 - B'$. But due to independence of g, this occurs with probability 0.
- **Remark:** our upper bound holds for any “nice” distribution.

Upper bound

> Completeness of generated ideal at degree d:
> For every homogeneous polynomial f of degree d, there are polynomials $a_1, \dots, a_m$ such that $\sum_{i \le m} a_i(p_i - b_i) = f$.

- **Recap:** we prove that degree-d Nullstellensatz succeeds when $m \ge O\left(\frac{n^2}{d}\right)$:
  - The generated ideal at degree d is complete → The matrix M is full row rank.
- **Remark:** It is crucial that our polynomials $p_1, \dots, p_m$ are dense.
  - For sparse polynomials, we need d to be very large for the completeness to hold.



Lower Bound

Lower bound

- No NP-hardness.
- No reduction from a “hard” average-case problem.
- People use certain subclasses of $n^{O(d)}$-time algorithms as proxy for all $n^{O(d)}$-time algorithms.
  - Not very strong, but so far we don’t know better lower bounds.

Background: low-degree polynomial method

- Consider the hypothesis testing problem of distinguishing between a null distribution ν_N and planted distribution ν_P over $\mathbb{R}^K$.
  - Example: $\nu_N = G(n, 1/2)$ and $\nu_P = G(n, 1/2) +$ a clique of size $\sqrt{n}$.
  - Any "testing" algorithm can be seen as a function $T_K : \mathbb{R}^K \to \mathbb{R}$ on the input z which outputs "planted" if $T_K(z)$ exceeds some threshold τ.
  - The classical Neyman-Pearson lemma shows that the "optimal" test is the likelihood ratio test L.
    - Hard to compute…

Background: low-degree polynomial method

- Restrict our test functions to low-degree polynomials of the input!
- Consider the following optimization problem:

$$\max_{f} \mathbb{E}_{\nu_P}[f]$$

$$\text{such that } \mathbb{E}_{\nu_N}[f^2] = 1 \text{ and } f \text{ is a degree } d \text{ polynomial}$$

- **Proposition** [Kunisky-Wein-Bandeira'19]**.** The optimizer is the normalized truncated likelihood ratio $L^{\leq d}$, and the optimal value is $\left\|L^{\leq d}\right\| = \mathbb{E}_{\nu_N}\left[\left(L^{\leq d}\right)^2\right]^{1/2}$.
  - Suffices to show: $\mathrm{Var}_{\nu_N}[L^{\leq d}] = \sum_{1\leq|\alpha|\leq d} \mathbb{E}_{\nu_P}[\chi_\alpha]^2 \leq 1$. We use the Hermite basis.
- Usually, the main step is to construct a *hard-to-distinguish* ν_P.

Background: low-degree polynomial method

- Too restricted?
- $O(\log n)$-degree polynomial:
  - They capture the strongest known algorithms for many canonical problems.
  - [Brennan-Bresler-Hopkins-Li-Schramm'20]: under appropriate assumptions, the $O(\log n)$-degree polynomials are as powerful as the statistical query model.
  - **Connection to SoS.** [Hopkins-Kothari-Potechin-Raghavendra-Schramm'17] conjectured that indistinguishability by degree-d polynomials implies lower bounds for $\tilde{O}(d)$-degree SoS.
  - Used in many works on average-case algorithmic thresholds [Hopkins'18, Gamarnik-Jagannath-Wein'20, Schramm-Wein'20].

Techniques: lower bound

- Planted distribution ν_P:
  - Fix a small parameter $c = o\left(\frac{1}{d\sqrt{m}}\right)$.
  - Sample z uniformly from $\left\{\pm\frac{1}{\sqrt{n}}\right\}^n$.
  - For each i, sample $b_i \sim N(0,1)$ independently.
  - For each i, sample $G_i \in \mathbb{R}^{n\times n}$ w/ Gaussian entries conditioned on $\langle G_i, zz^\top\rangle = c \cdot b_i$. Set $p_i(x) = x^\top G_i x$.
    - Sample g conditioned on $\langle g, v\rangle = b$ where $\|v\|_2 = 1$: sample $h \sim N(0, I)$, then set $g = bv + (I - vv^\top)h$.

Techniques: lower bound

Planted distribution ν_P: parameter $c = o(1/d\sqrt{m})$,

- $z \sim \{\pm 1/\sqrt{n}\}^n$, $b_i \sim N(0,1)$
- $G_i \sim N(0, I_{n\times n})$ conditioned on $z^\top G_i z = c \cdot b_i$. Set $p_i(x) = x^\top G_i x$.

- Remarks on the planted distribution:
  - $\{p_i(x) = b_i\}_{i \le m}$ always feasible with solution $x^* = z/\sqrt{c} \in \mathbb{R}^n$.
  - Different from the "natural" planted distribution:
    - Sample x^*, sample p_i with uniformly random coefficients, and set $b_i = p_i(x^*)$.
    - This is easy to distinguish! We can recover x^* when $m = \tilde{O}(n)$.
    - Instead, we choose b_i's independently, and choose coefficients of p_i's to be mildly correlated.
  - In ν_P, the norm of the planted solution is large: $\|x^*\|_2 = 1/\sqrt{c}$.
    - Necessary since there is an efficient distinguisher if $c \gg \sqrt{n/m}$.
    - Consider matrix $Q = \sum_{i=1}^m \operatorname{sgn}(b_i) \cdot G_i$. The spectral norm $\|Q\|$ is a distinguisher.

Techniques: lower bound

Planted distribution ν_P: parameter $c = o(1/d\sqrt{m})$,
- $z \sim \{\pm 1/\sqrt{n}\}^n$, $b_i \sim N(0,1)$
- $G_i \sim N(0, I_{n\times n})$ conditioned on $z^\top G_i z = c \cdot b_i$. Set $p_i(x) = x^\top G_i x$.

- We prove that

$$\mathrm{Var}[L^{\leq d}] = \sum_{\substack{\alpha,\beta: \\ 1\leq|\alpha|+|\beta|\leq d}} \mathbb{E}_{(G,b)\sim\nu_P}\left[h_\alpha(G) h_\beta(b)\right]^2 \leq 1.$$

- For simplicity, we will assume $b_i = 0$.
  - We're given input $G = (G_1, \ldots, G_m)$ where $z^\top G_i z = 0$ (z is our planted solution).
- Analyze $\mathbb{E}_{G\sim\nu_P}[h_\alpha(G)]$, where $\alpha \in \mathbb{N}^{m\times n\times n}$.

Techniques: lower bound

Planted distribution ν_P:

- $z \sim \{\pm 1/\sqrt{n}\}^n$,
- $G_i \sim N(0, I_{n\times n})$ conditioned on $z^\top G_i z = 0$. Set $p_i(x) = x^\top G_i x$.

- For $\alpha \in \mathbb{N}^{m\times n\times n}$, we view it as a labeled directed multigraph (with self-loops allowed) with n vertices and $|\alpha|$ edges with labels from $[m]$.

- Let $\alpha = (\alpha^1, \dots, \alpha^m)$ where each $\alpha^s \in \mathbb{N}^{n\times n}$.
  - For each $s \in [m]$, we view α^s as the adjacency matrix of the subgraph whose edges have label s.
  - Let $\Delta \in \mathbb{N}^n$ such that $\Delta_i = \sum_{s=1}^m \sum_{j=1}^n \alpha_{ij}^s + \alpha_{ji}^s$, the total degree of vertex i.

- **Lemma**. For $\alpha \in \mathbb{N}^{m\times n\times n}$,

$$\mathbb{E}_{G\sim\nu_P}[h_\alpha(G)] = n^{-|\alpha|}(-1)^{|\alpha|/2}\prod_{s=1}^{m}(|\alpha^s|-1)!!$$

if Δ_i is even $\forall i$ and $|\alpha^s|$ is even $\forall s$, and 0 otherwise.

Techniques: lower bound

Planted distribution ν_P:
- $z \sim \{\pm 1/\sqrt{n}\}^n$,
- $G_i \sim N(0, I_{n\times n})$ conditioned on $z^\top G_i z = 0$. Set $p_i(x) = x^\top G_i x$.

Fix $|\alpha| = e \leq d$.

- How many directed graphs with n vertices, e edges, even degrees?
  - Answer: $\leq (8n)^e$.
- How many ways can you assign labels in $[m]$ to the edges such that each label appears even number of times?
  - Answer: dominated by the case when each label appears twice: $\leq (2me)^{e/2}$.
- In total: when $m = O\left(\frac{n^2}{d}\right)$,

$$\sum_{e\geq 2,\text{ even}}^{d} n^{-2e} \cdot (8n)^e (2me)^{e/2} = \sum_{e\geq 2,\text{ even}}^{d} \left(\frac{e}{2d}\right)^{e/2} \leq 1$$

Techniques: lower bound

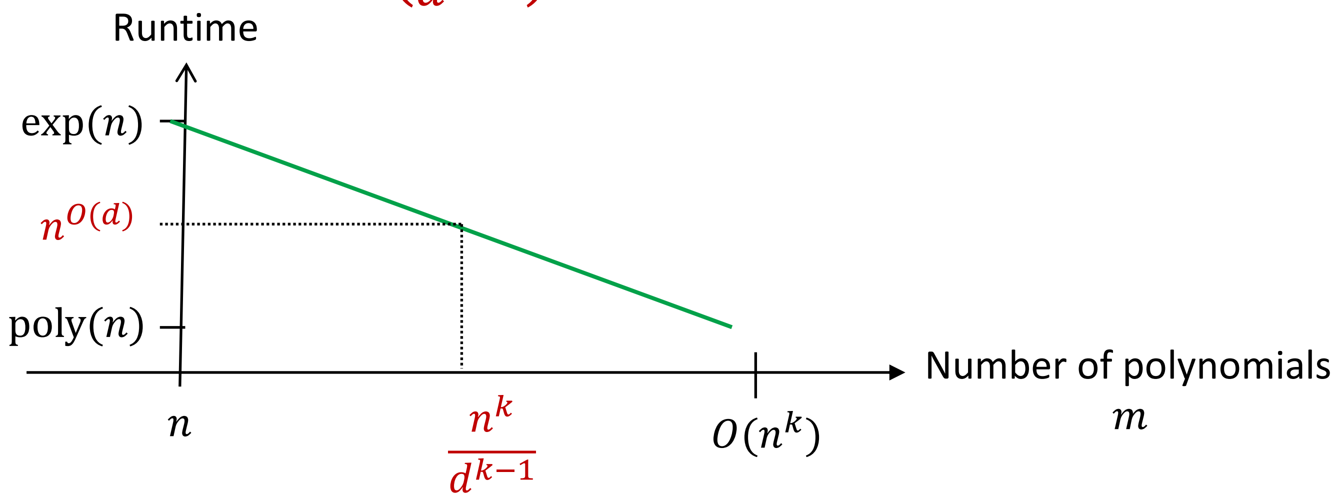
Planted distribution ν_P: parameter $c = o(1/d\sqrt{m})$,

- $z \sim \{\pm 1/\sqrt{n}\}^n,\ b_i \sim N(0,1)$
- $G_i \sim N(0, I_{n\times n})$ conditioned on $z^\top G_i z = c \cdot b_i$. Set $p_i(x) = x^\top G_i x$.

- **Recap:** we prove that degree-d polynomials fail when $m \le O\left(\frac{n^2}{d}\right)$:
  - We constructed a *hard* planted distribution ν_P.
  - $\mathrm{Var}[L^{\le d}] \lesssim 1$ shows that degree-d polynomials fail to distinguish between the null distribution ν_N and planted distribution ν_P.
  - Failure to distinguish ⇒ Failure to refute our random polynomial system.

Conclusion

- For random quadratic polynomial systems, $m = O\left(\frac{n^2}{d}\right)$ seems to be the algorithmic threshold (for $n^{O(d)}$ runtime).
  - Upper bound: degree-d Nullstellensatz.
  - Lower bound: degree-d low-degree hardness.
- For degree-k polynomials, $O\left(\frac{n^k}{d^{k-1}}\right)$ is the algorithmic threshold.



Thank you!