9. Polynomial Identity Testing

Testing if two polynomials $f(x), g(x)$ are equal is quite common in various problems. There are multiple ways of carrying this out

1. Compare the coefficients
2. Substitute x with some value and compare the resultant output.
3. Compare the roots of the polynomials
4. Subtract and check if the resultant is zero (Equivalent to the output being 0 irrespective of the value of x).

Fundamental Theorem

Any univariate polynomial of degree d has at most d roots. Therefore, we can check if a polynomial is 0 at $d+1$ distinct values. It also results in the property that no two polynomials of degree d can agree on more than d values.

Representation of Input

1. Sum of monomials $\{(a_i, i) | \text{coefficient of } x^i = a_i\}$
2. Set of roots
3. Oracle Access (An oracle is essentially a blackbox f, where we can give input a , and get $f(a)$ as the output quickly)

Polynomial Interpretation

Given $d+1$ values, $\{\alpha_0, \alpha_1, \alpha_2, \dots, \alpha_d\}$
We have to construct the univariate polynomial using oracle access (asking the oracle what's the value of $f(\alpha_i)$).
A polynomial can be written in the form -

$$P(x) = \sum_{i=0}^{d} a_i x^i$$

Using the oracle, we can find the values $P(\alpha_0), P(\alpha_1), \dots, P(\alpha_d)$

This can be represented in the form of a system of linear equations

$$\begin{bmatrix} \alpha_0^0 & \alpha_0^1 & \dots & \alpha_0^d \\ \alpha_1^0 & \alpha_1^1 & \dots & \alpha_1^d \\ & \vdots & & \\ \alpha_d^0 & \alpha_d^1 & \dots & \alpha_d^d \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_d \end{bmatrix} = \begin{bmatrix} P(\alpha_0) \\ P(\alpha_1) \\ \vdots \\ P(\alpha_d) \end{bmatrix}$$

The matrix is known as the Vandermonde matrix.
This can be solved in polynomial time using classic techniques like Gaussian elimination, etc.

Multivariate

Now the case of multivariate polynomials, how do we test the equality of $f(x_1, x_2), g(x_1, x_2)$? Unfortunately, the fundamental theorem that a polynomial of degree d has d roots, an example of this being $x^2 + y^2 - 1 = 0$.
We define the degree of a multivariate polynomial as d, where the sum of all exponents for each term $\leq d$

The number of such terms are $\binom{n+d}{d}$, which is bounded by n^d.

Testing the equality of two multivariate polynomials can be done in polynomial time using a randomized algorithm.

DeMillo-Lipton-Schwartz–Zippel Lemma

Let P be a non-zero polynomial on n variables of degree d.
Let S be the set of size at least $d + 1$.
We randomly sample n values, $a_1, a_2, \dots, a_i$ from S independently, uniformaly and randomly. The theorem states that

$$\mathbb{P}[P(a_1, a_2, \dots, a_n) = 0] \leq \frac{d}{|S|}$$

Proof

Proof using mathematical induction on n.
Base Case - $n = 1$
This case is trivial, as we've already stated that a d-degree polynomial can have at most d roots, therefore it satisfies the inequality.

Induction state - Assume it's true for $n \leq k-1$, prove for $n = k$

We have the variables $x_1, x_2, \ldots, x_n$

We can restructure the n variable polynomial in terms of coefficients of x_1

$$P(x_1, \ldots, x_n) = \sum_{i=0}^{d} x_1^i P_i(x_2, \ldots, x_n)$$

From our initial assumption, it's known that P is not identically 0, therefore we find the *largest* i such that P_i is not identically 0.

Note that for any i, $\deg P_i \leq d - i$, as the degree of $x_1^i P_i$ is at most d

We aim to prove the bound defined by the lemma using this structure

$$\begin{aligned}\mathbb{P}[A] &= \mathbb{P}[A \cap B] + \mathbb{P}[A \cap B^c] \\ &= \mathbb{P}[A|B]\mathbb{P}[B] + \mathbb{P}[A|B^c]\mathbb{P}[B^c] \\ &\leq \mathbb{P}[B] + \mathbb{P}[A|B^c]\end{aligned}$$

Here, we define events A, B as

$A - P(r_1, r_2, \ldots, r_n) = 0$

$B - P_i(r_2, r_3, \ldots, r_n) = 0$

Where r_i are i.u.a.r sampled from S.

$\mathbb{P}[B]$

We know that the degree of P_i is $\leq d - i$. Therefore, by the induction hypothesis, we get

$$\mathbb{P}[P_i(r_2, r_3, \ldots, r_n) = 0] \leq \frac{d-i}{|S|}$$

$\mathbb{P}[A|B^c]$

If $P_i(r_2, r_3, \ldots, r_n) \neq 0$, B^c occurs, we now know that the degree of the polynomial P is i, since all the $P_j \equiv 0, \forall j > i$. Therefore, with the help of the induction hypothesis, we can say that

$$\mathbb{P}[P(r_1, r_2, \ldots, r_n) = 0 | P_i(r_2, r_3, \ldots, r_n) \neq 0] \leq \frac{i}{|S|}$$

$\mathbb{P}[A]$

Therefore, we get

$$\begin{aligned}\mathbb{P}[P(r_1, r_2, \dots, r_n) = 0] &\leq \mathbb{P}[B] + \mathbb{P}[A|B^c] \\ &\leq \frac{d-i}{|S|} + \frac{i}{|S|} = \frac{d}{|S|}\end{aligned}$$

Hence, proved.

Using this lemma, we can test if two multivariate polynomials f, g are equal by identity testing polynomial $h = f - g$.

Randomized Algorithm for Polynomial Identity Testing

1. Pick a set S of size αd, where $\alpha > 1$
2. Evaluate at a point $\bar{a}$ (a vector of size n) whose corrdinates are sample i.u.a.r from S.
3. If $f(\bar{a}) = 0$ assert $f \equiv 0$, else $f \not\equiv 0$.

   $$\mathbb{P}[\text{Error}] \leq \frac{d}{|S|} = \frac{1}{\alpha}$$

   The possible error is that a non-zero polynomial is asserted as a zero-polynomial.

Verifying Matrix Multiplication

An application of Polynomial Identity Testing is verifying matrix multiplication
Given matrices A, B, C, we need to verify if $AB = C$.

This can be done in $O(n^w)$ (w is the matrix multiplication exponent), but we can speed this up using randomization.

Let S be a finite subset of $\mathbb{R}$, and we choose $\bar{x} = (x_1, x_2, \dots, x_n)$ i.u.a.r
We then verify if $ABx = Cx$. If true, we say that $AB = C$, else return $AB \neq C$.
Running this takes $O(n^2)$ time, as finding Bx, $A(Bx)$ and Cx are all $O(n^2)$ time operations.

Probability Analysis

ABx, Cx are both vectors, whose entries are linear forms in x.

$$ABx = (L_1(x), L_2(x), \ldots, L_n(x))^T$$
$$Cx = (L'_1(x), L'_2(x), \ldots, L'_n(x))^T$$

If $AB = C$, then we know that $\forall \bar{x}, AB\bar{x} = C\bar{x}$, which essentially mean $\forall 1 \leq i \leq n, L_i(\bar{x}) - L'_i(\bar{x}) = 0$
If $AB \neq C$, then there exists some vector $\bar{x}$ such that $L_i(\bar{x}) - L'_i(\bar{x}) \neq 0$

Lemma - For any linear polynomial $L(x)$

$$\mathbb{P}[L(\bar{a}) = 0] \leq \frac{1}{|S|}$$

Proof - We can arrive to this result using the principle of deferred decision.

We know that we can represent $L(x)$ as

$$L(x) = \sum_{i=1}^{n} b_i x_i$$

Assume we randomly pick the first $n-1$ values for x as $a_1, a_2, \ldots, a_{n-1}$, we'd get

$$L(x) = b_n x_n + \sum_{i=1}^{n-1} b_i a_i$$

For $L(x) = 0$,

$$b_n x_n = -\sum_{i=1}^{n-1} b_i a_i$$

There is at max 1 possible value for x_n in S for which this equality holds true, hence the probability of this being true is $\leq \frac{1}{|S|}$.

Claim - If $AB \neq C$, then $\mathbb{P}[ABx = Cx] \leq \frac{1}{|S|}$
Proof -
When $AB \neq C$, we say a *bad event* is when $L_i(\bar{x}) - L'_i(\bar{x}) = 0 \; \forall i \in [n]$, i.e we picked a vector such that $ABx = Cx$ event when $AB \neq C$, which causes us to incorrectly say that $AB = C$.

The probability of the bad event occurring is bounded by

$$\mathbb{P}\left[\bigwedge_{i=1}^{n}(L_i(\bar{x}) - L_i'(\bar{x}) = 0)\right] \leq \max_i \left\{\mathbb{P}[L_i(\bar{x}) - L_i'(\bar{x}) = 0]\right\}$$

And using the earlier Lemma, we can bound this further, giving us

$$\mathbb{P}\left[\bigwedge_{i=1}^{n}(L_i(\bar{x}) - L_i'(\bar{x}) = 0)\right] \leq \max_i \left\{\mathbb{P}[L_i(\bar{x}) - L_i'(\bar{x}) = 0]\right\} \leq \frac{1}{|S|}$$

Hence proved.