

CMPT 210: Probability and Computing

Lecture 11

Sharan Vaswani

October 10, 2024

(Basic) Freivald's Algorithm

Q: For $n \times n$ matrices A , B and D , is $D = AB$?

Algorithm:

1. Generate a random n -bit vector x , by making each bit x_i either 0 or 1 *independently* with probability $\frac{1}{2}$. E.g, for $n = 2$, toss a fair coin independently twice with the scheme – H is 0 and T is 1). If we get HT , then set $x = [0; 1]$.
2. Compute $t = Bx$ and $y = At = A(Bx)$ and $z = Dx$.
3. Output “yes” if $y = z$ (all entries need to be equal), else output “no”.

Computational complexity: Step 1 can be done in $O(n)$ time. Step 2 requires 3 matrix vector multiplications and can be done in $O(n^2)$ time. Step 3 requires comparing two n -dimensional vectors and can be done in $O(n)$ time. Hence, the total computational complexity is $O(n^2)$.

(Basic) Frievald's Algorithm

Let us run the algorithm on an example. Suppose we have generated $x = [1; 0]$

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad ; \quad B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad ; \quad D = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$
$$Bx = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad ; \quad y = A(Bx) = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad ; \quad z = Dx = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Hence the algorithm will correctly output “no” since $D \neq AB$.

Q: Suppose we have generated $x = [0; 0]$. What is y and z ?

In this case, $y = z$ and the algorithm will incorrectly output “yes” even though $D \neq AB$.

(Basic) Frievald's Algorithm

Let us run the algorithm on an example. Suppose we have generated $x = [1; 0]$.

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad ; \quad B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad ; \quad C = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$
$$Bx = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad ; \quad y = A(Bx) = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad ; \quad z = Cx = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Hence the algorithm will correctly output “yes” since $C = AB$.

Q: Suppose we have generated $x = [0; 1]$. What is y and z ?

In this case again, $y = z$ and the algorithm will correctly output “yes”.

(Basic) Freivald's Algorithm

Let us analyze the algorithm for general matrix multiplication.

Case (i): If $D = AB$, does the algorithm always output “yes”? Yes! Since $D = AB$, for any vector x , $Dx = ABx$.

Case (ii) If $D \neq AB$, does the algorithm always output “no”?

Claim: For any input matrices A, B, D if $D \neq AB$, then the (Basic) Freivald's algorithm will output “no” with probability $\geq \frac{1}{2}$.

	Yes	No
$D = AB$	1	0
$D \neq AB$	$< \frac{1}{2}$	$\geq \frac{1}{2}$

Table 1: Probabilities for Basic Freivalds Algorithm

(Basic) Frievald's Algorithm

Proof: If $D \neq AB$, we wish to compute the probability that algorithm outputs “yes” and prove that it less than $\frac{1}{2}$.

Define $E := (AB - D)$ and $r := Ex = (AB - D)x = y - z$. If $D \neq AB$, then $\exists(i, j)$ s.t. $E_{i,j} \neq 0$.

$$\begin{aligned}\Pr[\text{Algorithm outputs “yes”}] &= \Pr[y = z] = \Pr[r = \mathbf{0}] \\ &= \Pr[(r_1 = 0) \cap (r_2 = 0) \cap \dots \cap (r_i = 0) \cap \dots] \\ &= \Pr[(r_i = 0)] \Pr[(r_1 = 0) \cap (r_2 = 0) \cap \dots \cap (r_n = 0) | r_i = 0] \\ &\hspace{15em} (\text{By def. of conditional probability})\end{aligned}$$

$$\implies \Pr[\text{Algorithm outputs “yes”}] \leq \Pr[r_i = 0] \hspace{10em} (\text{Probabilities are in } [0, 1])$$

To complete the proof, on the next slide, we will prove that $\Pr[r_i = 0] \leq \frac{1}{2}$.

(Basic) Frievald's Algorithm

$$r_i = \sum_{k=1}^n E_{i,k} x_k = E_{i,j} x_j + \sum_{k \neq j} E_{i,k} x_k = E_{i,j} x_j + \omega \quad (\omega := \sum_{k \neq j} E_{i,k} x_k)$$

$$\Pr[r_i = 0] = \Pr[r_i = 0 | \omega = 0] \Pr[\omega = 0] + \Pr[r_i = 0 | \omega \neq 0] \Pr[\omega \neq 0]$$

(By the law of total probability)

$$\Pr[r_i = 0 | \omega = 0] = \Pr[x_j = 0] = \frac{1}{2} \quad (\text{Since } E_{i,j} \neq 0 \text{ and } \Pr[x_j = 1] = \frac{1}{2})$$

$$\Pr[r_i = 0 | \omega \neq 0] = \Pr[(x_j = 1) \cap E_{i,j} = -\omega] = \Pr[(x_j = 1)] \Pr[E_{i,j} = -\omega | x_j = 1]$$

(By def. of conditional probability)

$$\implies \Pr[r_i = 0 | \omega \neq 0] \leq \Pr[(x_j = 1)] = \frac{1}{2} \quad (\text{Probabilities are in } [0, 1], \Pr[x_j = 1] = \frac{1}{2})$$

$$\implies \Pr[r_i = 0] \leq \frac{1}{2} \Pr[\omega = 0] + \frac{1}{2} \Pr[\omega \neq 0] = \frac{1}{2} \Pr[\omega = 0] + \frac{1}{2} [1 - \Pr[\omega = 0]] = \frac{1}{2}$$

($\Pr[E^c] = 1 - \Pr[E]$)

$$\implies \Pr[\text{Algorithm outputs "yes"}] \leq \Pr[r_i = 0] \leq \frac{1}{2}.$$

(Basic) Freivald's Algorithm

- Hence, if $D \neq AB$, the Algorithm outputs “yes” with probability $\leq \frac{1}{2} \implies$ the Algorithm outputs “no” with probability $\geq \frac{1}{2}$.
- In the worst case, the algorithm can be incorrect half the time! We promised the algorithm would return the correct answer with “high” probability close to 1.
- A common trick in randomized algorithms is to have m independent trials of an algorithm and aggregate the answer in some way, reducing the probability of error, thus *amplifying the probability of success*.

Questions?

Frievald's Algorithm

- By repeating the *Basic Frievald's Algorithm* m times, we will amplify the probability of success. The resulting complete Frievald's Algorithm is given by:

- 1 Run the Basic Frievald's Algorithm for m independent runs.
- 2 If *any* run of the Basic Frievald's Algorithm outputs "no", output "no".
- 3 If *all* runs of the Basic Frievald's Algorithm output "yes", output "yes".

	Yes	No
$D = AB$	1	0
$D \neq AB$	$< \frac{1}{2^m}$	$\geq 1 - \frac{1}{2^m}$

Table 2: Probabilities for Frievald's Algorithm

If $m = 20$, then Frievald's algorithm will make mistake with probability $1/2^{20} \approx 10^{-6}$.

Computational Complexity: $O(mn^2)$

Probability Amplification

- Consider a randomized algorithm \mathcal{A} that is supposed to solve a binary decision problem i.e. it is supposed to answer either Yes or No. It has a one-sided error – (i) if the true answer is Yes, then the algorithm \mathcal{A} correctly outputs Yes with probability 1, but (ii) if the true answer is No, the algorithm \mathcal{A} incorrectly outputs Yes with probability $\leq \frac{1}{2}$.
- Let us define a new algorithm \mathcal{B} that runs algorithm \mathcal{A} m times, and if *any* run of \mathcal{A} outputs No, algorithm \mathcal{B} outputs No. If *all* runs of \mathcal{A} output Yes, algorithm \mathcal{B} outputs Yes.

Q: What is the probability that algorithm \mathcal{B} correctly outputs Yes if the true answer is Yes, and correctly outputs No if the true answer is No?

Probability Amplification - Analysis

If A_i denotes run i of Algorithm \mathcal{A} , then

$$\begin{aligned} & \Pr[\mathcal{B} \text{ outputs Yes} \mid \text{true answer is Yes}] \\ &= \Pr[\mathcal{A}_1 \text{ outputs Yes} \cap \mathcal{A}_2 \text{ outputs Yes} \cap \dots \cap \mathcal{A}_m \text{ outputs Yes} \mid \text{true answer is Yes}] \\ &= \prod_{i=1}^m \Pr[\mathcal{A}_i \text{ outputs Yes} \mid \text{true answer is Yes}] = 1 \end{aligned} \quad \text{(Independence of runs)}$$

$$\begin{aligned} & \Pr[\mathcal{B} \text{ outputs No} \mid \text{true answer is No}] \\ &= 1 - \Pr[\mathcal{B} \text{ outputs Yes} \mid \text{true answer is No}] \\ &= 1 - \Pr[\mathcal{A}_1 \text{ outputs Yes} \cap \mathcal{A}_2 \text{ outputs Yes} \cap \dots \cap \mathcal{A}_m \text{ outputs Yes} \mid \text{true answer is No}] \\ &= 1 - \prod_{i=1}^m \Pr[\mathcal{A}_i \text{ outputs Yes} \mid \text{true answer is No}] \geq 1 - \frac{1}{2^m}. \end{aligned}$$

When the true answer is Yes, both \mathcal{B} and \mathcal{A} correctly output Yes. When the true answer is No, \mathcal{A} incorrectly outputs Yes with probability $< \frac{1}{2}$, but \mathcal{B} incorrectly outputs Yes with probability $< \frac{1}{2^m} \ll \frac{1}{2}$. By repeating the experiment, we have “amplified” the probability of success.

Questions?