Assignment 1: Randomized Algorithm Design

Palash Dey
Indian Institute of Technology, Kharagpur

January 20, 2020

Assume that the random variables are discrete if not explicitly mentioned. Submit the answer of the questions colored red by January 21 in my mail box.

---

1. Let $\mathcal{X}_i, i \in [n]$ be n random variables each with finite support. Then prove the following.

$$\text{var}\left(\sum_{i=1}^{n} \mathcal{X}_i\right) = \sum_{i=1}^{n} \text{var}\left(\mathcal{X}_i\right) + 2 \sum_{1 \leqslant i < j \leqslant n} \text{cov}(\mathcal{X}_i, \mathcal{X}_j)$$

   where for any two random variables $\mathcal{X}$ and $\mathcal{Y}$, we define $\text{cov}(\mathcal{X}, \mathcal{Y}) = \mathbb{E}[\mathcal{X}\mathcal{Y}] - \mathbb{E}[\mathcal{X}]\mathbb{E}[\mathcal{Y}]$.

2. Let $\mathcal{X}$ and $\mathcal{Y}$ be two independent random variables. Then prove that $\mathbb{E}[\mathcal{X}\mathcal{Y}] = \mathbb{E}[\mathcal{X}]\mathbb{E}[\mathcal{Y}]$. From this conclude that, for n pairwise random variables $\mathcal{X}_i, i \in [n]$, we have the following.

$$\text{var}\left(\sum_{i=1}^{n} \mathcal{X}_i\right) = \sum_{i=1}^{n} \text{var}(\mathcal{X}_i)$$

3. Fix any input sequence of n integers to the quick sort algorithm. Let $\mathcal{X}$ be the random variable denoting the number of comparisons the the quick sort algorithm makes on the input sequence. Then prove that $\text{var}(\mathcal{X}) = \mathcal{O}(n^2)$.

4. Let $A_i, i \in [n]$ be n objects each having two attributes A_i^x and A_i^y. The attribute y is 0 for every A_i. Suppose we have a deterministic quick sort algorithm that can sort $A_i, i \in [n]$ on the attribute x or on the attribute y. Can you use this deterministic quick sort algorithm to design a randomized algorithm to sort $A_i, i \in [n]$ on the attribute x which makes an expected $\mathcal{O}(n \log n)$ comparisons? Please prove that your algorithm indeed makes $\mathcal{O}(n \log n)$ comparisons on expectation.

5. Let $\mathcal{X}_i, i \in [n]$ be n pairwise independent random variables each taking values in $\{0, 1\}$ with expectation μ and $\mathcal{S} = \sum_{i=1}^{n} \mathcal{X}_i$. Then for any positive real number δ we have the following.

$$\Pr\left[\mathcal{S} \leqslant (1-\delta)\mu\right] \leqslant \left(\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu}$$

6. Show that the expected number of balls one needs to through randomly into m bins to have every bin at least one ball is $\mathcal{O}(m \log m)$.

7. Give an example of a random variable whose expectation exists but variance does not exist.

8. Find the expectation and variance of the number of swaps that the bubble sort algorithm performs on a uniformly random permutation of n distinct integers.

9. Prove the weak law of large numbers using Chebyshev inequality. The weak law of large number states that, for random variables $X_i, i \in \mathbb{N}$ which are distributes independently and identically with mean μ and variance σ^2, we have the following for any constant $\varepsilon > 0$

$$\lim_{n\to\infty} \Pr\left[\left|\frac{X_1 + X_2 + \cdots + X_n}{n} - \mu\right| > \varepsilon\right] = 0$$

10. Let $\mathcal{X}_i, i \in [n]$ be n pairwise independent random variables each taking values in $\{0, 2\}$ with expectation μ and $\mathcal{S} = \sum_{i=1}^{n} \mathcal{X}_i$. Use standard Chernoff bound proved in class to upper bound the probability that $\mathcal{S}$ takes value more than $(1+\delta)\mu$.

11. Let $\mathcal{X}$ be a random variable with expectation μ and variance σ^2. Then for any $t \in \mathbb{R}_{\geqslant 0}$, prove the following.

$$\Pr\left[\mathcal{X} - \mu \geqslant t\sigma\right] \leqslant \frac{1}{1+t^2} \text{ and } \Pr\left[|\mathcal{X} - \mu| \geqslant t\sigma\right] \leqslant \frac{2}{1+t^2}$$

12. Let $\mathcal{X}$ be a non-negative integer valued random variable with positive expectation. Then prove the following.

$$\Pr\left[\mathcal{X} = 0\right] \leqslant \frac{\mathbb{E}[\mathcal{X}^2] - \mathbb{E}[\mathcal{X}]^2}{\mathbb{E}[\mathcal{X}]^2} \text{ and } \frac{\mathbb{E}[\mathcal{X}]^2}{\mathbb{E}[\mathcal{X}^2]} \leqslant \Pr[\mathcal{X} \neq 0] \leqslant \mathbb{E}[\mathcal{X}]$$

13. Design a randomized algorithm for computing if a given directed graph contains a cycle of length at least k. Your algorithm should run in time $\mathcal{O}(c^k \text{poly}(n))$ where c is some constant.