INDIAN STATISTICAL INSTITUTE

M. Tech (CS) - II Year, 2019-2020 (Semester - I)

Topics in Algorithms and Complexity

Problem Sheet II

(Q1) Show that for a random variable X and for any $a > 0$,

$$\Pr\left(|X - \mathbb{E}[X]| \geq a\right) \leq \frac{\text{Var}[X]}{a^2}$$

and hence, show the following for any $t > 1$, where $\sigma(X)$ is the standard deviation of X

$$\Pr(|X - \mathbb{E}[X]| \geq t\,\sigma(X)) \leq \frac{1}{t^2},$$

$$\Pr(|X - \mathbb{E}[X]| \geq t\,\mathbb{E}(X)) \leq \frac{\mathbb{V}ar[X]}{t^2(\mathbb{E}[X])^2}.$$

(Q2) Consider a sequence of n independent coin flips with a probability of $\frac{1}{2}$ of falling heads. Use both Markov and Chebyshev inequality to deduce concentration bounds on the probability of obtaining more than $c \cdot n$ heads, where $c > \frac{1}{2}$ is a constant.

(Q3) Consider the *coupon collector* problem. Let X be the random variable indicating the number of buys to collect n different brand of coupons.

- Find $\mathbb{E}[X]$.
- Use Markov inequality to bound $\Pr(X \geq c\mathbb{E}[X])$, where $c > 1$.
- Use Chebyshev inequality to bound $\Pr(|X - c\,\mathbb{E}[X]| \geq c\,\mathbb{E}[X])$, where $c > 1$.
- Use union bound to compute the probability that some coupon has not been collected after $n \ln n + cn$ steps.
- Use union bound to compute the probability that all coupons are not collected after $2n \ln n$ steps.

(Q4) Consider the following randomized median finding algorithm from a set S of n elements. Pick a multi-set (i.e., sets with repetition) R of size $\lceil n^{\delta} \rceil$ elements in S, chosen independently and uniformly at random with replacement. Sort the set R. Let u_1 be the δ_1-th and u_2 be the δ_2-th smallest element in the sorted set R, $\delta_2 > \delta_1$. Now, compare every element in the set S to u_1 and u_2 and compute the sets

- $C_1 = \{x \in S \mid u_1 \leq x \leq u_2\}$
- $C_2 = \{x \in S \mid x < u_1\}$
- $C_3 = \{x \in S \mid x > u_2\}$

If $|C_2| > \frac{n}{2}$ or $|C_3| > \frac{n}{2}$, then the algorithm FAILS to find the median. If $|C_1| \leq \delta_3 n^{\delta_4}$, then sort C_1, otherwise FAIL. Output the $(\lfloor n/2 \rfloor - |C_2| + 1)$-th element in the sorted order of C_1.

Our goal will be to get a linear time algorithm for median finding. The design criteria for that will be to ensure that sets R and C_1 are neither too big nor too small so that

(i) the set C_1 is large enough to include the median with high probability and

(ii) C_1 is sufficiently small so that it can be sorted in sub-linear, i.e., $o(n)$ time.

Our final goal is to prove that the probability that the randomized median finding algorithm FAILs is bounded above by $n^{-1/4}$. Use Chebyshev's inequality to find the above failure probability. In doing so, choose the parameters δ_1, δ_2, δ_3 and δ_4 properly so that the bound works out to $n^{-1/4}$.

Show how you can obtain an iterative algorithm that never fails by repeating the above randomized median finding algorithm until it succeeds in finding the median. This can be seen as a *waiting for success* bound. Find the expected running time.

(Q5) Let $X_1, \ldots, X_n$ be 0-1 independent random variables (Poisson trials) such that $\Pr(X_i) = p_i$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = \mathbb{E}[X]$. Then show that the following Chernoff bounds hold:

- for any $\delta > 0$, $\Pr(X \geq (1+\delta)\mu) < \left(\frac{e^\delta}{(1+\delta)^{(1+\delta)}}\right)^\mu$;
- for $0 < \delta \leq 1$, $\Pr(X \geq (1+\delta)\mu) \leq e^{-\mu\delta^2/3}$.
- for $R \geq 6\mu$, $\Pr(X \geq R) \leq 2^{-R}$

(Q6) Let $X_1, \ldots, X_n$ be independent random variables such that $\Pr(X_i = 1) = \Pr(X_i = -1) = \frac{1}{2}$. Let $X = \sum_{i=1}^{n} X_i$. For any $a > 0$, show that $\Pr(X \geq a) \leq e^{-a^2/2n}$.

(Q7) Let $X_1, \ldots, X_n$ be independent random variables such that $\Pr(X_i = 1) = \Pr(X_i = -1) = \frac{1}{2}$. Let $X = \sum_{i=1}^{n} X_i$. For any $a > 0$, show that $\Pr(|X| \geq a) \leq 2e^{-a^2/2n}$.

(Q8) Let $X_1, \ldots, X_n$ be independent random variables such that $\Pr(X_i = 1) = \Pr(X_i = 0) = \frac{1}{2}$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = \mathbb{E}[X] = \frac{n}{2}$.

- for any $a > 0$, $\Pr(X \geq \mu + a) \leq e^{-2a^2/n}$.
- for any $\delta > 0$, $\Pr(X \geq (1+\delta)\mu) \leq e^{-\delta^2\mu}$.

(Q9) Consider a set system (or, a hypergraph), (V, F) where $V = [n]$ is the ground set and $F = \{A_1, \ldots, A_m\}$, where $A_i \subseteq V$. Let $\chi : V \to \{-1, +1\}$, i.e. χ is a coloring (function) that assigns $+1$ or -1 values to the vertices. For any $A \subseteq V$, define $\chi(A) = \sum_{i \in A} \chi(i)$. Define the discrepancy of F with respect to χ by $\text{disc}_\chi(F) = \max_{A_i \in F} |\chi(A_i)|$. The discrepancy of F is $\text{disc}(F) = \min_\chi \text{disc}_\chi(F)$. Show that a random coloring on V can achieve $\text{disc}(F) \leq O(\sqrt{n \log m})$ with high probability.

(Q10) Show using Chernoff bounds that when there are n processors and $\Omega(n \log n)$ jobs and jobs are randomly assigned to processors, then with high probability every processor will have a load between half and twice the average.

(Q11) We had seen in the Algorithms class that the *randomized quicksort* does $O(n \log n)$ expected comparisons. Now show that the *randomized quicksort* does $O(n \log n)$ comparisons with high probability.