CS 503 Randomized Algorithms Midsem

Sparsh Sinha

Roll No - 170101085

1. Suppose randomized Monte Carlo Algorithm for a decision problem P has two sided error. Let the probability that the algorithm gives correct answer be $(\frac{1}{2} + \epsilon)$ for some ϵ. Suppose to have better confidence on the output, you repeat the algorithm t times and accept that result which comes with majority. What is the probability (bounded above) that the majority algorithm reports an error?

Solution: Let us assume that P has two outputs Y and N since it is a decision problem. Without loss of generality, let us assume that the correct output is Y. Let us define the repetition of the algorithm t times as our experiment in which "success" occurs if we obtain N. Let us consider a random variable X as the number of times we obtain N in this experiment. Clearly, $X \sim Bin(t, \frac{1}{2} - \epsilon)$, since the probability of success is $(\frac{1}{2} - \epsilon)$. We define the majority as the case when N occurs $\lceil \frac{t}{2} \rceil$ times. Hence, the majority algorithm reports error with probability given by $\Pr(X \geq \lceil \frac{t}{2} \rceil)$. We note that for any non-trivial experiment we have $t \geq 1$ and hence $\lceil \frac{t}{2} \rceil > 0$, so by Markov's inequality we have

$$\Pr\left(X \geq \left\lceil \frac{t}{2} \right\rceil\right) \leq \frac{E(X)}{\lceil \frac{t}{2} \rceil} = \frac{\left(\frac{t}{2} - \epsilon t\right)}{\lceil \frac{t}{2} \rceil} \leq \frac{\left(\frac{t}{2} - \epsilon t\right)}{\frac{t}{2}} \tag{1}$$

$$\implies \Pr\left(X \geq \left\lceil \frac{t}{2} \right\rceil\right) \leq (1 - 2\epsilon) \tag{2}$$

In order to obtain better estimates we can get the actual expression for probability and try to bound it as follows:

$$\implies \Pr\left(X \geq \left\lceil \frac{t}{2} \right\rceil\right) = \sum_{k=\lceil \frac{t}{2} \rceil}^{k=t} \binom{t}{k} \left(\frac{1}{2} - \epsilon\right)^{k} \left(\frac{1}{2} + \epsilon\right)^{t-k} \tag{3}$$

$$\leq \sum_{k=\lceil \frac{t}{2} \rceil}^{k=t} \binom{t}{k} \left(\frac{1}{2} - \epsilon\right)^{\lceil \frac{t}{2} \rceil} \left(\frac{1}{2} + \epsilon\right)^{\lceil \frac{t}{2} \rceil} \tag{4}$$

$$\leq \left(\frac{1}{4} - \epsilon^2\right)^{\lceil \frac{t}{2} \rceil} \sum_{k=\lceil \frac{t}{2} \rceil}^{k=t} \binom{t}{k} \tag{5}$$

$$\leq \left(\frac{1}{4} - \epsilon^2\right)^{\lceil \frac{t}{2} \rceil} 2^t \tag{6}$$

$$\leq \left(1 - 4\epsilon^2\right)^{\lceil \frac{t}{2} \rceil} \tag{7}$$

2. Consider a Monte Carlo algorithm A for a problem P whose expected running time is at most $T(n)$ on any instance of size n and that produces a correct solution with probability $\gamma(n)$. Suppose further that given a solution to P, we can verify its correctness in time $\tau(n)$. Show how to obtain a Las Vegas algorithm that always gives a correct answer to P and runs in expected time at most $(T(n) + \tau(n))/\gamma(n)$

Solution:

Algorithm: First we run the algorithm A on the given input. Then we verify the solution obtained to P. If the verification is successful(meaning that the solution is verified to be correct), then we terminate the algorithm and return our solution. Otherwise we repeat the algorithm again.

The above algorithm is a Las Vegas Algorithm because it returns a solution if and only if the verification is successful. This means that whenever we return a solution it is always correct.

To analyze the running time let us define a random variable X as the number of iterations required for the algorithm above to terminate. Since, this is possible only if the algorithm A gives correct output, so this occurs with probability of $\gamma(n)$. Therefore, quite clearly $X \sim Geo(\gamma(n))$, because X can be thought of as the number of trials to obtain the first success(which is our correct solution). Let, the running time of the algorithm be T. So, we can easily see that

$$T \leq (T(n) + \tau(n)) \times X \tag{8}$$

which is the time taken to run algorithm A followed by verification, for each iteration. (Runtime for each iteration is $(T(n) + \tau(n))$ and number of iterations is X)

Since, $T(n)$ and $\tau(n)$ are constant(here constant means that they are deterministic values, not random). Hence, we have the following results(using properties of expectation):

$$E(T) \leq (T(n) + \tau(n)) \times E(X) \tag{9}$$

$$\implies E(T) \leq (T(n) + \tau(n)) \times \frac{1}{\gamma(n)} \tag{10}$$

Here, the expected value of the geometric random variable has been directly used $(E(X) = \frac{1}{\gamma(n)})$.

Hence the expected running time of our Las Vegas algorithm is atmost $\frac{(T(n)+\tau(n))}{\gamma(n)}$

3. Consider the Coupon Collector's Problem where the goal is to gather k_i coupons of type i $(1 \leq i \leq n)$. There is altogether n different type of coupons. Assuming the distribution of the coupons uniform, analyze the expected time to reach the goal.

Solution: Let us define $a = \sum_{i=1}^{i=n} k_i$. So, there are a items in total. Let's say that a coupon j is good if it has occurred at most $(k_j - 1)$ times amongst the first $(i-1)$ times. Also, define N_{ij} as 1, if jth coupon is good after ith iterations, else 0. Let us denote a random variable X_i, $(1 <= i <= a)$ which measures the time required to obtain a new "good" coupon after $(i-1)$ iterations, i.e. on the ith step. So, here "success" (i.e occurrence of good coupon) occurs if we obtain a "good" coupon. So, the probability of success at the ith step, is given by:

$$P_i = \sum_{j=1}^{j=n} \frac{N_{(i-1)j}}{n} \tag{11}$$

Here we note that, P_i is another random variable.

It must be obvious that that the total time required is given by

$$T = X_1 + X_2 + \cdots + X_a \tag{12}$$

Using linearity of expectation we have,

$$E(T) = E(X_1 + X_2 + \cdots + X_a) = E(X_1) + E(X_2) + \cdots + E(X_a) \tag{13}$$

Let us now focus on one term,

$$E(X_i) = E(E(X_i|N_{(i-1)1}, N_{(i-1)2}, \ldots, N_{(i-1)n})) \tag{14}$$

Above equation follows from the property of expectation that $E(X) = E(E(X|Y))$.

We immediately observe that $(X_i|N_{(i-1)1}, N_{(i-1)2}, \ldots, N_{(i-1)n})$ is a geometric random variable with success probability (P_i)

So, we have

$$E(X_i) = E\left(\frac{1}{P_i}\right) \tag{15}$$

We note that the function $\frac{1}{X}$ is a convex function so, we can bound the expectation from below using Jensen's inequality as:

$$E\left(\frac{1}{P_i}\right) >= \frac{1}{E(P_i)} \tag{16}$$

Now, we can calculate $E(P_i)$ using linearity of expectation as follows:

$$E(P_i) = \sum_{j=1}^{j=n} \frac{E(N_{(i-1)j})}{n} \tag{17}$$

Now, we only need to figure out N_{ij} for a given i. Since N_ij is an indicator random variable, so it's expectation is equal to probability that it is 1, which is the probability that jth item occurs at most $(k_j - 1)$ times in the i iterations. This, can be estimated as follows (using binomials)

$$P(N_{ij} = 1) = \sum_{x=0}^{x=(k_j-1)} \left(\binom{i}{x}\left(\frac{1}{n}\right)^x \left(\frac{(n-1)}{n}\right)^{i-x}\right) \tag{18}$$

$$P(N_{ij} = 1) = \left(\frac{n-1}{n}\right)^i \left(\sum_{x=0}^{x=(k_j-1)} \left(\binom{i}{x}\left(\frac{1}{n-1}\right)^x\right)\right) \tag{19}$$

$$P(N_{ij} = 1) \leq \left(\frac{n-1}{n}\right)^i \left(\sum_{x=0}^{x=(k_j-1)} \left(2^i\left(\frac{1}{n-1}\right)^x\right)\right) \tag{20}$$

$$P(N_{ij}=1) \leq \left((2)\left(\frac{n-1}{n}\right)\right)^{i} \left(\sum_{x=0}^{x=(k_j-1)} \left(\left(\frac{1}{n-1}\right)^{x}\right)\right) \tag{21}$$

$$P(N_{ij}=1) \leq \left((2)\left(\frac{n-1}{n}\right)\right)^{i} J(k_j-1) \tag{22}$$

Putting it in the final equation we get,

$$E(X) >= \sum_{i} \frac{1}{\sum_j \left((2)\left(\frac{n-1}{n}\right)\right)^{i} J(k_j-1)} \tag{23}$$

Here, $J(\alpha)$ denotes $\sum_{x=0}^{x=\alpha} \left(\frac{1}{n-1}\right)^{x}$