E0234: Solution Sketch of Assignment 3

February 8, 2016

It is highly recommended you do not google for the answers to the questions below. You can discuss with your friends, but then mention that in your submission. The writing should solely be your own.

0. **(Not to be submitted but recommended.)** Implement the randomized median finding algorithm where the sorting algorithm is also implemented by you. Your sorting algorithm should also keep account of number of array comparisons it makes. Run the algorithm on an array of a million entries where each entry is a random real between 0 and 100. Compare the time **and** number of comparisons that the randomized median finding algorithm makes with the those of (a) the naive algorithm which just sorts (using your sorting implementation) and returns the middle entry, and (b) The Find algorithm done in class. Do you see a big difference?

1. (**From high probability to expectation.**) In class, we proved that the randomized median finding algorithm found the median with $2n + o(n)$ comparisons with probability $1 - o(1)$.

   Solution sketch: Follows immediately from the fact that the random variable denoting the number of times the "outer loop" of the algorithm "fails" is a geometric random variable with rate $1 - o(1)$.

2. Prove that the expected running time of the Find subroutine done in class (see MR, Chap 1.4) is $O(n)$.

   Solution sketch: Enough to prove that the number of comparisons made by the algorithm is $O(n)$. Prove that the expected running time of the algorithm is also $2n + o(n)$. Let X be the random variable counting the number of comparisons made by the algorithm. Let X_{ij} be the indicator random variable for the event that i^{th} and j^{th} element (in the sorted order) are compared by the algorithm for $1 \leq i < j \leq n$. Then we have the following.

$$X_{ij} = \begin{cases} Ber(\frac{2}{j-k+1}) & \text{if } k \leq i < j \\ Ber(\frac{2}{k-i+1}) & \text{if } i \leq j < k \\ Ber(\frac{2}{j-i+1}) & \text{if } i \leq k < j \end{cases}$$

   We now have the following.

$$
\begin{aligned}
X &= \sum_{1 \le i < j \le n} X_{ij} \\
E[X] &= \sum_{1 \le i < j \le n} E[X_{ij}] \\
&= \sum_{i=1}^{k-1} \sum_{j=i+1}^{k-1} \frac{2}{k-i+1} + \sum_{j=k+1}^{n} \sum_{i=k}^{j-1} \frac{2}{j-k+1} + \sum_{i=1}^{k} \sum_{j=k}^{n} \frac{2}{j-i+1} \\
&\le \sum_{i=1}^{k-1} 2 + \sum_{j=k+1}^{n} 2 + 2n \\
&\le 4n
\end{aligned}
$$

The second inequality follows from the fact that $\sum_{i=1}^{k} \sum_{j=k}^{n} \frac{2}{j-i+1} \le 2n$ since each $\frac{2}{j-i+1}$ term appears exactly $j-i+1$ many times in the summation and we have $1 \le j-i+1 \le n$.

3. (**Non-uniform Birthday Paradox.**) Consider m balls being thrown randomly into n bins, however, the distribution is not uniformly at random. In particular, each ball lands in bin 1 with probability p_1, bin 2 with probability p_2, and so on, where $\mathbf{p} := (p_1, \ldots, p_n)$ is a probability vector with $\sum_{i=1}^{n} p_i = 1$ and $p_i \ge 0$ for all i. What is the smallest m for which you can say that there would be at least one bin with at least 2 balls with probability $> 1/2$? **Hint:** Observe that if all $p_i = 1/n$, this is the birthday paradox question, and so $m \approx \sqrt{2n}$. Also note that if some p_i is close to 1, then the required $m = O(1)$.

   Solution sketch: Let X_{ij} be the indicator random variable for the event that balls i and j collide for $1 \le i < j \le m$. Then $X_{ij} \sim \text{Ber}(||p||_2^2)$. Let us define $X = \sum_{1 \le i < j \le m} X_{ij}$. Then we have the following.

$$E[X] = \binom{m}{2} ||p||_2^2$$

$$Var(X) \le \binom{m}{2} ||p||_2^2 + \frac{m^3}{6} ||p||_3^3$$

   Now we have the following from Chebyshev.

$$Pr[X > 0] \ge Pr[|X - E[X]| < E[X]] \ge 1 - \frac{1}{\binom{m}{2} ||p||_2^2} - \frac{m^3 ||p||_3^3}{6\binom{m}{2}^2 ||p||_2^4}$$

   From the inequality above, there exists a constant $c > 0$ such that for $m > \frac{c}{||p||_2}$, we have the probability that at least one bin with at least 2 balls is at least $\frac{1}{2}$.

4. (**Variance of QuickSort**.) Compute the variance of the number of comparisons made by the QuickSort algorithm. We are looking for the correct order of magnitude and not the exact constants. **Hint:** Recall the computation of the expectation from the 1st lecture. Apply first principles.

Solution sketch: Let X be the random variable counting the number of comparisons made by the algorithm. Let X_{ij} be the indicator random variable for the event that i^{th} and j^{th} element (in the sorted order) are compared by the algorithm for $1 \leq i < j \leq n$. Then we know that $X_{ij} \sim \text{Ber}(\frac{1}{j-i+1})$. From definitions we have the following.

$$\begin{aligned}
X &= \sum_{1 \leq i < j \leq n} X_{ij} \\
Var(X) &= \sum_{1 \leq i < j \leq n} Var(X_{ij}) + \sum_{1 \leq i < j \leq n, 1 \leq k < \ell \leq n} Cov(X_{ij}, X_{k\ell}) \\
Var(X) &\leq \sum_{1 \leq i < j \leq n} E(X_{ij}) + \sum_{1 \leq i < j \leq n, 1 \leq k < \ell \leq n} Cov(X_{ij}, X_{k\ell}) \\
Var(X) &\leq n \log n + \sum_{1 \leq i < j \leq n, 1 \leq k < \ell \leq n} Cov(X_{ij}, X_{k\ell})
\end{aligned}$$

Now we compute $\text{Cov}(X_{ij}, X_{k\ell})$ for $1 \leq i < j \leq n, 1 \leq k < \ell \leq n$ by the following case analysis. Let the sorted array be $A[1 \ldots n]$.

- Case 1: $1 \leq i < j < k < \ell \leq n$
  $\text{Cov}(X_{ij}, X_{k\ell}) = 0$ since X_{ij} and $X_{k\ell}$ are independent in this case.
- Case 2: $1 \leq k < i < \ell < j \leq n$

$$\begin{aligned}
Cov(X_{ij}, X_{k\ell}) &= E[X_{ij}X_{k\ell}] - E[X_{ij}]E[X_{k\ell}] \\
&= Pr[k \text{ is chosen first in } A[k \ldots j]] \cdot Pr[i \text{ or } j \text{ is chosen first in } A[i \ldots j]] \\
&\quad + Pr[k \text{ is chosen first in } A[k \ldots j]] \cdot Pr[k \text{ or } \ell \text{ is chosen first in } A[k \ldots \ell]] \\
&\quad - \frac{4}{(j-i+1)(\ell-k+1)} \\
&= \frac{2(\ell - j + k - i)}{(j-k+1)(j-i+1)(\ell-k+1)} \\
&\leq 0
\end{aligned}$$

- Case 3: $1 \leq i < k < j < \ell \leq n$
  Following argument along the line same as case 2, we have: $Cov(X_{ij}, X_{k\ell}) \leq 0$.
- Case 4: $1 \leq k < i < j < \ell \leq n$
  Following argument along the line same as case 2, we have: $Cov(X_{ij}, X_{k\ell}) = 0$.
- Case 5: $1 \leq i = k < \ell < j$

$$
\begin{aligned}
Cov(X_{ij}, X_{k\ell}) &= E[X_{ij}X_{k\ell}] - E[X_{ij}]E[X_{k\ell}] \\
&= Pr[j \text{ is chosen first in } A[k \dots j]] \cdot Pr[k \text{ or } \ell \text{ is chosen first in } A[k \dots \ell]] \\
&\quad + Pr[i \text{ is chosen first in } A[i \dots j]] \\
&\quad - \frac{4}{(j-i+1)(\ell-k+1)} \\
&= \frac{1}{j-i+1} - \frac{3}{(j-i+1)(\ell-i+1)} \\
&\leq \frac{1}{j-i+1}
\end{aligned}
$$

Hence, we have the following.

$$
\begin{aligned}
\sum_{1 \leq i < j \leq n, 1 \leq k < \ell \leq n} Cov(X_{ij}, X_{k\ell}) &= \sum_{1 \leq i < j \leq n} \sum_{1 \leq k < \ell \leq n} Cov(X_{ij}, X_{k\ell}) \\
&\leq \sum_{1 \leq i < j \leq n} 1 \\
&\leq n^2
\end{aligned}
$$

Hence, $Var(X) = O(n^2)$.