Exercises of Chapter 3

Chuang-Chieh Lin*

Department of Computer Science and Information Engineering,
National Chung Cheng University, Ming-Hsiung, Chiayi 621, Taiwan.

Exercise 3.1. *Let* X *be a number chosen uniformly at random from* $[1, n]$*. Find* $\mathbf{Var}[X]$*.*

Solution. Since we have

$$\mathbf{E}[X] = \sum_{i=1}^{n} \frac{1}{n} \cdot i = \frac{1+n}{2}$$

and

$$\mathbf{E}[X^2] = \sum_{i=1}^{n} \frac{1}{n} \cdot i^2 = \frac{(n+1)(2n+1)}{6}$$

we obtain that

$$\mathbf{Var}[X] = \mathbf{E}[X^2] - (\mathbf{E}[X])^2 = \frac{(n+1)(n-1)}{12}.$$

□

Exercise 3.2. *Let* X *be a number chosen uniformly at random from* $[-k, k]$*. Find* $\mathbf{Var}[X]$*.*

Solution. Similar to Exercise 3.1, we have

$$\mathbf{E}[X] = \sum_{i=-k}^{-1} \frac{1}{2k+1} \cdot i + \sum_{i=1}^{k} \frac{1}{2k+1} \cdot i + 0 \cdot \frac{1}{2k+1} = 0$$

and

$$\mathbf{E}[X^2] = 2\sum_{i=1}^{k} \frac{1}{2k+1} \cdot i^2 + 0 = \frac{k(k+1)}{3},$$

so we derive that

$$\mathbf{Var}[X] = \frac{k(k+1)}{3}.$$

□

Exercise 3.3. *Suppose that we roll a standard fair die 100 times. Let* X *be the sum of the numbers that appear over the 100 rolls. Use Chebyshev's inequality to bound* $\mathbf{Pr}[|X - 350| \geq 50]$*.*

*Email address: lincc@cs.ccu.edu.tw

Solution. Let X_i be a random variable denoting the outcome of the ith rolling of a standard fair die. Note that X_i's are mutually independent for all i. We can easily derive $\mathbf{E}[X_i] = 7/2$ and

$$\mathbf{Var}[X_i] = \mathbf{E}[X^2] - (\mathbf{E}[X])^2 = \sum_{j=1}^{6} j^2/6 - (\frac{7}{2})^2 = \frac{91}{6} - \frac{49}{4}$$

for each i. Since $X = \sum_{i=1}^{100} X_i$, we have

$$\mathbf{E}[X] = 100 \cdot \frac{7}{2} = 350.$$

and since X_i's are mutually independent, we have

$$\mathbf{Var}[X] = 100 \cdot \left(\frac{91}{6} - \frac{49}{4}\right) = \frac{875}{3}$$

Therefore, by using Chebyshev's inequality we have

$$\mathbf{Pr}[|X - 350| \geq 50] \leq \frac{875/3}{50^2} = \frac{7}{60} \approx 0.1167.$$

□

Exercise 3.4. *Prove that, for any real number c and any discrete random variable X, $\mathbf{Var}[cX] = c^2\mathbf{Var}[X]$.*

Solution.

$$\begin{aligned}
\mathbf{Var}[cX] &= \mathbf{E}[(cX)^2] - (\mathbf{E}[cX])^2 \\
&= \mathbf{E}[c^2X^2] - (c\mathbf{E}[X])^2 \text{ (by linearity of expectation)} \\
&= c^2\mathbf{E}[X^2] - c^2(\mathbf{E}[X])^2 \text{ (also by linearity of expectation)} \\
&= c^2(\mathbf{E}[X^2] - (\mathbf{E}[X])^2) \\
&= c^2\mathbf{Var}[X].
\end{aligned}$$

□

Exercise 3.8. *Suppose that we have an algorithm that takes as input a string of n bits. We are told that the expected running time is $O(n^2)$ if the input bits are chosen independently and uniformly at random. What can Markov's inequality tell us about the worst-case running time of this algorithm on inputs of size n?*

Solution. Let X be a random variable, which represents the running time of the algorithm. Since its expected running time is $O(n^2)$, we assume that $\mathbf{E}[X] \leq cn^2$ for some constant $c > 0$. Let $a \geq 2c$ be a constant. By Markov's inequality, the probability that the algorithm runs for an^2 time is

$$\mathbf{Pr}[X \geq an^2] \leq \frac{\mathbf{E}[X]}{an^2} \leq \frac{c}{a} \leq \frac{1}{2}.$$

If we choose k sufficiently and let $a \geq kc$, we will derive that the above probability is at most $1/k$, which can be very small. Therefore we know the worst-case running time of the algorithm will be still $O(n^2)$ with probability arbitrarily close to 1. Furthermore, if a is not $O(1)$, we can conclude that the algorithm has worst-case running time $\omega(n^2)$ with probability 0 when n approaches infinity (i.e., $n \to \infty$). □

Exercise 3.22. *Suppose that we flip a fair coin n times to obtain n random bits. Consider all $m = \binom{n}{2}$ pairs of these bits in some order. Let Y_i be the exclusive-or of the ith pair of bits, and let $Y = \sum_{i=1}^{m} Y_i$ be the number of Y_i that equal 1.*

(a) *Show that each Y_i is 0 with probability $1/2$ and 1 with probability $1/2$.*

(b) *Show that the Y_i are not mutually independent.*

(c) *Show that the Y_i satisfy the property that $\mathbf{E}[Y_iY_j] = \mathbf{E}[Y_i]\mathbf{E}[Y_j]$.*

(d) *Using Exercise 3.15, find $\mathbf{Var}[Y]$.*

(e) *Using Chebyshev's inequality, prove a bound on $\mathbf{Pr}[|Y - \mathbf{E}[Y]| \geq n]$.*

Solution.

(a) All possible ith pair of bits, say (b_{i_i}, b_{i_2}), are $(0,0), (0,1), (1,0)$, and $(1,1)$. The result of exclusive-or of b_{i_1} and b_{i_2}, which is denoted by $b_{i_1} \oplus b_{i_2}$, is 1 if $b_{i_1} \neq b_{i_2}$ and 0 otherwise. Thus we have $\mathbf{Pr}[Y_i = 0] = \mathbf{Pr}[Y_i = 1] = 1/2$.

(b) Let $b_1, b_2, \ldots, b_n$ be the n random bits. Let the symbol $\oplus$ denote the binary operator *exclusive-or*. With slight abuse of notation, let $Y_1 = b_1 \oplus b_2$, $Y_2 = b_2 \oplus b_3$, and $Y_3 = b_3 \oplus b_1$. From (1) we know that $\mathbf{Pr}[Y_1 = 1] = \mathbf{Pr}[Y_2 = 1] = \mathbf{Pr}[Y_3 = 1] = 1/2$. However, we can easily obtain that

$$\mathbf{Pr}[Y_1 = 1 \cap Y_2 = 1 \cap Y_3 = 1] = 0 \neq \mathbf{Pr}[Y_1 = 1] \cdot \mathbf{Pr}[Y_2 = 1] \cdot \mathbf{Pr}[Y_3 = 1],$$

since when $Y_1 = Y_2 = 1$, we have $b_1 = b_3$ so that Y_3 will never be 1. Therefore, the Y_i's are not mutually independent.

(c) Two pairs of bits Y_i and Y_j, which do not share any bit, are independent and hence we have $\mathbf{E}[Y_iY_j] = \mathbf{E}[Y_i]\mathbf{E}[Y_j]$ from Theorem 3.3 in Mitzenmacher and Upfal's textbook [1]. Consider the case that $Y_i = b_1 \oplus b_2$ and $Y_j = b_2 \oplus b_3$ (i.e., they share one bit, say b_2). By enumerating all possible outcomes of b_1, b_2, b_3, we can derive that $\mathbf{Pr}[Y_i = 1 \cap Y_j = 1] = 2/8 = 1/4$. Thus we have

$$\begin{aligned}\mathbf{E}[Y_iY_j] &= 1 \cdot \mathbf{Pr}[Y_i = 1 \cap Y_j = 1] \\ &= \frac{1}{4} \\ &= \mathbf{E}[Y_i]\mathbf{E}[Y_j].\end{aligned}$$

(d) Exercise 3.15 says that, *if* $\mathbf{E}[Y_iY_j] = \mathbf{E}[Y_i]\mathbf{E}[Y_j]$ *for every pair of i and j with*

$1 \le i < j \le m$, *then* $\mathbf{Var}[Y] = \sum_{i=1}^m \mathbf{Var}[Y_i]$. Hence by the result of (c) we have

$$\begin{aligned} \mathbf{Var}[Y] &= \sum_{i=1}^m \mathbf{Var}[Y_i] \\ &= \sum_{i=1}^m \mathbf{E}[Y_i^2] - (\mathbf{E}[Y_i])^2 \\ &= m \cdot \left(1 \cdot \frac{1}{2} - \left(1 \cdot \frac{1}{2}\right)^2\right) \\ &= \frac{\binom{n}{2}}{4}. \end{aligned}$$

(e) Since we have $\mathbf{Var}[Y] = \binom{n}{2}/4$ from the result of (d), we can derive

$$\begin{aligned} \mathbf{Pr}[|Y - \mathbf{E}[Y]| \ge n] &\le \frac{\mathbf{Var}[Y]}{n^2} \\ &= \frac{n(n-1)/8}{n^2} \\ &= \frac{1}{8} - \frac{1}{8n}. \end{aligned}$$

That is,

$$\mathbf{Pr}[|Y - \mathbf{E}[Y]| \ge n] \le \frac{1}{8} - \Omega(n^{-1}).$$

□

Exercise 3.24. *Generalize the median-finding algorithm to find the* k*th largest item in a set of* n *items for any given value of* k*. Prove that your resulting algorithm is correct and bound its running time.*

Solution. For simplicity, we give each element in S a minus weight so that the smallest k element will be the largest k element in the original S. The pseudo-code of the generalized algorithm is as follows.

Randomized kth largest Element Finding Algorithm

Input: A set S of n elements over a totally ordered universe.
Output: The kth largest element of S, denoted by K.

1. Pick a multiset R of $\lceil n^{3/4} \rceil$ elements in S, chosen independently and uniformly at random with replacement.
2. Sort the set R.
3. Let d be the $\left(\lfloor \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n} \rfloor\right)$th smallest element in the sorted set R.
4. Let u be the $\left(\lfloor \left(\frac{k}{n}\right) n^{3/4} + \sqrt{n} \rfloor\right)$th smallest element in the sorted set R.
5. By comparing every element in S to d and u, compute the set $C = \{x \in S \mid d \leq x \leq u\}$ and the numbers $l_d = |\{x \in S : x < d\}|$ and $l_u = |\{x \in S : x > u\}|$.
6. If $l_d > k$ or $l_u > n - k$ then FAIL.
7. If $|C| \leq 4n^{3/4}$ then sort the set C, otherwise FAIL.
8. Output the $(k - l_d + 1)$th element in the sorted order of C.

Theorem 1. *The randomized algorithm terminates in linear time, and if it does not output FAIL, then it outputs the correct kth largest element of the input set S.*

Proof: Correctness follows because the algorithm could only give an incorrect answer if the kth largest element were not found in the set C. But then either $l_d > k$ or $l_u > n-k$ and thus step 6 of the algorthm guarantees that, in these cases, the algorithm outputs FAIL. Similarly, as long as C is sufficiently small, the total work is only linear in the size of S. Step 7 of the algorithm therefore guarantees that the algorithm does not take more than linear time; if the sorting might take too long, the algorithm outputs FAIL without sorting.

■

Now we try to bound the error probability of the algorithm as follows. We identify "bad" events, as the textbook shows, such that if none of these bad events occurs, the algorithm does not fail. In a series of lemmas, we then bound the probability of each of these events and show that the sum of these probabilities is only $O(n^{-1/4})$.

Consider the following bad events:

$\mathcal{E}_1$: $Y_1 = |\{r \in R \mid r \leq m\}| < \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}$;

$\mathcal{E}_2$: $Y_2 = |\{r \in R \mid r \geq m\}| < \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}$;

$\mathcal{E}_3$: $|C| > 4n^{3/4}$.

Lemma 1. *The randomized algorithm fails if and only if at least one of $\mathcal{E}_1$, $\mathcal{E}_2$, or $\mathcal{E}_3$ occurs.*

Proof: Failure in step 7 of the algorithm is equivalent to the event $\mathcal{E}_3$. Failure in step 6 of the algorithm occurs if and only if $l_d > k$ or $l_u > n - k$. But for $l_d > k$, the $((k/n)n^{3/4} - \sqrt{n})$th smallest element of R must be larger than m; this is equivalent to the event $\mathcal{E}_1$. Similarly, $l_u > n - k$ is equivalent to the event $\mathcal{E}_2$.

■

Lemma 2.

$$\mathbf{Pr}[\mathcal{E}_1] \le \frac{1}{4} n^{-1/4}.$$

Proof: Define a random variable X_i such that $X_i = 1$ if the ith sample is less than or equal to K, i.e., the kth largest element of S, and 0 otherwise. The X_i's are independent, since the sampling is done with replacement. Because there are k elements in S that are less than or equal to K, the probability that a randomly chosen element of S is less than or equal to K can be written as

$$\mathbf{Pr}[X_i = 1] = \frac{k}{n}.$$

The event $\mathcal{E}_1$ is equivalent to

$$Y_1 = \sum_{i=1}^{n^{3/4}} X_i < \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}.$$

Since Y_1 is the sum of Bernoulli trials, it is a binomial random variable with parameters $n^{3/4}$ and k/n. Hence, using the result of Section 3.2.1 (i.e., the variance of $B(n, p)$ is $np(1-p)$) yields

$$\begin{aligned} \mathbf{Var}[Y_1] &= n^{3/4} \left(\frac{k}{n}\right) \left(1 - \frac{k}{n}\right) \\ &\le \frac{1}{4} n^{3/4}. \end{aligned}$$

The above inequality holds since $x(1-x) \le 1/4$ for any real number x. Applying Chebyshev's inequality then yields

$$\begin{aligned} \mathbf{Pr}[\mathcal{E}_1] &= \mathbf{Pr}[Y_1 < \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}] \\ &\le \mathbf{Pr}[|Y_1 - \mathbf{E}[Y_1]| > \sqrt{n}] \\ &\le \frac{\mathbf{Var}[Y_1]}{n} \\ &< \frac{1}{4} n^{-1/4}. \end{aligned}$$

■

Similarly we can obtain the same bound for the probability of the event $\mathcal{E}_2$. We now bound the probability of the third bad event $\mathcal{E}_3$.

Lemma 3.

$$\mathbf{Pr}[\mathcal{E}_3] \le \frac{1}{2} n^{-1/4}.$$

Proof: If $\mathcal{E}_3$ occurs, so $|C| > 4n^{3/4}$, then at least one of the following two events occurs:

$\mathcal{E}_{3.1}$: at least $2n^{3/4}$ elements of C are greater than K.

$\mathcal{E}_{3.2}$: at least $2n^{3/4}$ elements of C are smaller than K.

Let us bound the probability that the first event occurs; the second will have the same bound by symmetry. If there are at least $2n^{3/4}$ elements of C above K, then the order of u in the sorted order of S was at least $k+2n^{3/4}$ and thus the set R has at least $\left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}$ samples among the $k - 2n^{3/4}$ elements in S. Let $X = \sum_{i=1}^{n^{3/4}} X_i$, where $X_i = 1$ if the ith sample is among the $k - 2n^{3/4}$ largest elements in S, and 0 otherwise. Again, X is a binomial random variable, and we can derive

$$\mathbf{E}[X] = n^{3/4} \cdot \left(\frac{k}{n} - 2n^{-1/4}\right) = kn^{-1/4} - 2\sqrt{n}$$

and

$$\mathbf{Var}[X] = n^{3/4} \cdot (kn^{-1/4} - 2\sqrt{n})(1 - (kn^{-1/4} - 2\sqrt{n})) \leq \frac{1}{4}n^{3/4}.$$

Applying Chebyshev's inequality yields

$$\begin{aligned} \mathbf{Pr}[\mathcal{E}_{3.1}] &= \mathbf{Pr}[X \geq \left(\frac{k}{n}\right) n^{3/4} - \sqrt{n}] \\ &\leq \mathbf{Pr}[|X - \mathbf{E}[X]| \geq \sqrt{n}] \\ &\leq \frac{\mathbf{Var}[X]}{n} \\ &\leq \frac{1}{4}n^{-1/4}. \end{aligned}$$

Similarly,

$$\mathbf{Pr}[\mathcal{E}_{3.2}] \leq \frac{1}{4}n^{-1/4}$$

and

$$\mathbf{Pr}[\mathcal{E}_3] \leq \mathbf{Pr}[\mathcal{E}_{3,1}] + \mathbf{Pr}[\mathcal{E}_{3,2}] \leq \frac{1}{2}n^{-1/4}.$$

■

Combining the bounds we derive, we conclude that the probability that the algorithm outputs FAIL is bounded by

$$\mathbf{Pr}[\mathcal{E}_1] + \mathbf{Pr}[\mathcal{E}_2] + \mathbf{Pr}[\mathcal{E}_3] \leq n^{1/4}.$$

Thus we have proved the bound of the error probability of the algorithm. □

References

[1] M. Mitzenmacher and E. Upfal: *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.