Randomized Algorithms 2013A
Lecture 1 – Introduction and Concentration Bounds*

Robert Krauthgamer

1 Introduction

We shall consider algorithms that have access to randomnesss (equivalently, can toss coins) during the execution. Throughout, we shall we consider worst-case inputs, and analyze the algorithm's performance in expectation or with high probability.

2 Randomized Quicksort

Problem definition: The input is a list (array) of n integers, and the goal is to sort the array. We use the RAM model, but assume unit-time comparisons.

Quicksort algorithm: Pick a pivot, split the list into those smaller/bigger than pivot, sort each one recursively, and output in order.

Randomized version: the pivot is chosen uniformly at random from the list.

Theorem 1: The expected number of comparisons is at most $2nH_n$, where $H_n = \ln n + \Theta(1)$ is the nth harmonic number. Thus, the expectation of the running time is $O(n \log n)$.

Proof: seen in class, based on letting the random variable X_{ij} be an indicator for the event that the i-th and j-th smallest elements in the list are compared, and analyzing $\mathbb{E}[\sum_{i<j} X_{ij}]$. Analyzing $\mathbb{E}[X_{ij}]$ relies on the principle of deferred decision.

Exer: Let X, T be random variables taking real values, and suppose there is $a \in \mathbb{R}$ such that for all $t \in \mathbb{R}$,

$$\mathbb{E}[X \mid T = t] \leq a.$$

Prove that $\mathbb{E}[X] \leq a$. Note: This problem formalizes the principle of deferred decision used in class, e.g., T could be the "time" (iteration or tree-node) in which the value of X is determined.

*These notes summarize the material covered in class, usually skipping proofs, details, examples and so forth, and possibly adding some remarks, or pointers. The exercises are for self-practice and need not be handed in. In the interest of brevity, most references and credits were omitted.

Can we get a high-probability bound? Using Markov's inequality below, we see that with probability at least $1-\varepsilon$, the runtime is $O(\frac{1}{\varepsilon} n \log n)$. We will soon see a stronger bound by more direct computation.

Markov's inequality: Let X be a nonnegative random variable with finite expectation. Then

$$\forall t > 1, \quad \Pr[X \geq t \cdot \mathbb{E}] \leq 1/t.$$

3 Concentration bounds

Chernoff-Hoeffding bound: Let $X = \sum_{i \in [n]} X_i$ where $X_i \in [0,1]$ for $i \in [n]$ are independently distributed random variables. Then

$$\begin{aligned}
\forall t > 0, &\qquad \Pr[|X - \mathbb{E}[X]| \geq t] \leq 2e^{-2t^2/n}. \\
\forall 0 < \varepsilon \leq 1, &\qquad \Pr[X \leq (1-\varepsilon)\mathbb{E}[X]] \leq e^{-\varepsilon^2 \mathbb{E}[X]/2}. \\
\forall 0 < \varepsilon \leq 1, &\qquad \Pr[X \geq (1+\varepsilon)\mathbb{E}[X]] \leq e^{-\varepsilon^2 \mathbb{E}[X]/3}. \\
\forall t \geq 2e\mathbb{E}[X], &\qquad \Pr[X \geq t] \leq 2^{-t}.
\end{aligned}$$

Exer: Let X have a binomial distribution $B(n, 1/3)$. What is the probability that X deviates from its expectation additively by $r > 1$ standard deviations? Think of r being $10, \log n, \sqrt{n}$.

Exer: Let $a_1, \ldots, a_n$ be an array of numbers in the range $[0,1]$. Design a randomized algorithm that estimates their average within additive error $\pm\varepsilon$, by reading only $O(1/\varepsilon^2)$ elements. The algorithm should succeed with probability at least 90%.

Exer: Let $S_1, \ldots, S_n$ be subsets of $[n]$. Design an algorithm for 2-coloring the elements $[n]$, such that in every set S_i the balance, defined as $|\#\text{black} - \#\text{white}|$, is at most $O(\sqrt{n \log n})$.

Exer: Let $a_1, \ldots, a_n$ be again an array of numbers in the range $[0,1]$. Now design similarly a randomized algorithm that estimates their population variance $\frac{1}{n}\sum_i a_i^2 - (\frac{1}{n}\sum_i a_i)^2$. (Note: population variance refers to a set of n reals, while the usual word variance refers to a random variable.) Hint: estimate each of the two terms separately using the previous exercise.

4 High-probability bound for quicksort

Theorem: With probability at least $1 - 1/n$, the algorithm terminates in $O(n \log n)$ time.

Proof: seen in class, by considering a fixed element $a \in L$, and bounding the probability it will participate in more than $l = 21 \log n$ levels of the recursion. This analysis uses a Chernoff-Hoeffding bound from above. Finally, we apply a union bound over all $a \in L$.

Exer: Analyze the following algorithm, a variant of binary search, for finding a query element q in a sorted array A of size n, and show that with high probability it finishes in $O(\log n)$ steps.

Algorithm Randomized-Search: Starting with the interval $[l, h] = [1, n]$, repeatedly choose uniformly at random a pivot $p \in [l, h]$, compare q to $A[p]$ and update the interval to be either $[l, p-1]$ or $[p+1, h]$, stopping if $A[p] = q$ or $l > h$.

5 Balls and Bins (Occupancy Problems)

Problem definition: Suppose we throw balls independently and uniformly into n bins. We stop after $m = m(n)$ balls and examine the most/least loaded bin.

Expected behavior: Let X_i be the load of bin $i \in [n]$. The expected load of a fixed bin i is $\mathbb{E}[X_i] = m/n$.

But don't we expect deviations?

Empty bins for $m = n$ balls:

We expect one ball per bin. But how many bins will be empty?

$$\Pr[\text{bin } i \text{ is empty}] = \Pr[X_i = 0] = (1 - 1/n)^n \approx 1/e.$$

Therefore,

$$\mathbb{E}[\#\text{ of empty bins}] = \mathbb{E}[\sum_i I_{\{X_i=0\}}] = \sum_i \Pr[X_i = 0] \approx n/e.$$

Exer: How many bins are expected to have load 1?

Maximum load for $m = n$ balls:

$$\Pr[X_i \geq 2\log n] \leq 2^{-2\log n} = 1/n^2$$

and therefore

$$\Pr[\max_i X_i \geq 2\log n] \leq \sum_i \Pr[X_i \geq 2\log n] \leq 1/n.$$

We can actually do better by a direct computation:

$$\Pr[X_i = k] = \binom{n}{k}(1/n)^k(1-1/n)^{n-k} \leq (\tfrac{ne}{k})^k(1/n)^k = (\tfrac{e}{k})^k.$$

It follows that $\Pr[X_i \geq k] \leq O(1) \cdot (\frac{e}{k})^k$ by geometric series, and by plugging $k^* = e\ln n/\ln\ln n$, we have

$$\Pr[X_i \geq k^*] \leq 1/n^2,$$

and by a union bound,

$$\Pr[\max_i X_i \geq k^*] \leq 1/n.$$

What about expected maximum? Denote $X_{\max} = \max_i X_i$, then by the law of total expectation

$$\begin{aligned}\mathbb{E}[X_{\max}] &= \Pr[X_{\max} < k^*] \cdot \mathbb{E}[X_{\max} \mid X_{\max} < k^*] + \Pr[X_{\max} \geq k^*] \cdot \mathbb{E}[X_{\max} \geq k^*] \\ &\leq k^* + \tfrac{1}{n} \cdot n = k^* + 1.\end{aligned}$$

Exer: Suppose the expected load of a fixed bin is $m/n \geq 10 \log n$. Show that with high probability the maximum load among all bins is within factor 2 of the expected load. Show a similar bound for and the minimum bin load. For what value of m this ratio will be $1 \pm \varepsilon$ (assuming $\varepsilon < 1$)?