Homework 3

Out: 3 Feb. Due: 10 Feb.

Submit your solutions in pdf format on Gradescope by **5pm on Friday, February 3**. *Solutions may be written either in LaTeX (with either machine-drawn or hand-drawn diagrams) or* **legibly** *by hand. (The LaTeX source for this homework is provided in case you want to use it as a template.) Please be sure to begin the solution for each problem on a new page, and to tag each of your solutions to the correct problem! Per course policy, no late solutions will be accepted. Take time to write* **clear** *and* **concise** *answers; confused and long-winded solutions may be penalized. You are encouraged to form small groups (two to four people) to work through the homework, but you* **must** *write up all your solutions on your own. Depending on grading resources, we reserve the right to grade a random subset of the problems and check off the rest; so you are advised to attempt all the problems.*

1. Let $X_1, X_2, \dots, X_n, \dots$ be an infinite sequence of independent, identically distributed (i.i.d.) random variables. (For example, each of the X_i's might be the outcome of rolling some die once.) Suppose the X_i's have expectation μ and (finite) standard deviation σ. Use Chebyshev's inequality to prove that, for any fixed $\varepsilon > 0$,

$$\Pr\left[\left|\frac{X_1 + X_2 + \ldots + X_n}{n} - \mu\right| \geq \varepsilon\right] \;\rightarrow\; 0 \quad \text{as } n \rightarrow \infty.$$

[NOTE: This fact is referred to as the "Weak Law of Large Numbers," and captures what is commonly called the "Law of Averages."]

2. In the problem MAXCUT, we are given an undirected graph $G = (V, E)$ and asked to find a cut of *maximum* size in G. (Recall that a *cut* in G is a partition of the vertex set V into two parts; the *size* of the cut is the number of edges with one endpoint in each part of the partition.) In contrast to the seemingly very similar problem MINCUT discussed in class (Karger's algorithm), MAXCUT is a famous NP-hard problem, so we do not expect to find an efficient algorithm (even a randomized one) that solves it exactly. Here is a very simple linear-time randomized algorithm that gives a pretty good approximation:

- randomly and independently color each vertex $v \in V$ red or blue with probability $\frac{1}{2}$ each
- output the cut defined by the red/blue partition of vertices

(a) Let the r.v. X denote the size of the cut output by the algorithm. Compute $\mathrm{E}[X]$ as a function of the number of edges in G, and deduce that $\mathrm{E}[X] \geq \frac{\mathrm{OPT}}{2}$, where OPT is the size of a maximum cut in G. [HINT: Write X as a sum of indicators.]

(b) Let p denote the probability that the cut output by the algorithm has size at least $0.49\mathrm{OPT}$. Show that $p \geq 1/51$. [HINT: Applying Markov's inequality to X will not work here. Try applying Markov's inequality to a different r.v.]

(c) Now compute the variance $\mathrm{Var}[X]$. [HINT: Again write X as the sum of indicators, as in part (a).]

(d) Let p be the probability defined in part (b). Use Chebyshev's inequality together with part (c) to show that $p = 1 - O(1/|E|)$. [Note how Chebyshev's inequality gives us a much sharper bound here than Markov in part (b).]

(e) How would you modify the algorithm so that it *always* finds a cut of size at least $0.49\mathrm{OPT}$ but has only *expected* linear running time?

[Turn over for problems 3 & 4!]

3. Let π be a permutation of the numbers $\{1, \ldots, n\}$. An *inversion* in π is a pair (i, j) such that $i < j$ and $\pi(i) > \pi(j)$ (i.e., i and j are "in the wrong order" in π). Inversions are important in the study of permutations; for example, the number of inversions is exactly equal to the number of pairs of elements that are exchanged by any sorting algorithm that exchanges neighboring elements, such as BubbleSort. In this problem, we look at the number of inversions in a random permutation π.

(a) Let the r.v. X denote the number of inversions in a uniformly random permutation π. Prove that $\mathrm{E}[X] = \frac{1}{2}\binom{n}{2}$. [HINT: Write $X = \sum X_{ij}$ for suitable indicator r.v.'s X_{ij}.]

(b) Prove that $\mathrm{Var}[X] = \frac{n^3}{36} + O(n^2)$. [HINT: Most of the pairs X_{ij}, X_{kl} are independent, but not all of them. You will need to do a careful case analysis of the non-independent ones: for each case, compute the covariance and count the number of pairs that fall into this case.]

(c) Deduce from parts (a) and (b) and Chebyshev's inequality that the probability that π has more than $(\frac{1}{4} + \varepsilon)n^2$ inversions, for any constant $\varepsilon > 0$, tends to 0 as $n \to \infty$.

4. There were several questions in class about the choice of the parameters $n^{3/4}$ and $\sqrt{n}$ in the randomized median-finding algorithm (see MU section 3.5). In this question we explore the constraints on these parameters and see what other choices are possible.

Suppose we modify the algorithm so that the random subset R we pick has size n^α, and that we take d, u to be the elements of rank $\frac{1}{2}n^\alpha - n^\beta$ and $\frac{1}{2}n^\alpha + n^\beta$ in R, respectively[1]. Here $\alpha, \beta \in (0, 1)$ are parameters to be chosen. (In the version we discussed, $\alpha = 3/4$ and $\beta = 1/2$.)

(a) Let us now redefine the first two "bad" events (called $\mathcal{E}_d, \mathcal{E}_u$ in class, $\mathcal{E}_1, \mathcal{E}_2$ in MU) so that the bound on the right-hand side is $\frac{1}{2}n^\alpha - n^\beta$ instead of $\frac{1}{2}n^{3/4} - \sqrt{n}$. Follow through the same calculations using Chebyshev's inequality as we did in class to show that the probabilities of each of these two events are $O(n^{-2\beta+\alpha})$.

(b) Now redefine the third "bad" event ($\mathcal{E}_C$ in class, $\mathcal{E}_3$ in MU) to say that $|C|$ is larger than $4n^{1-\alpha+\beta}$. (Line 7 of the algorithm is also modified by replacing $4n^{3/4}$ by $4n^{1-\alpha+\beta}$.) By breaking this into two events, as we did in class, show again using Chebyshev's inequality as we did in class that it also has probability at most $O(n^{-2\beta+\alpha})$.

(c) Use parts (a) and (b) to deduce that, for any choice of $\alpha, \beta \in (0, 1)$ such that $\beta < \alpha < 2\beta$, the algorithm with parameters n^α, n^β as described above still runs in linear time and outputs the median of S with probability at least $1 - O(n^{-2\beta+\alpha})$ (which of course tends to 1 as $n \to \infty$). Be sure to justify the linear running time (which is not directly covered by parts (a), (b))!

---

[1] As in class, we don't worry about details like rounding to integers, or the ± 1 in the rank of the median, etc. You should also omit these details.