CS 170 HW 13

Due **2020-27-04, at 10:00 pm**

1 Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, you must explicitly write none.

2 Opting for releasing your solutions

We are considering releasing a subset of homework submissions written by students for students to see what a full score submission looks like. If your homework solutions are well written, we may consider releasing your solution. If you wish that your solutions not be released, please respond to this question with a "No, do not release any submission to any problems". Otherwise, say "Yes, you may release any of my submissions to any problems".

3 QuickSelect

Let $\boldsymbol{X}_{i,j}$ be an indicator random variable for the event that the i-th smallest number is ever compared with the j-th smallest in $\mathsf{QuickSelect}(A, k)$.

(a) Write an exact expression for $\mathbf{E}[\boldsymbol{X}_{i,j}]$.

Solution:

Since $\boldsymbol{X}_{i,j}$ is defined the way it is, we want to essentially determine what is the probability that the i-th smallest number is ever compared with the j-th smallest number.

We notice that where k is with respect to i and j actually affects our probability because in the algorithm of QuickSelect we only keep the side which contains k. What derives naturally from this observation is that i and j are only ever compared if they are chosen as the pivot, out of the range $[\min(k, i), \max(j, k)]$ as any other number chosen from that range would cause either i and j to be on a different side of the pivot as k or both i and j to be on the other side of the pivot compared to where k is. Notice how there are $\max(j, k) - \min(k, i) + 1$ elements in this range that could have been chosen as pivots.

Without loss of generality, let $i < j$ and assume the list is $1, 2, \ldots, n$. We will break up the indicator into 3 cases: $k \geq \{i, j\}$, $k \leq \{i, j\}$, and $i < k < j$.

If $k \geq i, j$, then $\max(j, k) = k$ so the probability is

$$\frac{2}{k - i + 1}.$$

If $k \leq i, j$, then $\min(i, k) = k$ so the probability is

$$\frac{2}{j - k + 1}.$$

If $i < k < j$, then $\max(j, k) = j$ and $\min(i, k) = i$ so the probability is

$$\frac{2}{j - i + 1}.$$

(b) Show that the expected runtime of $\mathsf{QuickSelect}(A, k)$ is $O(n)$.

Solution:

Our approach to computing the expected number of comparisons will be to upper bound

$$\mathbf{E}\left[\sum_{i<j} \boldsymbol{X}_{i,j}\right] = \sum_{i<j} \mathbf{E}\boldsymbol{X}_{i,j}.$$

Since our indicator is divided into three cases, let's also divide our expectation into three separate sums by linearity of expectations. We can simplify each case separately and then combine them together in the end.

Case 1 $k \geq i, j$:

The expectation of comparisons of pairs of (i, j) such that $k \geq j$ is:

$$\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \boldsymbol{X}_{i,j}$$

Simplifying we get:

$$\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \boldsymbol{X}_{i,j} = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{2}{k-i+1} = \sum_{i=1}^{k-1} \frac{2(k-i)}{k-i+1} \leq 2(k-1) \leq 2n$$

Case 2 $k \leq i, j$:

This is similar to case 1.

$$\sum_{j=k+1}^{n} \sum_{i=k}^{j-1}$$

Simplifying we get:

$$\sum_{j=k+1}^{n} \sum_{i=k}^{j-1} \boldsymbol{X}_{i,j} = \sum_{j=k+1}^{n} \sum_{i=k}^{j-1} \frac{2}{j-k+1} = \sum_{j=k+1}^{n} \frac{2(j-k)}{j-k+1} \leq 2(n-k) \leq 2n$$

Case 3 $i < k < j$:

We will break this case up into length, specifically $d := j - i$. We notice that there are at most $j - i - 1$ pairs of (i, j) pairs which has the same distance. Thus we get the summation Thus our expectation becomes

$$\sum_{d=2}^{n} \sum_{i=\max(1,k+1-d)}^{\max(k-1,1)} \boldsymbol{X}_{i,j}$$

Simplifying we get:

$$\sum_{d=2}^{n} \sum_{i=\max(1,k+1-d)}^{\max(k-1,1)} \boldsymbol{X}_{i,j} = \sum_{d=2}^{n} \sum_{i=\max(1,k+1-d)}^{\max(k-1,1)} \frac{2}{j-i+1} \leq \sum_{d=2}^{n} \frac{2(d-1)}{d+1} \leq 2n$$

Since all three summations are upperbounded by $2n$, we getour overall runtime to be upperbounded by $6n$ thus proving that $\mathsf{QuickSelect}(A, k)$ is $O(n)$.

4 $\sqrt{n}$ alloys

(a) Let G be a graph of maximum degree Δ. Show that G is $(\Delta + 1)$-colorable. **Solution:** Let the color palette be $[\Delta + 1]$. While there is a vertex with an unassigned color, give it a color that is distinct from the color assigned to any of its neighbors (such a color always exists since we have a palette of size $\Delta + 1$ but only at most Δ neighbors).

(b) Suppose G is a 3-colorable graph. Let v be any vertex in G. Show that the graph induced on the neighborhood of v is 2-colorable. *Clarification: the graph induced on the neighborhood of v refers to the subgraph of G obtained from the vertex set V' comprising vertices adjacent to v and edge set comprising all edges of G with both endpoints in V'.*
Solution:

In an induced graph, we only care about the direct vertices adjacent to v and v itself. In any 3-coloring of G, v must have a different color from its neighborhood and therefore the neighborhood must be 2-colorable.

(c) Give a polynomial time algorithm that takes in a 3-colorable graph G as input and outputs a valid coloring of its vertices using $O(\sqrt{n})$ colors. Prove that your algorithm is correct and also analyze its runtime.
Hint: think of an algorithm that first colors "high-degree" vertices and their neighborhoods, and then colors the rest of the graph. The previous two parts might be useful.
Solution:

While there is a vertex of degree-$\geq \sqrt{n}$, choose the vertex v, pick 3 colors, use one color to color v, and the remaining 2 colors to color the neighborhood of v (2-coloring is an easy problem). Never use these colors ever again and delete v and its neighborhood from the graph. Since each step in the while loop deletes at least $\sqrt{n}$ vertices, there can be at most $\sqrt{n}$ iterations. This uses only $3(\sqrt{n} + 1)$ colors. After this while loop is done we will be left with a graph with max degree at most $\sqrt{n}$. We can $\sqrt{n} + 1$ color with fresh colors this using the greedy strategy from the solution to part b. The total number of colors used is $O(\sqrt{n})$.

5 Cuts from electric charges

Given a graph $G = (V, E)$ on n vertices and m edges, and a vector $x \in \{\pm 1\}^n$, we say

$$\mathsf{Cut}(G, x) := \frac{1}{m} \sum_{\{i,j\} \in E} \left(\frac{x_i - x_j}{2} \right)^2$$

and define

$$\mathsf{MaxCut}(G) := \max_{x \in \{\pm 1\}^n} \mathsf{Cut}(G, x).$$

For every algorithmic question below, please analyze your runtime and prove that your algorithm is correct.

(a) (Warmup; ungraded) Let G be any graph. Prove that

$$\mathsf{MaxCut}(G) \geq \frac{1}{2}$$

always, and give a polynomial time randomized algorithm that outputs $\boldsymbol{x} \in \{\pm 1\}^n$ satisfying $\mathbf{E}[\mathsf{Cut}(G, \boldsymbol{x})] \geq \frac{1}{2}$.
What if each x_i is chosen to be $+1$ or -1 uniformly independently? **Solution:** Let $\boldsymbol{x} \sim \{\pm 1\}^n$ and apply linearity of expectation.

(b) Let G be any graph. Prove that

$$\mathsf{MaxCut}(G) \geq \frac{1}{2} + \Omega\left(\frac{1}{n}\right)$$

always, and give a polynomial time randomized algorithm that outputs $\boldsymbol{x} \in \{\pm 1\}^n$ satisfying $\mathbf{E}[\mathsf{Cut}(G, \boldsymbol{x})] \geq \frac{1}{2} + \Omega\left(\frac{1}{n}\right)$.
Hint: try to construct a simple distribution over ± 1 vectors such that for $\boldsymbol{x}$ drawn from this distribution any $i, j \in [n]$, $\mathbf{E}[\boldsymbol{x}_i \boldsymbol{x}_j] = -\frac{c}{n}$ for some absolute constant $c > 0$.
Solution: The idea in this problem is to sample a random balanced cut. Let $\boldsymbol{x} \sim \{y : y \in \{\pm 1\}^n, \sum_{i=1}^n y_i = 0\}$ (restrict the sum to be 1 if n is an odd number) and apply linearity of expectation.

(c) Let G be any k-colorable graph. Prove that

$$\mathsf{MaxCut}(G) \geq \frac{1}{2} + \Omega\left(\frac{1}{k}\right).$$

Note that we are not asking you to give an algorithm to find such a cut, but instead just asking you to prove existence. Try to reduce this problem to the previous part. **Solution:** Consider the graph G_k obtained by collapsing each color class to a super-vertex and placing an edge of weight equal to the total weight of edges between corresponding color classes. Apply the same algorithm from question 4 to G_k, and assign every vertex in a color class the same sign as its super-vertex; any edge in G_k is cut with probability $\frac{1}{2} + \Omega\left(\frac{1}{k}\right)$.

(d) Let G be any 3-colorable graph. Give a polynomial time randomized algorithm to find a $\boldsymbol{x} \in \{\pm 1\}^n$ satisfying:

$$\mathbf{E}[\mathsf{Cut}(G, \boldsymbol{x})] \geq \frac{1}{2} + \Omega\left(\frac{1}{\sqrt{n}}\right).$$

Hint: Part (b) of Question 3 may be useful. **Solution:** Use the algorithm from question 4 c to $O(\sqrt{n})$ color the graph and apply the solution from the previous part.

(e) Let G be any graph with maximum degree Δ. Prove that

$$\mathsf{MaxCut}(G) \geq \frac{1}{2} + \Omega\left(\frac{1}{\Delta}\right).$$

Give a polynomial time randomized algorithm to find a $\boldsymbol{x} \in \{\pm 1\}^n$ satisfying:

$$\mathbf{E}[\mathsf{Cut}(G, \boldsymbol{x})] \geq \frac{1}{2} + \Omega\left(\frac{1}{\Delta}\right).$$

Hint: Part (a) of Question 3 might be useful here. **Solution:** Use the greedy algorithm from 4 a to $\Delta + 1$ color G and then apply the solution given in 4c.

6 Fixing Blemishes

We use $G(n,p)$ to denote the distribution of graphs obtained by taking n vertices and for each pair of vertices i,j placing edge $\{i,j\}$ independently with probability p.

(a) Compute the expected number of edges in $G(n,p)$? **Solution:** By linearity of expectation, $p\binom{n}{2}$ since there are $\binom{n}{2}$ potential edges in the graph.

(b) Compute the expected number of 4-cycles in $G(n,p)$? **Solution:** There are 3 possible 4-cycles on a given set of 4 vertices so by linearity of expectation, the answer is $p^4 \cdot \frac{n(n-1)(n-2)(n-3)}{8}$.

(c) Give a polynomial time randomized algorithm that takes in n as input and in poly(n)-time outputs a graph G such that G has no 4-cycles and the expected number of edges in G is $\Omega(n^{4/3})$. **Solution:** Let $\boldsymbol{G} \sim G(n,p)$ for $p = Cn^{-2/3}$; let $\boldsymbol{e}$ be the number of edges and let $\boldsymbol{c}_4$ be the number of 4-cycles in $\boldsymbol{G}$. Let $\boldsymbol{G}'$ be the graph obtained by deleting an edge from every 4-cycle, and output it; this deletion removes at most $\boldsymbol{c}_4$ edges, so the number of edges $\boldsymbol{e}'$ remaining in the graph satisfies: $\boldsymbol{e}' \geq \boldsymbol{e} - \boldsymbol{c}_4$, which means $\mathbf{E}[\boldsymbol{e}'] \geq \mathbf{E}[\boldsymbol{e}] - \mathbf{E}[\boldsymbol{c}_4] = \frac{pn(n-1)}{2} - \frac{p^4 n(n-1)(n-2)(n-3)}{4} \geq \frac{pn^2 - p^4 n^4}{4} = \frac{Cn^{4/3} - C^4 n^{4/3}}{4}$. Choosing C as, say, .5 finishes the proof, and each step takes polynomial time.

7 Porcupine Trio

Let L be a vector of integers in $[-M,M]^n$ given to us as input. For any other vector x, we will use $\langle L, x\rangle$ to denote $\sum_{i=1}^n L_i x_i$. In this problem we will assume $M < 2^n/(4n)$.

(a) Prove that there exists distinct $x_1, x_2 \in \{\pm 1\}^n$ such that $\langle L, x_1\rangle = \langle L, x_2\rangle$.
Hint: try to prove this using the pigeonhole principle. **Solution:** $\langle L, x\rangle$ is always in $[-Mn, Mn]$ and hence with our bound on M there are strictly less than 2^n possibilities for $\langle L, x\rangle$. Since there are exactly 2^n possibilities for x, by the pigeonhole principle there must exist distinct x_1, x_2 such that $\langle L, x_1\rangle = \langle L, x_2\rangle$.

(b) Let $\boldsymbol{x}$ be sampled uniformly at random from $\{\pm 1\}^n$. Use Chebyshev's inequality to prove that:

$$\Pr[|\langle L, \boldsymbol{x}\rangle| > 10M\sqrt{n}] \leq \frac{1}{100}.$$

Solution: $\mathbf{E}[\langle L, \boldsymbol{x}\rangle^2] = \sum_{i=1}^n L_i^2 \leq M^2 n$. Since $\mathbf{E}[\langle L, \boldsymbol{x}\rangle] = 0$, we know $\sqrt{\mathbf{Var}[\langle L, \boldsymbol{x}\rangle]} \leq M\sqrt{n}$. The desired inequality is then a direct consequence of this chatter combined with Chebyshev's inequality.

(c) Let $\boldsymbol{Y}$ be a random variable that is equal to 1 with probability $\frac{1}{k}$ and 0 otherwise. Let $Y_1, \ldots, Y_{2k\ln r}$ be $2k\ln r$ independent copies of $\boldsymbol{Y}$. Prove:

$$\Pr[\boldsymbol{Y}_1 + \cdots + \boldsymbol{Y}_{2k\ln r} < 2] \leq \frac{2}{r}.$$

Hint: you may use the fact that $\left(1 - \frac{1}{k}\right)^k \leq \frac{1}{e}$ *without proof.* **Solution:** $\Pr[\boldsymbol{Y}_1 +$

$\cdots + \boldsymbol{Y}_{k\ln r} = 0] = \left(1-\frac{1}{k}\right)^{k\ln r} \leq \frac{1}{r}$. Identically, $\Pr[\boldsymbol{Y}_{k\ln r+1} + \cdots + \boldsymbol{Y}_{2k\ln r} = 0] = \left(1-\frac{1}{k}\right)^{k\ln r} \leq \frac{1}{r}$. A union bound on the two tells us that the probability that "at least one of the sums is 0 is at most $\frac{2}{r}$". The event in the quotations "" is a superset of the event whose probability we wish to bound as if the total sum is ¡ 2 at least one of the halves must sum to 0, which complete the proof.

(d) Give a randomized algorithm that runs in time $O(Mn^{3/2}\log(n))$ and with probability $1 - o_n(1)$ outputs distinct $x_1, x_2 \in \{\pm 1\}^n$ such that $\langle L, x_1\rangle = \langle L, x_2\rangle$. **Solution:** For $\boldsymbol{x} \sim \{\pm 1\}^n$, the most likely outcome P for $\langle L, \boldsymbol{x}\rangle$ in $[-10M\sqrt{n}, 10M\sqrt{n}]$ has probability at least $\frac{1}{\ell} := \frac{.99}{20M\sqrt{n}+1}$ from part b. If we sample $\lceil 2\ell \ln n\rceil$ independent $\boldsymbol{x}$, by part c at least two of the samples $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ satisfy $\langle L, \boldsymbol{x}_1\rangle = \langle L, \boldsymbol{x}_2\rangle = P$ with probability at least $1-\frac{1}{n}$. Now conditioned on this $1-\frac{1}{n}$ probability event, we turn our attention to lower bounding the probability that $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ (the first two samples satisfying $\langle L, \boldsymbol{x}_1\rangle = \langle L, \boldsymbol{x}_2\rangle = P$ are *distinct*. The number of different x such that $\langle L, x\rangle = P$ is at least $\frac{2^n}{\ell} \geq C\sqrt{n}$ for some constant C. Thus, the probability that $\boldsymbol{x}_2 \neq \boldsymbol{x}_1$ is at least $1 - \frac{1}{C\sqrt{n}}$. To summarize, with probability at least $\left(1-\frac{1}{n}\right)\cdot\left(1-\frac{C}{\sqrt{n}}\right)$ there will be two distinct samples $\boldsymbol{x}_1, \boldsymbol{x}_2$ among $\lceil 2\ell\ln n\rceil$ independent samples such that $\langle L, \boldsymbol{x}_1\rangle = \langle L, \boldsymbol{x}_2\rangle = P$.

This suggests the following algorithm: sample $\lceil 2\ell \ln n\rceil$ independent $\boldsymbol{x} \sim \{\pm 1\}^n$, compute each $\langle L, \boldsymbol{x}\rangle$, store pairs $(\boldsymbol{x}, \langle L, \boldsymbol{x}\rangle)$ in a list, and then sort the list by $\langle L, \boldsymbol{x}\rangle$ values. Iterate through the list, and output the first adjacent pair $\boldsymbol{x}_i, \boldsymbol{x}_{i+1}$ such that $\langle L, \boldsymbol{x}_i\rangle = \langle L, \boldsymbol{x}_{i+1}\rangle$ and $\boldsymbol{x}_i \neq \boldsymbol{x}_{i+1}$.

(e) (Fun bonus question worth no points) Can you improve the runtime in the previous part to $O(\sqrt{M}n^{5/4}\log n)$?