CS 6550: Randomized Algorithms April 5, 2019

Problem Set 4

Instructor: Eric Vigoda *Caitlin Beecham (worked with Qiaomei Li)*

Problem 1 FPRAS

For any $0 < \varepsilon < 1$ and $0 < \delta < 1$, we say a randomized algorithm for a problem outputs an (ε, δ)-approximation if

$$\Pr\left((1-\varepsilon)N \leq \text{Output} \leq (1+\varepsilon)N\right) \geq 1-\delta$$

where N is the actual solution the algorithm wants to compute. Show that the following alternative definition is equivalent to the definition of an FPRAS given in class: A *fully polynomial randomized approximation scheme (FPRAS)* for a problem is a randomized algorithm for which, given an input x and any parameter ε with $0 < \varepsilon < 1$, the algorithm outputs an $(\varepsilon, 1/4)$-approximation in time that is polynomial in $1/\varepsilon$ and the size of the input x.

Namely, we wish to show that the following two definitions are equivalent.

Definition 1:

$$Pr((1-\epsilon)N \leq OUTPUT \leq (1+\epsilon)N) \geq 1-\delta$$

and the running time is polynomial in $n =$ size of the input x, $\frac{1}{\epsilon}$ and $log(\frac{1}{\delta})$.

Defintion 2:

$$Pr((1-\epsilon)N \leq OUTPUT \leq (1+\epsilon)N) \geq \frac{3}{4}$$

and the running time is polynomial in $n =$ size of the input x and $\frac{1}{\epsilon}$.

Clearly, if I have an algorithm satisfying Definition 1, it also satisfies Definition 2 by plugging in $\delta = \frac{1}{4}$.

Now, say I have an algorithm satisfying Definition 2, I also wish to show that some modification of it satisfies Definition 1. What I wish to do is run this algorithm many times and take the median of its outputs. Say I run it $2A+1$ times, then take the median of its outputs. In order for this median to fall outside the desired range, at least $A+1$ of these outputs must have fallen below $(1-\epsilon)N$ or at least $A+1$ outputs must have fallen above $(1+\epsilon)N$. Say we have outputs $\{Q_1, Q_2, \ldots, Q_{2A+1}\}$. What is the probability

$$Pr(|\{i|Q_i \leq (1-\epsilon)N\}| \geq A+1 \text{ OR } |\{i|Q_i \geq (1+\epsilon)N\}| \geq A+1)?$$

Well, first note that Definition 2 implies that

$$Pr\Big(\big((1-\epsilon)N \geq OUTPUT\big) \text{ OR } \big((1+\epsilon)N \leq OUTPUT\big)\Big) \leq \frac{1}{4}.$$

Then, also

$$Pr\big((1-\epsilon)N \geq OUTPUT\big) \leq \frac{1}{4}, \text{ AND}$$
$$Pr\big((1+\epsilon)N \leq OUTPUT\big) \leq \frac{1}{4}.$$

Now, we calculate

$$Pr\Big(|\{i|Q_i \leq (1-\epsilon)N\}| \geq A+1\Big) \leq \frac{1}{4}^{A+1} \text{ AND}$$
$$Pr\Big(|\{i|Q_i \geq (1+\epsilon)N\}| \geq A+1\Big) \leq \frac{1}{4}^{A+1},$$

which then by the union bound implies that

$$\begin{aligned} Pr(|\{i|Q_i \leq (1-\epsilon)N\}| \geq A+1 \text{ OR } |\{i|Q_i \geq (1+\epsilon)N\}| \geq A+1) &\leq 2\frac{1}{4}^{A+1} \\ &= \frac{2}{4^{A+1}} \\ &= \frac{2}{(2^2)^{A+1}} \\ &= \frac{2}{2^{2A+2}} \\ &= \frac{1}{2^{2A+1}}. \end{aligned}$$

Now, one desires

$$Pr(|\{i|Q_i \leq (1-\epsilon)N\}| \geq A+1 \text{ OR } |\{i|Q_i \geq (1+\epsilon)N\}| \geq A+1) \leq \delta.$$

It would suffice to choose A such that

$$\frac{1}{2^{2A+1}} = 2^{-2A-1} \leq \delta.$$

The following are equivalent

$$\begin{aligned} 2^{-2A-1} &\leq \delta \\ (-2A-1)ln(2) &\leq ln(\delta) \\ (-2A-1) &\leq \frac{ln(\delta)}{ln(2)} \\ -2A &\leq \frac{ln(\delta)}{ln(2)} + 1 \\ A &\geq \frac{-1}{2}\frac{ln(\delta)}{ln(2)} - \frac{1}{2} \\ &= \frac{-1}{2}log_2(\delta) - \frac{1}{2} \\ &= \frac{-1}{2}(log_2(\delta) - log_2(2)) \\ &= \frac{1}{2}(log_2(2) - log_2(\delta)) \\ &= \frac{1}{2}(log_2(\frac{2}{\delta})) \\ &= \frac{1}{2}(log_2(2\frac{1}{\delta})) \\ &= \frac{1}{2}(log_2(2) + log_2(\frac{1}{\delta})) \\ &= \frac{1}{2}(1 + log_2(\frac{1}{\delta})) \\ &= \frac{1}{2} + \frac{1}{2}log_2(\frac{1}{\delta})). \end{aligned}$$

So, if we run the FPRAS algorithm A_2 which satisfies Definition 2 at least $2\hat{A}+1$ times where

$$\hat{A} \geq \lceil \frac{1}{2} log_2(\frac{1}{\delta})) \rceil,$$

then such an algorithm called A_1 which consists of running the A_2 algorithm $2\hat{A}+1$ times and then taking the median of all the outputs will succeed with probability (note I am assuming $\delta < 0.5$ in all this which makes sense as no one would want error probability greater than 0.5) at least $1-\delta$. Now, is the algorithm A_1 polynomial in n, $\frac{1}{\epsilon}$ and $log_2(\frac{1}{\delta})$? Yes, A_2 is polynomial in n and $\frac{1}{\epsilon}$. So, say the runtime of A_2 is $T_2(n, \frac{1}{\epsilon})$ which is a polynomial. Now, the runtime of A_1 is $O((2\hat{A}+1)T_2(n, \frac{1}{\epsilon})) = O((2\lceil \frac{1}{2} log_2(\frac{1}{\delta})) \rceil + 1)T_2(n, \frac{1}{\epsilon}))$ which is polynomial in n, $\frac{1}{\epsilon}$, and $log(\frac{1}{\delta})$.

Problem 2 #DNF

(a) Let $S_1, S_2, \ldots, S_m$ be subsets of a finite universe U. We know $|S_i|$ for $1 \leq i \leq m$. We wish to obtain an (ε, δ)-approximation to the size of the set

$$S = \bigcup_{i=1}^{m} S_i.$$

We have available a procedure that can, in one step, choose an element uniformly at random from a set S_i. Also, given an element $x \in U$, we can determine the number of sets S_i for which $x \in S_i$. We call this number $c(x)$.

Define p_i to be

$$p_i = \frac{|S_i|}{\sum_{j=1}^{m} |S_j|}.$$

The j'th trial consists of the following steps. We choose a set S_j, where the probability of each set S_i being chosen is p_i, and then we choose an element x_j uniformly at random from S_j. In each trial the random choices are independent of all other trials. After t trails, we estimate $|S|$ by

$$\left(\frac{1}{t} \sum_{j=1}^{t} \frac{1}{c(x_j)} \right) \left(\sum_{i=1}^{m} |S_i| \right).$$

Determine - as a function of m, ε and δ - the number of trials needed to obtain an (ε, δ)-approximation to $|S|$. The following theorem might be helpful:

Theorem (Hoeffding's inequality)**.** *Let $Z_1, \ldots, Z_n$ be independent and identically distributed random variables over $[0, 1]$, and define $Z = \sum_{i=1}^{n} Z_i$. Then for all $\delta \geq 0$,*

$$\Pr\left(|Z - \mathbb{E}[Z]| \geq \delta\right) \leq 2\exp(-2\delta^2/n).$$

We define

$$Z_k := \frac{1}{c(x_k)},$$

$$Z := \sum_{k=1}^{t} Z_k.$$

Then, one asks what is $E[Z]$? Well, it is

$$\begin{aligned}
E[Z] &= \sum_{k=1}^{t} E[Z_k] \\
&= \sum_{k=1}^{t} E[\frac{1}{c(x_j)}] \\
&= \sum_{k=1}^{t} \sum_{i=1}^{m} \sum_{j=1}^{|S_i|} \frac{|S_i|}{\sum_{l=1}^{m} |S_l|} \frac{1}{|S_i|} \frac{1}{c(x_j^i)} \text{ (where } x_j^i \text{ is the jth element of the ith subset)} \\
&= \sum_{k=1}^{t} \sum_{i=1}^{m} \sum_{j=1}^{|S_i|} \frac{1}{\sum_{l=1}^{m} |S_l|} \frac{1}{c(x_j^i)} \\
&= (\frac{1}{\sum_{l=1}^{m} |S_l|}) \left(\sum_{k=1}^{t} \left(\sum_{i=1}^{m} \sum_{j=1}^{|S_i|} \frac{1}{c(x_j^i)} \right) \right) \\
&= (\frac{1}{\sum_{l=1}^{m} |S_l|}) \left(\sum_{k=1}^{t} \left(\sum_{x \in S} c(x) \frac{1}{c(x)} \right) \right) \\
&= (\frac{1}{\sum_{l=1}^{m} |S_l|}) \left(\sum_{k=1}^{t} \left(\sum_{x \in S} 1 \right) \right) \\
&= (\frac{1}{\sum_{l=1}^{m} |S_l|}) \left(\sum_{k=1}^{t} |S| \right) \\
&= \sum_{k=1}^{t} \frac{|S|}{\sum_{l=1}^{m} |S_l|} \\
&= \frac{t|S|}{\sum_{l=1}^{m} |S_l|}.
\end{aligned}$$

Now, we wish to show that with probability $\geq 1 - \delta$ one has that

$$(1-\epsilon)|S| \leq \left(\frac{1}{t} \sum_{j=1}^{t} \frac{1}{c(x_j)} \right) \left(\sum_{i=1}^{m} |S_i| \right) \leq (1+\epsilon)|S|.$$

Now, recall that

$$|S| = \left(\frac{1}{t} \sum_{i=1}^{m} |S_i| \right) E[Z].$$

So,

$$\begin{aligned}
(1-\epsilon)|S| &= \left(\frac{1}{t} \sum_{i=1}^{m} |S_i| \right) \left((1-\epsilon) E[Z] \right), \\
(1+\epsilon)|S| &= \left(\frac{1}{t} \sum_{i=1}^{m} |S_i| \right) \left((1+\epsilon) E[Z] \right).
\end{aligned}$$

Now, note that substituting in these expressions

$$(1-\epsilon)|S| \leq \left(\frac{1}{t} \sum_{j=1}^{t} \frac{1}{c(x_j)} \right) \left(\sum_{i=1}^{m} |S_i| \right) \leq (1+\epsilon)|S|$$

becomes

$$\left(\frac{1}{t}\sum_{i=1}^{m}|S_i|\right)\Big((1-\epsilon)E[Z]\Big) \leq \left(\frac{1}{t}\sum_{j=1}^{t}\frac{1}{c(x_j)}\right)\left(\sum_{i=1}^{m}|S_i|\right) \leq \left(\frac{1}{t}\sum_{i=1}^{m}|S_i|\right)\Big((1+\epsilon)E[Z]\Big),$$

which holds if and only if

$$\Big((1-\epsilon)E[Z]\Big) \leq \left(\sum_{j=1}^{t}\frac{1}{c(x_j)}\right) \leq \Big((1+\epsilon)E[Z]\Big).$$

Finally, recalling $Z = \sum_{j=1}^{t}\frac{1}{c(x_j)}$, one sees that the above is equivalent to

$$E[Z] - \epsilon E[Z] \leq Z \leq E[Z] + \epsilon E[Z],$$

which happens exactly when

$$|Z - E[Z]| \leq \epsilon E[Z].$$

We first note that $Z_k \in [0,1]$ because $c(x_k) \geq 1$. Then, $Z_k = \frac{1}{c(x_k)} \leq 1$ and also, since $c(x_k) \geq 0$, $\frac{1}{c(x_k)} \geq 0$. Additionally, the trials are independent and each chosen from the same distribution (namely using the same Monte Carlo process). So, the $\{Z_1, \ldots, Z_t\}$ are IID variables in $[0,1]$, which means that we can apply the Hoeffding bound.

Now, one applies Hoeffding's equality to obtain that

$$\Pr\left(|Z - \mathbb{E}[Z]| \geq \epsilon E[Z]\right) \leq 2\exp(-2(\epsilon E[Z])^2/t).$$

One desires this probability to be bounded above by δ, thus one writes

$$\Pr\left(|Z - \mathbb{E}[Z]| \geq \epsilon E[Z]\right) \leq 2\exp(-2(\epsilon E[Z])^2/t) \leq \delta$$

and solves for the appropriate t. Then, recall that $E[Z] = \frac{t|S|}{\sum_{i=1}^{m}|S_i|}$. So, equivalently, we know

$$\begin{aligned}
\Pr\left(|Z - \mathbb{E}[Z]| \geq \epsilon E[Z]\right) &\leq 2\exp(-2(\epsilon\frac{t|S|}{\sum_{i=1}^{m}|S_i|})^2/t) \\
&= 2\exp(-2\epsilon^2\frac{t^2|S|^2}{(\sum_{i=1}^{m}|S_i|)^2}/t) \\
&= 2\exp(-2\epsilon^2\frac{t|S|^2}{(\sum_{i=1}^{m}|S_i|)^2})
\end{aligned}$$

Now, the following are equivalent.

$$\begin{aligned}
2\exp(-2(\epsilon E[Z])^2/t) &\leq \delta \\
\exp(-2\epsilon^2\frac{t|S|^2}{(\sum_{i=1}^{m}|S_i|)^2}) &\leq \frac{\delta}{2} \\
\exp((t)(-2\epsilon^2\frac{|S|^2}{(\sum_{i=1}^{m}|S_i|)^2})) &\leq \frac{\delta}{2} \\
e^t &\geq (\frac{\delta}{2})^{\left(\frac{(\sum_{i=1}^{m}|S_i|)^2}{-2\epsilon^2|S|^2}\right)} \\
t &\geq ln(\frac{\delta}{2})^{\left(\frac{(\sum_{i=1}^{m}|S_i|)^2}{-2\epsilon^2|S|^2}\right)}.
\end{aligned}$$

So taking

$$\hat{t} = \left\lceil ln(\frac{\delta}{2})^{\left(\frac{(\sum_{i=1}^{m}|S_i|)^2}{-2\epsilon^2|S|^2}\right)} \right\rceil + 1$$

trials suffices to obtain an (ϵ, δ) approximation of $|S|$.

(b) Explain how to use your results from part (a) to obtain an alternative approximation algorithm for counting the number of solutions to a DNF formula.

Let $S_i = \{$ set of assignments satisfying clause i$\}$. Then, the set of assignments satisfying at least one clause is $S = \cup_{i=1}^{m} S_i$. We wish to approximate $|S|$ which is the number of satisfying assignments. We do so by choosing clause i with probability $\frac{|S_i|}{\sum_{i=1}^{m}|S_i|}$. (Note that $|S_i|$ is easily computed as 2^{N-l_i} where N is the number of variables and l_i is the length of clause i). Now, once we have chosen a clause i, pick uniformly some element $a \in S_i$, meaning some assignment that satisfies clause i. Then, see how many clauses this assignment satisfies, call it $c(a)$. Then, one notes that similarly to part (a), one can output

$$\left(\frac{1}{t}\sum_{j=1}^{t}\frac{1}{c(x_j)}\right)\left(\sum_{i=1}^{m}|S_i|\right).$$

as an estimate of $|S|$ and all the same analysis follows through as this is just an application of part (a) where we define S_i, S, c as stated above.

Problem 3 Select a paper for your final project

Please write down here which paper you would like to read and write a report for. Feel free to choose a paper vaguely related to the course topics and your interests. Some specific suggestions are here

`https://www.cc.gatech.edu/~vigoda/6550/project.html`,

but you are welcome to ignore and select a paper of your choice.