

# CS 6150: HW2

Due Date: Oct 1

This assignment has 7 questions, for a total of 100 points. Unless otherwise specified, complete and reasoned arguments will be expected for all answers.

Question	Points	Score
Tail Bounds I	20	
Tail Bounds II	15	
Medians	10	
Amplification	10	
The “median” trick	15	
Hashing with open addressing	15	
Derandomizing MAX CUT	15	
Total:	100	

Here are the three increasingly-strong inequalities we use to analyze the tail of a distribution.

**Lemma 0.1** (Markov). *Let  $X$  be a random variable taking nonnegative values. Then for any  $a > 0$ ,*

$$\Pr(X \geq a) \leq \frac{E[X]}{a}$$

**Lemma 0.2** (Chebyshev). *Let  $X$  be a random variable with bounded variance. Then*

$$\Pr(|X - E[X]| \geq a) \leq \frac{\text{Var}[X]}{a^2}$$

**Lemma 0.3** (Chernoff). *Let  $X_1, \dots, X_n$  be independent random variables taking the values 0, 1 with  $E[X_i] = \Pr(X_i = 1) = p_i$ . Let  $\mu = \sum_i p_i$ . Then for any  $\delta > 0$ ,*

$$\Pr(X \geq \mu(1 + \delta)) \leq \exp\left(-\frac{\mu\delta^2}{3}\right)$$

$$\Pr(X \leq \mu(1 - \delta)) \leq \exp\left(-\frac{\mu\delta^2}{2}\right)$$

Question 1: Tail Bounds I ..... [20]

- (a) [5] The Markov inequality yields an *upper* bound on the probability of going far away from the mean. It is reasonable to ask whether we can do better. Show that you can't, by constructing some nonnegative random variable  $X$  and a value  $a > 0$  such that

$$\Pr(X \geq a) = \frac{E[X]}{a}$$

**Solution:** Lets take a random variable  $X = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ . The probability of selecting  $X_i$  is  $1/10$ .

$$E[X] = \sum_{i=1}^{10} x \Pr(X_i) = \frac{1}{10} \cdot \frac{10 \cdot 11}{2} = 5.5$$

By Markov inequality, taking  $a = 5.5$  we get,

$$\Pr(X \geq 5.5) \leq \frac{5.5}{5.5} = 1$$

Therefore random variable  $\geq 5.5$  are 6, 7, 8, 9, 10. As the probability of selecting any of these values is  $1/10 = 0.1$  which will be lesser than  $E[X]/a$ . the value of  $a$  cannot be lesser than 5.5, because then the probability would become greater than 1 which is wrong as per the laws of probability.

- (b) [5] Suppose we roll a fair die 100 times. Determine the probability that the sum of the rolls is at least 400 (using Markov's inequality) and the probability it is not in the range [301, 399] (using Chebyshev's inequality).

**Solution:**

The minimum sum of the rolls can be 100 and maximum sum can be 600. The expectation of a roll of a die can be 3.5. If we roll 100 times we get Expected value as  $100 * 3.5$ .

$$E(X_i) = \sum_{i=1}^6 i \cdot \Pr(i) = \sum_{i=1}^6 i \cdot (1/6) = 7/2 = 3.5$$

$$E(X) = \sum_{i=1}^{100} E[X_i] = \sum_{i=1}^{100} 3.5 = 350$$

$$Var(X_i) = E(X_i^2) - E(X_i)^2$$

The probability of a random variable from a roll is  $1/6$ .

$$Var(X_i) = \sum_{i=1}^6 i^2 * \frac{1}{6} - \left(\frac{7}{2}\right)^2$$

$$Var(X_i) = \frac{6.7.13}{6} \frac{1}{6} - \left(\frac{7}{2}\right)^2$$

$$Var(X_i) = \frac{91}{6} - \left(\frac{7}{2}\right)^2$$

$$Var(X_i) = \frac{35}{12}$$

As all the random variables are independent, therefore

$$Var(X) = \sum Var(X_i) = 100 \cdot \frac{35}{12}$$

Solving for first part using Markov's inequality:

$$\Pr(X \geq 400) \leq \frac{350}{400}$$

$$\Pr(X \geq 400) \leq \frac{7}{8}$$

The difference of 301 and 399 from 350(mean) is atmost 49. So in order to not be in that range, we have to be away from the mean by more than 50. Using Chebyshev's Inequality :

$$\Pr(|X - E[X]| \geq a) \leq \frac{Var[X]}{a^2}$$

$$\Pr(|X - 350| \geq 50) \leq \frac{100 * \frac{35}{12}}{50^2}$$

$$\Pr(|X - 350| \geq 50) \leq \frac{7}{60}$$

- (c) [5] Suppose we're given 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 bits are chosen independently and uniformly at random. What can we say about the *worst-case* behavior of the algorithm on inputs of size  $n$  (using Markov's inequality).

**Solution:**

As the expected running time is  $O(n^2)$ , lets take  $X$  as a random variable which shows the running time of the algorithm.  $E[X] = O(n^2) = cn^2$  where  $c > 0$ .

Now lets take  $a$  in terms of  $c$  so that we can nullify  $c$ .  $a \geq 2c$ .

Using Markov's Inequality

$$\Pr(X \geq a) \leq \frac{E[X]}{a}$$

$$\Pr(X \geq an^2) \leq \frac{cn^2}{an^2}$$

$$\Pr(X \geq a) \leq \frac{cn^2}{2cn^2}$$

$$\Pr(X \geq a) \leq \frac{1}{2}$$

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 our worst case of the algorithm will always be  $O(n^2)$

- (d) [5] We prove Chebyshev's inequality by applying Markov's inequality to the positive random variable  $Y = (X - E[X])^2$ . Can you generalize Chebyshev's inequality to higher moments of  $X$  (i.e. values  $E[X^k]$  for large  $k$ ). In particular, set  $k > 2$  to be some **even** number and derive a Chebyshev-like bound for the tail. What do you notice about such a bound ?

**Solution:** The proof of Chebyshev's inequality using Markov's inequality for positive random variable  $Y = (X - E[X])^2$

Markov's inequality:

$$\Pr(Y \geq a) \leq \frac{E[Y]}{a}$$

$$\Pr(|X - E[X]| \geq a) = \Pr((X - E[X])^2 \geq a^2)$$

$$\Pr((X - E(X))^2 \geq a^2) \leq \frac{E[(X - E(X))^2]}{a^2}$$

$$\Pr((X - E(X))^2 \geq a^2) \leq \frac{\text{Var}(X)}{a^2}$$

$$\Pr(|(X - E(X))| \geq a) \leq \frac{\text{Var}(X)}{a^2}$$

The proof of Chebyshev for higher moments is almost exactly the same as the one above for  $k > 2$  to be some even number.

$$Y = (X - E[X])^k$$

From above equation of 2 as higher moments, we can derive :

$$\Pr(Y \geq a) \leq \frac{E[Y]}{a}$$

As  $k$  is even we can write:

$$\Pr(|X - E[X]| \geq a) = \Pr((X - E[X])^k \geq a^k)$$

$$\Pr((X - E(X))^k \geq a^k) \leq \frac{E[(X - E(X))^k]}{a^k}$$

$$\Pr(|(X - E(X))| \geq a) \leq \frac{E[(X - E(X))^k]}{a^k}$$

The bound of Chebyshev's Inequality for higher moments of  $X$  decreases polynomially.

Question 2: Tail Bounds II ..... [15]

- (a) [5] Repeat the Chebyshev analysis for the fair die above but using a Chernoff bound instead. Since your random variables are now not 0 – 1, you will need a slightly different version of the Chernoff bound called a Hoeffding bound:

**Lemma 0.4** (Hoeffding). *Let  $X_1, \dots, X_n$  be independent random variables where  $a_i \leq X_i \leq b_i$ . Let  $S = \sum X_i$ . Then*

$$\Pr(|S - E[S]| \geq t) \leq 2 \exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right)$$

**Solution:** As per Chebyshev analysis, we got that our distance from mean was greater than 50, and our variable has a space defined by  $1 \leq X_i \leq 6$  for fair die. Using the given Lemma for Hoeffding,

$$\Pr(|S - 350| \geq 50) \leq 2 \exp\left(-\frac{2 * 50^2}{\sum_{i=1}^{100} (6 - 1)^2}\right)$$

$$\Pr(|S - 350| \geq 50) \leq 2 \exp\left(-\frac{2 * 2500}{\sum_{i=1}^{100} 25}\right)$$

$$\Pr(|S - 350| \geq 50) \leq 2 \exp\left(-\frac{5000}{2500}\right)$$

$$\Pr(|S - 350| \geq 50) \leq 2 \exp^{-2}$$

- (b) [10] Suppose you toss  $2n$  fair coins and compute the difference between the number of heads and tails. On average, this difference is zero. But what is the probability that the difference is more than  $t$ ? (**Hint:** Consider just the number of heads).

**Solution:**

Lets take number of heads is greater than tails. Let  $H$  denote number of heads and  $T$  denote number of tails. Hence

$$(H - T) \geq t$$

As total number of tosses is given by  $2n = H + T$ , i.e.  $T = 2n - H$

Substituting value of  $T$

$$(H - (2n - H)) \geq t$$

$$2H - 2n \geq t$$

$$H - n \geq \frac{t}{2}$$

Applying the Chernoff inequality, deviation from the mean  $\mu\delta = t/2$  where  $\mu = n$  therefore  $\delta = t/2n$

$$Pr(X \geq \mu(1 + \delta)) \leq \exp(-\frac{\mu\delta^2}{3})$$

$$Pr(X \geq n(1 + \frac{t}{2n})) \leq \frac{(t/2)^2}{3} = \exp(-\frac{t^2}{12n})$$

Question 3: Medians.....[10]

We saw in class that taking a median of three samples suffices to obtain a good approximation to the rank of the median. Assume that our goal is to return an element whose rank  $r$  such that  $|r - n/2| \leq \epsilon n/2$ . Assume also that we desire this result with probability  $1 - \delta$ .

We can take a sample of size  $t$  and select the median as our result. What value of  $t$  must we pick (as a function of  $\epsilon, \delta$  and  $n$ ) ? Do you notice something interesting/unusual ?

**Solution:**

For calculating median we divide our sample space in 3 partition, and we say if our rank will not lie in the middle region then, the result of the algorithm will be considered as false.

According to the above definition, the algorithm to return success/correct median of the samples, should lie in the range

$$(1 - \epsilon)E[X] \leq \text{rank} \leq (1 + \epsilon)E[X]$$

If our sample space is of  $t$  samples, at least more than  $\frac{t}{2}$  i.e,  $\lceil \frac{t}{2} \rceil$  samples should lie in the range, only then will the median be acceptable.

Applying Chernoff's inequality (for getting sample from unacceptable regions i.e,  $[1, (1 - \epsilon)E[X]]$  and  $[(1 + \epsilon)E[X], n]$ ):

$$Pr(X \geq \frac{t}{2}) \leq Pr(X \geq (1 + \epsilon)E[X]) \leq \exp^{-\frac{E[X]\epsilon^2}{3}}$$

The probability of wrong answer is  $\delta$ .

So, the probability of getting sample from any one unacceptable region is  $\frac{\delta}{2}$ .

Therefore,

$$\begin{aligned} Pr(X \geq \frac{t}{2}) &\leq Pr(X \geq (1 + \epsilon)E[X]) \leq \exp^{-\frac{E[X]\epsilon^2}{3}} \leq \frac{\delta}{2} \\ \frac{2E[X]}{t} &\geq \exp^{-\frac{E[X]\epsilon^2}{3}} \leq \frac{\delta}{2} \end{aligned}$$

Solving R.H.S of equation:

$$\begin{aligned} \exp^{-\frac{E[X]\epsilon^2}{3}} &\leq \frac{\delta}{2} \\ \mu &\geq 3\epsilon^{-2} \log 2\delta^{-1} \end{aligned}$$

Putting the value of  $\mu$  in first part of the above equation:

$$\begin{aligned} \frac{2E(X)}{t} &\leq \delta = t \geq \frac{2E(X)}{\delta} \\ t &\geq 6\delta\epsilon^{-2} \log(2\delta)^{-1} \end{aligned}$$

The value of  $t$  is not dependent upon size of the input  $n$ .

Question 4: Amplification ..... [10]

Consider a decision problem  $f$  (i.e one where the output is either zero or one). Suppose we are given a randomized algorithm  $A$  that on input  $x$  has the following properties:

- If  $f(x) = 0$ , then  $A(x) = 0$
- If  $f(x) = 1$ , then  $A(x) = 1$  with probability at least  $2/3$ .

Such an algorithm is said to have *one-sided error*, and as we've discussed in class we can amplify the probability of being correct merely by repeating it a number of times, and returning 1 if we ever see a 1, returning 0 otherwise.

But suppose our algorithm instead had the following properties:

- If  $f(x) = 0$ , then  $A(x) = 0$  with probability at least  $2/3$
- If  $f(x) = 1$ , then  $A(x) = 1$  with probability at least  $2/3$ .

In other words, it has *two-sided error* rather than one-sided error.

- (a) [5] What is the probability of success (in either case) if we implement the above procedure (returning a 1 if we ever see a 1, else returning a zero) after  $k$  iterations ?

**Solution:** The probability of success after  $k$  iterations means that we have failed in the  $k$  iteration and now got a success. The probability of a failure is  $1/3$ . Hence when we run for  $k$  iterations,

$$\Pr[\text{failure}] = (1/3)^k$$

Hence probability of success after  $k$  iterations in either case is :

$$\Pr[\text{success}] = 2 \cdot \left(\frac{2}{3}\right) \left(\frac{1}{3}\right)^k$$

- (b) [5] What is the probability of success (in either case) of the following procedure: Run the algorithm  $k$  times and return the majority answer.

**Solution:**

The majority answer of the probability of success in  $k$  iteration is when atleast more than  $k/2$  are success. i.e,

$$\begin{aligned} \Pr(\text{success in } \geq \frac{k}{2} + 1 \text{ runs}) &= \sum_{i=1}^{\frac{k}{2}} \Pr(\text{failure})^{\frac{k}{2}-i} \Pr(\text{success})^{\frac{k}{2}+i} \\ &= \sum_{i=1}^{\frac{k}{2}} \left(\frac{1}{3}\right)^{\frac{k}{2}-i} \left(\frac{2}{3}\right)^{\frac{k}{2}+i} \end{aligned}$$

Expanding it we get:

$$\begin{aligned} &= \left(\frac{1}{3}\right)^{\frac{k}{2}-1} \left(\frac{2}{3}\right)^{\frac{k}{2}+1} + \left(\frac{1}{3}\right)^{\frac{k}{2}-2} \left(\frac{2}{3}\right)^{\frac{k}{2}+2} + \dots + \left(\frac{1}{3}\right)^0 \left(\frac{2}{3}\right)^k \\ &= \frac{2^{\frac{k}{2}+1}}{3^k} [2^0 + \dots + 2^{\frac{k}{2}-1}] \end{aligned}$$

Therefore the majority answer is:

$$= \frac{2^{\frac{k}{2}+1}}{3^k} [2^{\frac{k}{2}} - 1]$$

Question 5: The “median” trick ..... [15]

Suppose we have an algorithm that generates independent samples  $X_1, \dots$  of a random variable  $X$ . A natural estimate of the value of  $X$  is the mean  $S = \sum_{i=1}^t X_i/t$  for some value of  $t$ . What is the probability that  $S$  is close to  $E[X]$  ? Specifically, what is the probability that  $|S - E[X]| \leq \epsilon E[X]$  ?

We can analyze this if we know that  $\text{Var}[X]$  is bounded. In particular, set  $r = \sqrt{\text{Var}[X]}/E[X]$ .

- (a) [5] Show that if we set  $t = O(r^2/\epsilon^2\delta)$  then the probability is at least  $1 - \delta$  (or in other words the probability of the difference *exceeding*  $\epsilon E[X]$  is at most  $\delta$ ).

**Solution:**

Given that  $S = \sum_{i=1}^t X_i/t$ , Taking Expectation on both sides,

$$E[S] = E\left[\sum_{i=1}^t X_i/t\right]$$

Using linearity property of expectation, we get:

$$E[S] = \sum_{i=1}^t E[X_i/t]$$

$$E[S] = \frac{1}{t} \sum_{i=1}^t E[X_i]$$

$$E[S] = \frac{1}{t} E(X) * t = E[X]$$

Similarly we can figure out Variance of  $S = \sum_{i=1}^t X_i/t$

$$\text{Var}[S] = \text{Var}\left[\sum_{i=1}^t X_i/t\right]$$

By applying rule of Variance - Multiplying a random variable by a constant increases the variance by the square of the constant, we get:

$$\text{Var}[S] = \frac{1}{t^2} \text{Var}\left(\sum_{i=1}^t X_i\right)$$

As all  $X_i$ 's are independent, therefore using linearity of variance we get

$$\text{Var}[S] = \frac{t}{t^2} \text{Var}[X]$$

$$\text{Var}[S] = \frac{1}{t} \text{Var}[X]$$

Now if we apply Chebyshev's inequality we get:

$$\Pr(|S - E[S]| \geq \epsilon E[S]) \leq \frac{\text{Var}[S]}{\epsilon^2 E[S]^2}$$

Substituting the equations from above:

$$\Pr(|S - E[S]| \geq \epsilon E[S]) \leq \frac{\frac{\text{Var}[X]}{t}}{\epsilon^2 E[X]^2}$$



We are given that:

$$r = \frac{\sqrt{\text{Var}[X]}}{E[X]}$$

$$r^2 = \frac{\text{Var}[X]}{E[X]^2}$$

Substituting:

$$\Pr(|S - E[S]| \geq \epsilon E[S]) \leq \frac{r^2}{t\epsilon^2}$$

$$\Pr(|S - E[S]| \geq \epsilon E[S]) = \delta$$

Substituting:

$$\delta \leq \frac{r^2}{t\epsilon^2}$$

$$t \leq \frac{r^2}{\delta\epsilon^2}$$

$$t = O\left(\frac{r^2}{\delta\epsilon^2}\right)$$

- (b) [2] Suppose we only want a *weak estimate*: an estimate  $\tilde{X}$  that is within  $\epsilon E[X]$  with probability  $2/3$ . Show that in this case we need only set  $t = O(r^2/\epsilon^2)$ .

**Solution:**

Considering we have  $t$  weak estimations. For weak estimation the probability is at least  $1 - \delta = \frac{2}{3}$ . Therefore,  $\delta = \frac{1}{3}$ . Using the above equation, we can write

$$t = c \left( \frac{r^2}{\delta\epsilon^2} \right) = 3c \left( \frac{r^2}{\epsilon^2} \right)$$

$$t = O\left(\frac{r^2}{\epsilon^2}\right)$$

- (c) [8] Now take the *median* of  $O(\log(1/\delta))$  such weak estimates. Prove that this estimator does have the desired property of being within  $\epsilon E[X]$  of the true estimate with probability  $1 - \delta$ . Note that we have done two things.

1. We've reduced the number of samples needed from  $r^2/\epsilon^2\delta$  to  $r^2 \log(1/\delta)/\epsilon^2$ , which is an exponential reduction in dependence on  $1/\delta$ .
2. We've achieved a strong bound without using full Chernoff-like assumptions: we only needed  $X$  to have bounded variance.

**Solution:**

If the median of the weak estimates satisfies the condition, it means less than half of the weak estimates are not within  $\epsilon E[X]$  of the true value of  $E[X]$ .

Lets use a new random variable  $X_i$ :

$X_i = 1$  if the  $i^{th}$  weak estimate fall above  $\epsilon E[X]$  of  $E[X]$

$X_i = 0$  if the  $i^{th}$  weak estimate fall below  $\epsilon E[X]$  of  $E[X]$

$X_i$  follows binomial distribution with probability of 1/4 or more to be 1 and 3/4 or less to be 0. For simplicity, we use 1/4 in this problem. Lower probability will need lower number of estimates.

If we use  $X = \sum_{i=1}^m X_i$  to represent how many weak estimates fall above  $(1 + \epsilon)E[X]$ , we will be able to use Chernoff bound to for the value of  $m$  so that

$$Pr(X \geq m/2) < \delta$$

Chernoff bound gives:

$$Pr(X \geq (1 + \delta')E[X]) \leq e^{-E[X]\delta'^2/3}$$

where  $E[X] = m/4$ . Use  $\delta' = 1$ , we have

$$Pr(X \geq m/2) \leq e^{-m/12}$$

By using  $m = 12 \cdot \log(1/\delta)$ , we have  $Pr(X \geq m/2) \leq \delta$ , so that the probability that the median of weak estimates gives result within  $\epsilon E[X]$  is at least  $1 - \delta$ .

*Solution found on [http : //people.cs.pitt.edu/ ihsan/hw2.pdf](http://people.cs.pitt.edu/ihsan/hw2.pdf)*

Question 6: Hashing with open addressing ..... [15]

We've seen how to use random hash function to ensure only a small number of collisions in a hash bucket. We stored the overflow items in a *chain*, which is why we wanted to minimize its length. Such a hashing scheme is called *chaining*.

An alternate approach to hashing is called *open addressing*. Suppose we want to store  $n$  items in an array. We maintain an array of  $2n$  slots, and also construct a hash function  $h : U \times \mathbb{Z} \rightarrow [1 \dots 2n]$ . For each key  $k$ , the sequence  $h(k, 0), h(k, 1), \dots$  defines a *probe sequence* as follows.

- When  $k$  appear, the algorithm first tries to place it in  $h(k, 0)$ . If that entry is full, it tries  $h(k, 1)$ . If that is full, it tries  $h(k, 2)$  and so on.
- When searching for an item  $q$ , we search the entries  $h(q, i), i = 0, 1, \dots$  until we either find  $q$  or find an empty spot (which means that  $q$  was never in the set).

Assume that  $h(k, j)$  is uniform over  $[1 \dots 2n]$  for any  $k$  and that all  $h(k, i)$  are independent.

- (a) [5] Show that the probability an insertion takes more than  $r$  steps is at most  $2^{-r}$ .

**Solution:**

As we have to insert  $n$  items in  $2n$  slots therefore at any point of time probabily of getting the empty slot  $\frac{2n-i}{2n} \geq \frac{1}{2}$ , where  $i = 1 \dots n$  and hence probability of failure is at most  $\frac{1}{2}$ . If the insertion requires more than  $r$  steps and supposing all the  $r$  steps fail, then the probabily this happens at most is  $(\frac{1}{2})^r$  or  $2^{-r}$

- (b) [5] Show that for the  $i^{th}$  insertion ( $i = 1, 2, \dots, n$ ), the probability that more than  $2 \log n$  probes are needed is  $1/n^2$ .

**Solution:**

In above question we proved that the probability an insertion takes more than  $r$  probes is at most  $2^{-r}$ .

Therefore for  $i^{\text{th}}$  insertion, the probability that more than  $2 \log_2 n$  probes are needed is proved by replacing  $r$  with  $2 \log_2 n$  in above equation which is  $2^{-2 \log_2 n} = \frac{1}{n^2}$

- (c) [5] Let  $X_i$  be the number of probes needed to insert item  $i$ , and let  $X = \max X_i$ . Show that  $\Pr(X > 2 \log n) \leq 1/n$ .

**Solution:**

As  $X_i$  is the number of probes needed to insert item  $i$ , the value  $i$  can vary from 1 to  $n$ .

As proved in 6b  $\Pr(X_i > 2 \log_2 n) = \frac{1}{n^2}$

The probability of all random variables  $X_1 \dots X_n$  is at least  
 $\Pr(X_1 > 2 \log_2 n) + \Pr(X_2 > 2 \log_2 n) \dots + \Pr(X_n > 2 \log_2 n)$

Summing up the above equation we get:

Therefore  $\Pr(X > 2 \log_2 n) = n \cdot \frac{1}{n^2} = \frac{1}{n}$

Question 7: Derandomizing MAX CUT ..... [15]

While randomness is an easy way to design an algorithm, random bits are an expensive resource, and we'd like to use as few of them as possible. The process of removing randomness from an algorithm is called *derandomization*. There are three basic approaches to derandomization: using weak randomness, eliminating randomness by brute force, and the method of conditional expectations. We will examine the latter two methods here.

- (a) [5] Suppose we have a randomized algorithm that runs in time  $T(n)$ , uses  $O(\log n)$  random bits and returns the minimum of some function with a probability of success greater than  $2/3$ . Construct a deterministic algorithm that solves the same problem correctly in time  $T'(n)$ , expressed in terms of  $T(n)$  and  $n$  (**Hint:** this is called the method of brute force).

**Solution:** When we select  $\log n$  random bits the entire combination of the bits exhaustively is  $2^{\log n} = n$ . When we wanna perform brute force we will have to run over the entire set of bits exhaustively. Hence our random selection becomes polynomial.

When we run an algorithm to get minimum of some function using polynomial bits until we get success atleast  $2/3$  and the algorithm runs in  $T(n)$ , therefore  $T'(n) = n \cdot T(n)$

Algorithm:

1. Iterate from 1 to  $n$ :
2. find the minimum of function

The other approach works as follows. Suppose our randomized algorithm  $A$  produces some answer whose expectation is  $\mu$ . We can imagine the algorithm as a sequence of deterministic operations punctuated by coin tosses  $r_1, r_2, \dots, r_m$ . The value output by  $A$  is a function of the input and these coin tosses: ignoring the input, we can write the output value as  $A(r_1, \dots, r_m)$ , and so  $E[A(r_1, \dots, r_m)] = \mu$

Consider the first coin toss  $r_1$ . We can write

$$E[A(r_1, \dots, r_m)] = (1/2)E[A(0, r_2, \dots, r_m)] + (1/2)E[A(1, r_2, \dots, r_m)]$$

This is because the probability of  $r_1 = 1$  is  $1/2$ . Notice however that one of the two expectations  $E_0 = E[A(0, r_2, \dots, r_m)]$  and  $E_1 = E[A(1, r_2, \dots, r_m)]$  must be at least as large as  $\mu$  (since  $\mu = (E_0 + E_1)/2$ ). Suppose we can compute these two numbers. Then we can *deterministically* pick the value of  $r_1$  and we are sure that the final answer we get is no worse than the expected value, while having used one less random bit.

Repeating this over and over again yields a final result in which all bits are picked deterministically, and yet the final answer is at least as large as  $\mu$ . Thus, we've derandomized the algorithm without paying a penalty. The catch is of course the estimation of  $E_0$  and  $E_1$ .

Now consider the randomized algorithm for MAX CUT that yields a 0.5-approximation: Pick a vertex and label it as being black or white with equal probability. Repeat for all vertices, and return the partition into black and white as the desired cut.

- (b) [10] Derandomize this algorithm using the method of conditional expectations. What is the resulting deterministic algorithm? (**Hint:** think about how we might deterministically decide which color to give to a vertex based on the graph structure and how it influences  $E_0, E_1$ .)

**Solution:** Let there be a graph defined by  $G = (V, E)$ , where  $V$  is vertices and  $E$  is edge.  $|V|$  is number of vertices and  $|E|$  is number of edges. Instead of randomly selecting based on coin tosses how to divide the vertices we can check conditionally.

Lets take  $B$  and  $W$  as sets for black and white vertices.

Algorithm:  $G = (V, E)$

1. Set  $B$  &  $W$  as empty

2. Iterate from 1 to  $n$

(a) If  $|cut(i, B)| > |cut(i, W)|$ , then set vertex as white and add  $i$  to  $W$

(b) Else set as black and add to  $B$

When a cut is made we check for if the number of vertices connected to the edges are present in either set. If the total weight of the edges is lesser in either set, add the vertex to the set with lesser weight. Therefore the addition of a vertex is conditional on the minimum of the total weight of edges in either set.

## References

Collaboration with Shweta, Sunny and Yogesh

<http://cjluser.v.iis.sinica.edu.tw/~josephccln/research/randalg/exercises/exercise3.pdf>

<http://www.cs.helsinki.fi/u/jkivinen/opetus/ra1/spring2013/sol5.pdf>

<http://courses.csail.mit.edu/6.006/fall11/lectures/lecture10.pdf>

<http://www.cse.buffalo.edu/~hungngo/classes/2008/694/notes/handout3.pdf>

<http://sublinear.wikischolars.columbia.edu/file/view/scribe2.pdf/559024365/scribe2.pdf>

<http://crypto.stanford.edu/~blynn/pr/markov.html>