

CS5234 Cheat Sheet Gabriel Yeo

Inequalities

Markov:	$P(X \geq \alpha) \leq \frac{E(X)}{\alpha}$
Chebyshev:	$P(X - \mu \geq k) \leq \frac{Var(X)}{k^2}$
Chernoff:	$P(X \leq (1 - \delta)\mu) \leq e^{-\frac{\delta^2 \mu}{2}} \leq e^{-\frac{\delta^2 \mu}{3}}$ $P(X \geq (1 + \delta)\mu) \leq e^{-\frac{\delta^2 \mu}{2 + \delta}} \leq e^{-\frac{\delta^2 \mu}{3}}$ $X = X_1 + \dots + X_n$, $X_i \in \{0, 1\}$, (ind. Bernoulli)
Chernoff:	$P(X - \mu \geq \delta\mu) \leq 2e^{-\frac{\delta^2 \mu}{3}}$
Hoeffding:	$P(X - \mu \geq t) \leq 2e^{-\frac{t^2}{n}}$ $X = X_1 + \dots + X_n$, $X_i \in [0, 1]$
Hoeffding:	$P(\bar{X} - E[\bar{X}] \geq t) \leq 2e^{-2nt^2}$
Hoeffding (gen):	$P(\bar{X} - E[\bar{X}] \geq t) \leq 2e^{-\frac{2t^2}{\sum (a_i - b_i)^2}}$ $X_i \in [a_i, b_i]$

Facts

Taylor expansion:	$e^x = 1 + x + \frac{x^2}{2!} + \dots = \sum_{n=0}^{\infty} \frac{x^n}{n!}$
For $0 < x \leq 1$:	$1 - x \leq e^{-x} \leq 1 - \frac{x}{2}$
For $0 < x \leq 1$:	$\frac{1}{e^2} \leq (1 - x)^{1/x} \leq \frac{1}{e}$
Approx $\binom{a}{b}$:	$\left(\frac{a}{b}\right)^b \leq \binom{a}{b} \leq \left(\frac{ea}{b}\right)^b$
For $0 < x < 1$:	$\sum_{i=0}^{\infty} a^i = \frac{1}{1-a}$
Natural log:	$\ln(n-1) \leq \sum_{i=1}^n \frac{1}{i} \leq \ln(n) + 1$
Powers of 2:	$\sum_{i=0}^n 2^i = 2^{n+1} - 1$
$1/e$:	$= 0.36787944117$
$1/e^2$:	$= 0.13533528323$

Probability

Expectation:	$E[X] = \sum_{v \in D} v \cdot Pr[X = v]$
Variance:	$Var[X] = E[(X - E[X])^2]$ $= E[X^2] - E[X]^2$
More variance:	$Var[aX] = a^2 Var[X]$
Linearity of expectation:	$E[\sum X_i] = \sum E[X_i]$
If independent X_i :	$Var[\sum X_i] = \sum Var[X_i]$
Conditional:	$P[X \text{ and } Y] = P[X Y] \cdot P[Y]$
Independence:	$P[X \text{ and } Y] = P[X] \cdot P[Y]$
Union bound:	$P[\bigcup E_i] \leq \sum P[E_i]$

Complexities

Algorithm	Classical	Approx
BFS	$O(n + m)$	
Connected?	$O(n + m)$	$O(\frac{1}{\epsilon^2 d})$
# CCs	$O(n + m)$	$O(\frac{d}{\epsilon^3})$
MST weight	$O(m \log n)$	$O(\frac{dW^4 \log W}{\epsilon^3})$
Prim's	$O((m + n) \cdot \log n)$	
Prim's (Fib heap)	$O(m + n \log n)$	
Kruskal's (MST)	$O(m \log n)$	
Dijkstra	$O((m + n) \cdot \log n)$	
Dijkstra (Fib heap)	$O(m + n \log n)$	

Lecture 1 All 0s

Give an array A with n elements, $A[i] \in \{0, 1\}$:
 (1) If array has $\geq \epsilon n$ 1's, return False with probability at least $1 - \delta$:
 Assume $\geq \epsilon n$ 1's, then for sample i :
 $Pr[A[i] = 1] \geq \epsilon n / n \geq \epsilon$.

$$\begin{aligned}
 Pr(\text{all samples are 0}) &\leq (1 - \epsilon)^s \\
 &\leq (1 - \epsilon)^{\frac{\ln(1/\delta)}{\epsilon}} \\
 &\leq e^{-\ln(1/\delta)} \\
 &\leq \delta
 \end{aligned}$$

$$\text{Fix } s = \frac{\ln(1/\delta)}{\epsilon}$$

Lecture 1 Number of 1s

Give an array A with n elements, $A[i] \in \{0, 1\}$:
 (1) Find number of 1's $\pm \epsilon$ with probability at least $1 - \delta$.
 Let Y_i be s independent samples in $\{0, 1\}$.
 Output $= Z = 1/s \sum Y_i$
 Probability of failure:

$$\begin{aligned}
 Pr(|Z - E[Z]| \geq \epsilon) &\leq 2e^{-2s\epsilon^2} \\
 &\leq 2e^{-2\frac{\ln(2/\delta)}{2\epsilon^2}\epsilon^2} \\
 &\leq 2e^{-\ln(2/\delta)} \\
 &\leq 2e^{\ln(\delta/2)} \\
 &\leq 2 \cdot \delta/2 \\
 &\leq \delta
 \end{aligned}$$

$$\text{Fix } s = \frac{\ln(2/\delta)}{2\epsilon^2}$$

Lecture 2 Connectivity

G is ϵ -close to connected if you can modify at most ϵn entries in the adjacency list to make it connected.
 If $G(V, E)$ is connected, output True, else if ϵ -far from connected, output False.

```

Connected(G, n, d, ε)
  Repeat 16/εd times:
    - Choose a random node u
    - Do a BFS from u, stopping after 8/εd nodes seen.
    - If CC of u has ≤ 8/εd nodes, return FALSE.
  return TRUE
  
```

BFS cost $= \frac{8}{\epsilon d} \cdot d$
 Total complexity $= O(\frac{1}{\epsilon^2 d})$

Lecture 2 Connected Components

Output CC such that: $|CC(G) - C| \leq \epsilon n$, w.p. $> 1 - \delta$
 Define $n(u)$ = num nodes in the CC of u .
 $cost(u) = 1/n(u)$.
 $\sum_{u \in CC_i} cost(u) = 1$.

```

sum = 0
for j = 1 to s:
  - Sample u randomly
  - BFS from u, stop when see up to 2/ε nodes
  - If BFS found > 2/ε node:
    - sum := sum + ε/2
  - Else:
    - sum := sum + cost(u)
return n * (sum/s)
  
```

Let $\bar{C} = \sum cost(u_j)$
 Let $Y_j = cost(u_j)$ of our sample j
 $|CC(G) - \bar{C}| \leq \epsilon n/2$
 $E[Y_j] = \sum \frac{1}{n} cost(u_i) = \frac{1}{n} \bar{C}$
 $E[\sum Y_j] = s \cdot E[Y_j] = \frac{s}{n} \bar{C}$
 Since we output $\frac{n}{s} \sum Y_j$, we get $E[\frac{n}{s} \sum Y_j] = \bar{C}$

$$\begin{aligned}
 P(|\bar{C} - \frac{n}{s} \sum Y_j| > \epsilon n/2) \\
 &= P(|E[\sum Y_j] - \sum Y_j| > \frac{s}{n} \epsilon n/2) \\
 &= P(|E[\sum Y_j] - \sum Y_j| > s\epsilon/2) \\
 &\leq 2e^{-2\epsilon^2 s^2 / 4s} \\
 &\leq 2e^{-\epsilon^2 s/2} \leq \delta
 \end{aligned}$$

Set $s = \frac{2}{\epsilon^2} \ln(2/\delta)$.
 Complexity $= 2d/\epsilon \cdot O(\frac{1}{\epsilon^2} \ln(1/\delta)) = O(\frac{d}{\epsilon^3} \ln(1/\delta))$.

Lecture 3 MST weight

Output M such that: $M = MST(G)(1 \pm \epsilon)$ w.p. $> 1 - \delta$.
 Let G_j be the graph with edge weights j and below.
 Let C_j be the connected components in G_j .
 $MST(G)$ contains $C_j - 1$ edges of weight $> j$.
 Therefore:

$$\begin{aligned} MST(G) &= (n - C_1) + \sum_{j=1}^{W-1} (j+1)(C_j - C_j + 1) \\ &= n - W + \sum_{j=1}^{W-1} C_j \end{aligned}$$

```
sum := n - W
for j = 1 to W - 1:
    -  $X_j = \text{ApproxCC}(G_j, d, \epsilon' \delta')$ 
    - sum := sum +  $X_j$ 
return sum
```

Sum of errors: $(\epsilon' n)(W - 1)$
 Set $\epsilon' = \epsilon/W$, then sum of errors $\leq \epsilon n$.
 Set $\delta' = \delta/W$. Then $P(\text{any fail}) \leq \sum_1^{W-1} \delta/W \leq \delta$.
 $MST(G) - \epsilon n \leq \text{sum} \leq MST(G) + \epsilon n$
 Since $MST(G) \geq n - 1 \geq n/2$, $n \leq 2MST(G)$,
 $MST(G)(1 - 2\epsilon) \leq \text{sum} \leq MST(G)(1 + 2\epsilon)$

Lecture 3 Maximal Matching

Return the size of the Maximal Matching.

```
query(e):
    for all neighbours  $e'$  of  $e$ :
        if hash( $e'$ ) < hash( $e$ )
            if query( $e'$ ) = TRUE
                return FALSE
    return TRUE

sum := 0
for j = 1 to s:
    - Choose edge  $e$  uniformly at random.
    - if (query( $e$ ) = True)
        sum := sum + 1
return m.sum/s
```

$$E[\text{cost}] = 2 \sum_{k=1}^{\infty} \frac{d^k}{k!} = O(e^d).$$

$$\text{sum} = MM(G) \pm \epsilon m.$$

$$\text{Complexity} = O\left(\frac{e^d}{\epsilon^2}\right).$$

Can do better: $O(\frac{d^4}{\epsilon^2})$, even $O(\frac{d^2}{\epsilon^2})$, and reduce error to $\pm \epsilon n$.

Lecture 3 Yao's Mini-Max

Every randomized algorithm on a worst-case input is always slower than the best deterministic algorithm on the worst distribution.

$$\begin{aligned} \forall A \in R : \max_{x \in X} (E[\text{cost}(A, x)]) \\ \geq \min_{B \in D} (E[\text{cost}(B, x \text{ chosen from } \gamma)]) \end{aligned}$$

Recipe:

- Choose distribution γ .
- Show that the expected cost of every deterministic algorithm from γ is greater than some cost c .
- Conclude that every randomized algorithm has at least one input with expected cost at least as bad as c .