Prob in Computing (Assgn-3)

Sai Harsha K
CS17BTECH11036

1) ⓐ To prove: $E[2^{C_n}] = n+1.$

$$E[2^{C_n}] = \sum_{1\le k\le n} 2^k \Pr(C_n = k)$$

and $\Pr(C_n = k) = \frac{1}{2^{k-1}} \Pr(C_{n-1} = k-1) + \left(1-\frac{1}{2^k}\right) \Pr(C_{n-1} = k) \quad \ldots ①$

$$E[2^{C_n}] = \sum_{1\le k\le n} \left(2P(C_{n-1}=k-1) + (2^k-1) P(C_{n-1}=k)\right)$$

$$= \sum_{1\le k\le n} \left(2P(C_{n-1}=k-1) - P(C_{n-1}=k) + 2^k P(C_{n-1}=k)\right)$$

using,
$\sum_{1\le k\le n} P(C_n=k) = 1$ and $P(C_{n-1}=n) = 0$

we get,

$$E[2^{C_n}] = 2 - 1 + \sum_{1\le k\le n-1} 2^{k} P(C_{n-1}=k). \quad (\because P(C_{n-1}=n)=0)$$
$$+ P(C_{n-1}=0) - 2^0 P(C_{n-1}=0)$$

$$= E[2^{C_n}] = 1 + E[2^{C_{n-1}}].$$

Base case:

$E[2^{C_1}] = 2.P(C_1=1) = \underline{\underline{2}}. \quad (\because P(C_1=1) = \frac{1}{2^0} = 1)$

$\Rightarrow E[2^{C_n}] = n+1.$

$$E[2^{2C_n}] = \sum_{1\le k\le n} 2^{2k}\, Pr(C_n=k).$$ using prev stmt ①,

$$= \sum_{1\le k\le n}\left(2^{k+1} P(C_{n-1}=k-1) - 2^{k} P(C_{n-1}=k) + 2^{2k} P(C_{n-1}=k)\right)$$

$$= 2^{2} E[2^{C_{n-1}}] - 1\,E[2^{C_{n-1}}] + 2^{0} P(C_{n-1}=0) - 2^{0} P(C_{n-1}=0)$$
$$+ E[2^{2C_{n-1}}]$$

$(\because P(C_{n-1}=n)=0)$

$$\Rightarrow E[2^{2C_n}] = 3E[2^{C_{n-1}}] + E[2^{2C_{n-1}}]$$
$$= 3n + E[2^{2C_{n-1}}].$$

<u>Base case</u>:

$$E[2^{2C_1}] = 2^{2}.\ Pr(C_1=1) = \underline{\underline{4}}.$$

$$\Rightarrow E[2^{2C_n}] = 3n + 3(n-1) + \dots + (3+1)$$
$$= 3\left(\frac{n(n+1)}{2}\right) + 1$$

$$Var(2^{C_n}) = E[2^{2C_n}] - \left(E[2^{C_n}]\right)^2$$
$$= 3\left(\frac{n(n+1)}{2}\right) + 1 - (n+1)^2$$
$$= \frac{3}{2}(n^2+n) + 1 - (n^2+1+2n)$$
$$= \frac{n^2-n}{2} = \frac{n(n-1)}{2}$$

2) <u>Algo</u>

1) choose t hash functions $h_1, \dots h_t : \{1, 2, \dots m\} \rightarrow \{1, 2, \dots k\}$ ~~[illegible]~~

$h_{[t/2]+1}, \dots h_t : \{m, m+1, \dots\} \rightarrow \{1, 2, \dots k\}$

each pairwise independent hash family

2) For $1 \le i \le t$ & $1 \le j \le k$ initialize counters $C[i, j]$ to zero. The counters $C[i, 1], \dots C[i, k]$ correspond to i^{th} hash function.

~~3) For each tuple $(x_i, a(x_i))$ in stream, do:~~
~~set $i_1 = h_1(x_i), \dots, i_t = h_t(x_i)$.~~
~~Increment each of $C[1, i_1], C[2, i_2], \dots C[t, i_t]$ by $a(x_i)$.~~

~~4) For each $y \in \{1, \dots, m\}$. output $\{C[1, h_1(y)], C[2, h_2(y)], \dots C[t, h_t(y)]\}$~~

~~as estimated frequency defined.~~

3) divide the hash functions into two halves. first $[t/2]$ for positive $a(x_i)$ and other half $t - [t/2]$ for negative $a(x_i)$.

4) For each tuple $(x_i, a(x_i))$ in stream do:
if $a(x_i) > 0$
set $i_1 = h_1(x_i), \dots, i_{[t/2]} = h_{[t/2]}(x_i)$
Increment each of $C[1, i_1], C[2, i_2], \dots, C[[t/2], h_{[t/2]}(x_i)]$
by $a(x_i)$.

else
set $i_{[t/2]+1} = h_{[t/2]+1}(x_i), \dots \; i_t = h_t(x_i)$
Increment each of $C[[t/2]+1, h_{[t/2]+1}(x_i)], \dots C[t, h_t(x_i)]$
by $-a(x_i)$.

5) For each $y \in \{1, \dots m\}$ output $\min\{C[1, h_1(y)], \dots C[[t/2], h_{[t/2]}(y)]\}$
$- \min\{C[[t/2]+1, h_{[t/2]+1}(y)], \dots C[t, h_t(y)]\}$.
as estimated frequency defined.

Now, Let $F_p(y) = \sum_{y \in stream} a(y)$, where all such ~~St.~~ $a(y) > 0$

$$F_N(y) = - \sum_{y \in stream} a(y) \text{, where all such } a(y) < 0$$

Now, Fix $y \in \{1, \dots m\}$ where $a(y) > 0$ & $i \in \{1, \dots [t/2]\}$

$$E[c(i, h_i(y))] = f(y) + \frac{F_p(y) - f(y)}{k} \leq f(y) + \frac{F_p(y)}{k}.$$

(where $f(y)$ is ~~at~~ $\sum_{i \in X} a(x_i)$, X is all such occurrence of x.)

Similarly for negative part,
$y \in \{1, \dots m\}$ where $a(y) < 0$ & $j \in \{[t/2]+1, \dots t\}$.

$$E[c(j, h_j(y))] = f(y) + \frac{F_N(y) - f(y)}{k} \leq f(y) + \frac{F_N(y)}{k}.$$

(where $f(y)$ is $-\sum_{i \in X} a(x_i)$, X is occurrence of all such x)

Let $k = \left[\frac{2}{\varepsilon}\right]$. Using markov's inequality,

$$Pr\left(c[i, h_i(y)] - f(y) \geq \varepsilon F_p(y)\right) \leq \frac{1}{\varepsilon k} \leq \frac{1}{2}.$$

$$Pr\left(\min_{1 \leq k \leq [t/2]} (c[k, h_k(y)]) - f(y) \geq \varepsilon F_p(y)\right) \leq \frac{1}{2^{[t/2]}} \quad \dots \textcircled{1}$$

Similarly for negative part,

$$Pr\left(\min_{[\frac{t}{2}]+1 \leq k \leq t} (c[k, h_k(y)] - f(y) \geq \varepsilon F_N(y)\right) \leq \frac{1}{2^{t - [\frac{t}{2}]}} \approx \frac{1}{2^{[t/2]}}$$

$$\dots \textcircled{2}$$

we need to find error,

so $|$ ~~Pred~~ output (x) - actual $(x)|$

$$\le \left| \min\left(\underset{\text{counter}}{\text{Pos}}(x)\right) - \min\left(\underset{\text{counter}}{\text{Neg}}(x)\right) - \left(\text{Pos sum}(x) + \text{neg sum}(x)\right) \right|$$

$$\le \left| \min\left(\underset{\text{counter}}{\text{Pos}}(x)\right) - \min\left(\underset{\text{counter}}{\text{Neg}}(x)\right) - \left(\text{Pos sum}(x) - (-\text{neg sum}(x)\right) \right|.$$

(where,
Pos counter is our C[ ] ~~for~~ value & Pos sum is f_P, neg sum is f_N)

Let K be sum of all $a(x_i)$ in steam.

$\Rightarrow$ Prob $\left(|\text{output}(x) - \text{actual}(x)| \ge \varepsilon(K)\right)$

$$\le \text{Prob}\left(\min\left(\underset{\text{counter}}{\text{Pos}}(x)\right) - \text{Pos sum}(x) \ge \varepsilon F_P(x)\right)$$

$$+ \text{Prob}\left(\min\left(\underset{\text{counter}}{\text{Neg}}(x)\right) - (-\text{neg sum}(x)) \ge \varepsilon F_N(x)\right)$$

Using ① and ②;

(Let $t = 2(1+c)\log_2 m$)

$$\le \frac{2}{2^{\lceil t/2 \rceil}} \le \frac{2}{2^{t/2}} \le \frac{1}{m^{1+c}}$$

for $c = \frac{1}{\log m} \log\left(\frac{1}{\delta}\right)$

we get error to be at most δ.

we get error to be at most δ after applying union bound on m elements as seen in lecture notes.

Let $Z = \text{Max}(F_P, F_N)$. Space comp $= O\left(\frac{\log(Z)}{\varepsilon}\left(\log m + \log\left(\frac{1}{\delta}\right)\right)\right)$

3) Algorithm

// Let $E_A(i)$ be expected no of edges in cut when v_i is placed in A, and similarly $E_B(i)$, $E_C(i)$.

// $E_A(i) = m_0 + dB + dC + (dD \times 2/3)$; where m_0 is no of edges currently in cut
dB = ~~edg~~ neighbours of v_i currently in B, similarly dC and dD is unplaced neighbours of v_i.

```
set $A = \phi, B = \phi, C = \phi$

for $i = 1$ to $n$, do:
    find number of current neighbours of $v_i$ in A, B, C respectively.
    call them $d_A, d_B, d_C$.
    if $dB + dC > dA + dC$ and $dB + dC > dA + dB$, set $A = A \cup \{v_i\}$
else if $dA + dC > dB + dC$ and $dA + dC > dA + dB$, set $B = B \cup \{v_i\}$
else set $C = C \cup \{v_i\}$

output A, B, C.
```

Let $\mu(i)$ be expected number of edges in cut after placing first i vertices.

~~$\mu(0) =$~~

Let $H_e = \begin{cases} 1, & \text{edge } (u,v) \text{ s.t. } u \text{ and } v \text{ are in different set} \\ 0, & \text{else} \end{cases}$

$$\Rightarrow \mu(0) = \sum_{e \in E} H_e = \sum_{e \in E} \Pr(u, v \text{ are in different set})$$

$$= \frac{2m}{3}, \text{ where } m = |E|, \Pr(u, v \text{ in diff set}) = \frac{2}{3}.$$

Also, $\mu(i-1) = \frac{1}{3}\left(\mu(i|_{v_i \in A}) + \mu(i|_{v_i \in B}) + \mu(i|_{v_i \in C})\right)$

& $\mu(i) = \max\left(\mu(i|_{v_i \in A}), \mu(i|_{v_i \in B}), \mu(i|_{v_i \in C})\right)$

as max $\geq$ avg $\Rightarrow \mu(i) \geq \mu(i-1)$. $\Rightarrow$ ~~for any i~~ $\mu(n) \geq \frac{2m}{3}$.