Counting: Stream $e_1, e_2, \ldots, e_N$

Goal: return an estimate of N using small space

- Morris Counter: $(1 \pm \varepsilon, \frac{1}{2\varepsilon^2})$ estimator
  - ↳ with prob. $1 - \frac{1}{2\varepsilon^2}$: $|\hat{n} - N| \leq \varepsilon \cdot N$
  - $(1-\varepsilon)N \leq \hat{n} \leq (1+\varepsilon)N$
- Morris+ Counter: $\Theta(\frac{1}{\varepsilon^2})$ independent instances of Morris Counter
  - ↳ $(1 \pm \varepsilon, \frac{1}{4})$
- Morris++ Counter: $\Theta(\log(\frac{1}{\delta}))$ independent instances of Morris+ Counter
  - ↳ $(1 \pm \varepsilon, \delta)$

Median trick

$\ell = \lceil 12 \ln(\frac{1}{\delta}) \rceil$ independent instances of Morris+ Counter

return $\hat{n}_{final} = \text{median}(\hat{n}_1, \ldots, \hat{n}_\ell)$

($\hat{n}_i$: result of i-th instance)

Analysis: (We want: $(1-\varepsilon)\cdot N \le \hat{n}_{final} \le (1+\varepsilon)\cdot N$)

otherwise: fails (to give correct approximation)

$\hat{n}_{final}$ fails only if majority of $\hat{n}_1, \dots, \hat{n}_\ell$ fails

random variables $Z_i := \begin{cases} 1 & \text{if } \hat{n}_i \text{ fails} \\ 0 & \text{o.w.} \end{cases}$

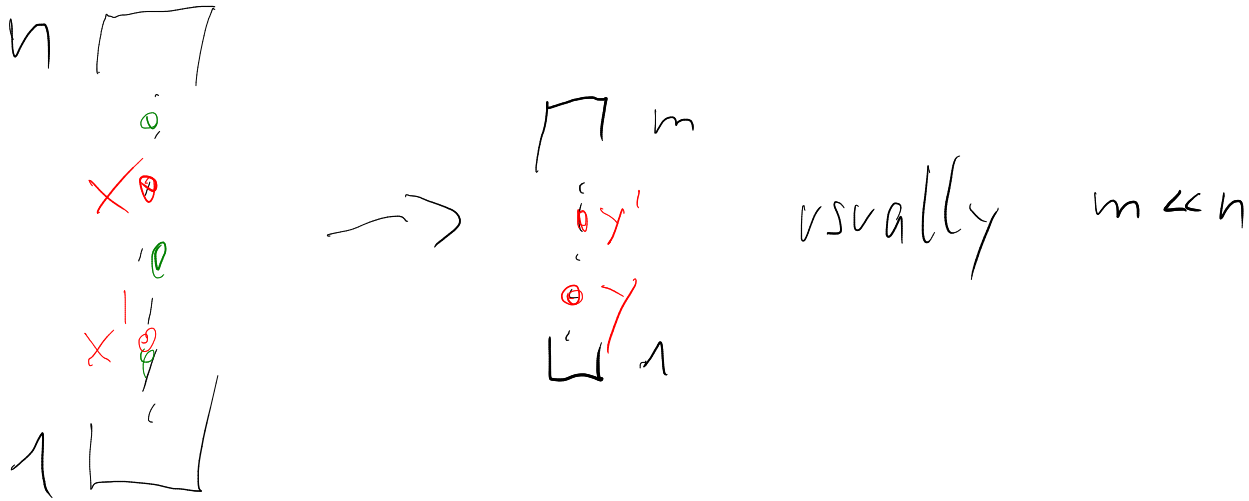
By analysis of Morris counter: $\Pr[Z_i = 1] \le \underbrace{\frac{1}{4}}_{p}$ for all $1 \le i \le \ell$

Total number of failures $Z := \sum_{i=1}^{\ell} Z_i$ $\quad \mu := p\cdot\ell = \frac{1}{4}\cdot\ell$

$$\Pr[\hat{n}_{final} \text{ fails}] = \Pr\left[Z \ge \frac{\ell}{2}\right] = \Pr\left[Z \ge \underbrace{(1+1)}_{1+\delta} \cdot \underbrace{\frac{\ell}{4}}_{\mu}\right] \le \frac{1}{e^{\frac{1}{3}\cdot\mu}} = \frac{1}{e^{\frac{1}{12}\ell}} \le \frac{1}{e^{\ln(\frac{1}{\delta})}} = \delta$$

Chernoff Bound

Hash functions



Families of hash functions $h \sim \mathcal{H}$ $h : [n] \to m$ $\quad [n] := \{1, 2, \ldots, n\}$

- uniform $\forall x \in [n] \; \forall y \in [m] \; \Pr[h(x) = y] = \frac{1}{m}$
- universal $\forall x, x' \in [n]$ (if $x \neq x'$) $\Pr_{h \sim \mathcal{H}}[h(x) = h(x')] = \frac{1}{m}$
- 2-universal: $\forall x, x' \in [n] \; \forall y, y' \in [m]$ if $x \neq x'$, then $\Pr[h(x) = y \wedge h(x') = y'] = \frac{1}{m^2}$

Remarks:
- 2-universal => uniform
- 2-universal families of hash functions with small space and time consumption exist

Estimate # Distict Element

... from a stream $e_1, \dots, e_N \in [n]$

frequency vector of stream $f = (f_1, \dots, f_n)$ where $f_j = |\{i : e_i = j\}|$

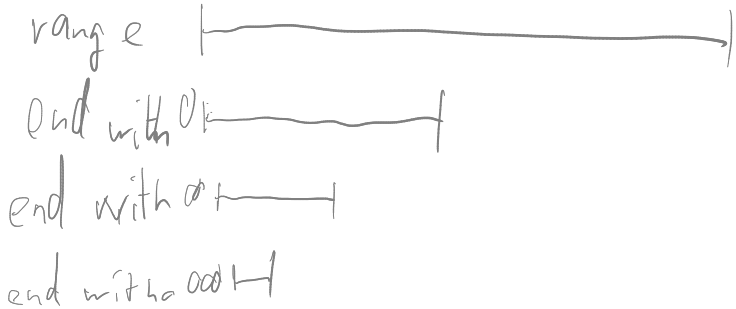
Goal: estimate $d := |\{j \in [n] : f_j > 0\}|$ =

Think representing numbers in binary

$\text{zeros}(p) = \max\{i : 2^i \text{ divides } p\}$

$= $ "number of 0s that p ends with

$p = 1101011000$



$O(\log n)$ space

Algorithm

Initialize: Choose random hash function $h : [n] \to [n]$ from 2-universal family

$z \leftarrow 0$

Process token e_j: if $\text{zeros}(h(e_j)) > z$

$z \leftarrow \text{zeros}(h(e_j))$

Output: $2^{z+\frac{1}{2}}$

← at most $\log n$ zeros in any hash

$O(\log \log n)$ space to store z

T.r.v. for final value of z

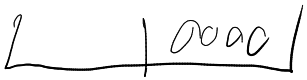
Analysis: for each $j \in [n]$, $r \geq 0$

random variables $X_{r,j} = \begin{cases} 1 & \text{if } \text{zeros}(h(j)) \geq r \\ 0 & \end{cases}$

$$Y_r = \sum_{j : f_j > 0} X_{r,j}$$

Thus: $Y_r > 0 \iff T \geq r$

$Y_r = 0 \iff T \leq r-1$

h uniform $\Rightarrow$ j $\Rightarrow$ 

fixed r, j

$$E[X_{r,j}] = \Pr[X_{r,j} = 1] = \Pr[\text{zeros}(h(j)) \geq r] = \frac{1}{2^r}$$

$$E[Y_r] = \sum_{j : f_j > 0} E[X_{r,j}] = \sum_{j : f_j > 0} \frac{1}{2^r} = \frac{d}{2^r}$$

By 2-universality $X_{r,j}$'s are independent

$$\text{Var}[Y_r] = \sum_{j : f_j > 0} \text{Var}[X_{r,j}] \leq \sum_{j : f_j > 0} E[X_{r,j}^2] = \sum_{j : f_j > 0} E[X_r] = \frac{d}{2^r}$$

$$\Pr[T \geq r] = \Pr[Y_r > 0] = \Pr[Y_r \geq 1] = \Pr\left[Y_r \geq \frac{2^r}{d} \cdot E[Y_r]\right] \leq \frac{d}{2^r}$$

$1 \leq \frac{1}{E[Y_r]} \cdot E[Y_r]$

↑ Markov Bound

$$\Pr[T < r] = \Pr[Y_r = 0] \leq \Pr\left[\left|Y_r - \frac{d}{2^r}\right| \geq \frac{d}{2^r}\right] \leq \frac{1}{\sqrt{\frac{d}{2^r}}^2} = \frac{2^r}{d}$$

$E[Y_r] = \frac{d}{2^r}$ $\quad \sqrt{\frac{d}{2^r}} \cdot \sqrt{\mathrm{Var}[Y_r]}$

Let $\hat{d}$ be estimate output by algorithm

$\rightarrow \hat{d} = 2^{T+\frac{1}{2}} = \sqrt{2} \cdot 2^T$

a: smallest integer s.t. $2^{a+\frac{1}{2}} \geq 3d$

$$\Pr[\hat{d} \geq 3d] = \Pr[T \geq a] \leq \frac{d}{2^a} = \frac{\sqrt{2}d}{2^{a+1/2}} \leq \frac{\sqrt{2}d}{3d} = \frac{\sqrt{2}}{3} < 0.5$$

b: largest integer s.t. $2^{b+1/2} \leq d/3$

$$\Pr\left[\hat{d} \leq \frac{d}{3}\right] = \Pr[T \leq b] = \Pr[T < b+1] \leq \frac{2^{b+1}}{d} = \frac{\sqrt{2} \cdot 2^{b+1/2}}{d} \leq \frac{\sqrt{2}\, d/3}{d} = \frac{\sqrt{2}}{3} < 0.5$$

$$\Pr\left[\frac{d}{3} < \hat{d} < 3d\right] = \Pr\left[\hat{d} > \frac{d}{3} \wedge \hat{d} < 3d\right] = 1 - \Pr\left[\hat{d} \leq \frac{d}{3} \vee \hat{d} \geq 3d\right]$$

$$= 1 - \left(\Pr\left[\hat{d} \leq \frac{d}{3}\right] + \Pr\left[\hat{d} \geq 3d\right]\right)$$

$$\leq 1 - 2 \cdot \frac{\sqrt{2}}{3} \approx 1 - \frac{2.828}{3} > 5\%$$

↳ Boost this to success probability $1-\delta$ by running $\Theta(1/\delta)$ instances of the algorithm and returning the median of the estimates (→ median trick)