جلسه سوم

خلاصه‌سازی برای مه‌داده

شمارش تقریبی

شمارش تقریبی

- صورت مساله:

  - `init();` sets $n \leftarrow 0$
  - `update();` increments $n \leftarrow n+1$
  - `query();` outputs n or an estimate of n

- حافظه؟

- کران پایین برای حافظه

- هدف:

  - تقریب n با حافظه کم‌تر
  - و $\mathbb{P}(|\tilde{n}-n| > \varepsilon n) < \delta,$

اندازه خطا

احتمال خطا

شمارش تقریبی

- صورت مساله:

  - `init();` sets $n \leftarrow 0$
  - `update();` increments $n \leftarrow n + 1$
  - `query();` outputs n or an estimate of n

- حافظه؟

- کران پایین برای حافظه

  برای الگوریتم همیشه درست: حداقل $\lg n$ حافظه

- هدف:

  - تقریب n با حافظه کم‌تر

  - و $\mathbb{P}(|\tilde{n} - n| > \varepsilon n) < \delta,$

الگوریتم موریس

- `init();` sets $X \leftarrow 0$
- `update();` increments X with probability 2^{-X}
- `query();` outputs $\tilde{n} = 2^X - 1$

کوئیزک:
پس از n بار اجرا چقدر احتمال دارد X=1 باشد؟

تحلیل الگوریتم موریس

- Xn: مقدار X پس از n بار به‌روزرسانی

Claim 5. *For Morris's algorithm,* $\mathbb{E}2^{X_n} = n + 1$.

اثبات

- استقرا

تحلیل الگوریتم موریس

- Xn: مقدار X پس از n بار به‌روزرسانی

Claim 5. *For Morris's algorithm,* $\mathbb{E}2^{X_n} = n + 1$.

اثبات

- استقرا

$$\mathbb{E}[2^{X_{n+1}}] = \sum_{j=0}^{\infty} P[X_n=j] \Big(\underbrace{2^{-j}\, 2^{j+1}}_{2} + \underbrace{(1-2^{-j})\, 2^{j}}_{2^j - 1} \Big)$$

$$= \underbrace{\sum_{j} P[X_n=j]}_{1} + \underbrace{\sum_{j=0}^{\infty} 2^{j} P[X_n=j]}_{n+1}$$

$$1 + n + 1$$

واریانس تخمین‌گر الگوریتم موریس

- یادآوری هدف:

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) < \delta,$$

- => باید واریانس را محاسبه کنیم

واریانس تخمین‌گر الگوریتم موریس

- یادآوری هدف:

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) < \delta,$$

$$< \frac{1}{(\varepsilon n)^2} \times \underbrace{E\left[(\tilde{n}-n)^2\right]}_{\underbrace{Var[\tilde{n}]}_{E[\tilde{n}^2] - (E[\tilde{n}])^2}}$$

- => باید واریانس را محاسبه کنیم

Claim 6. *For Morris's algorithm, we have*

$$\mathbb{E}2^{2X_n} = \frac{3}{2}n^2 + \frac{3}{2}n + 1. \tag{2.9}$$

اثبات

- استقرا

Claim 6. *For Morris's algorithm, we have*

$$\mathbb{E}2^{2X_n} = \frac{3}{2}n^2 + \frac{3}{2}n + 1. \tag{2.9}$$

اثبات

- استقرا

$$E[2^{2X_{n+1}}] = \sum_j E[2^{2X_{n+1}} \mid X_n = j] \cdot P[X_n = j]$$

$$= \sum_j P[X_n = j]\left(\underbrace{2^{2(j+1)} \cdot 2^{-j}}_{4\cdot 2^j} + \underbrace{(1-2^{-j}) \cdot 2^{2j}}_{2^{2j} - 2^j}\right)$$

$$= 3E\,2^{X_n} + E[2^{2X_n}]$$

تحلیل الگوریتم موریس، در نهایت

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) <$$

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) < \frac{1}{\varepsilon^2 n^2} \cdot \frac{n^2}{2} = \frac{1}{2\varepsilon^2},$$

تحلیل الگوریتم موریس، در نهایت

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) <$$

$$\frac{1}{(\varepsilon n)^2}\left(\frac{3}{2}n^2 + \frac{3}{2}n + 1 - (n+1)^2\right) < \frac{1}{2\varepsilon^2}$$

الگوریتم موریس+

چرا جواب‌های الگوریتم‌های موریس مستقل‌اند؟

الگوریتم موریس+

- اجرای s تا موریس، هر کدام جواب $\tilde{n}_i$
- جواب: $\tilde{n} = \frac{1}{s}\sum_{i=1}^{s} \tilde{n}_i$

تحلیل الگوریتم:

-

$$\mathbb{P}(|\tilde{n} - n| > \varepsilon n) < \frac{1}{2s\varepsilon^2} < \delta$$

for $s > 1/(2\varepsilon^2\delta) = \Theta(1/(\varepsilon^2\delta))$.

الگوریتم موریس+

چرا جواب‌های الگوریتم‌های موریس مستقل‌اند؟

الگوریتم موریس+

- اجرای s تا موریس، هر کدام جواب $\tilde{n}_i$
- جواب: $\tilde{n} = \frac{1}{s}\sum_{i=1}^{s}\tilde{n}_i$

تحلیل الگوریتم:

-

$$Var[\tilde{n}] = V[\frac{1}{s}\Sigma \cdot \cdot] = \frac{1}{s^2}\Sigma V[\tilde{n}_i]$$

$$= \frac{1}{s} Var[\tilde{n}_i]$$

-

الگوریتم موریس++

الگوریتم موریس++

- اجرای t تا موریس+، با $\delta = 1/3$
- جواب: میانه موریس+ها

تحلیل الگوریتم:

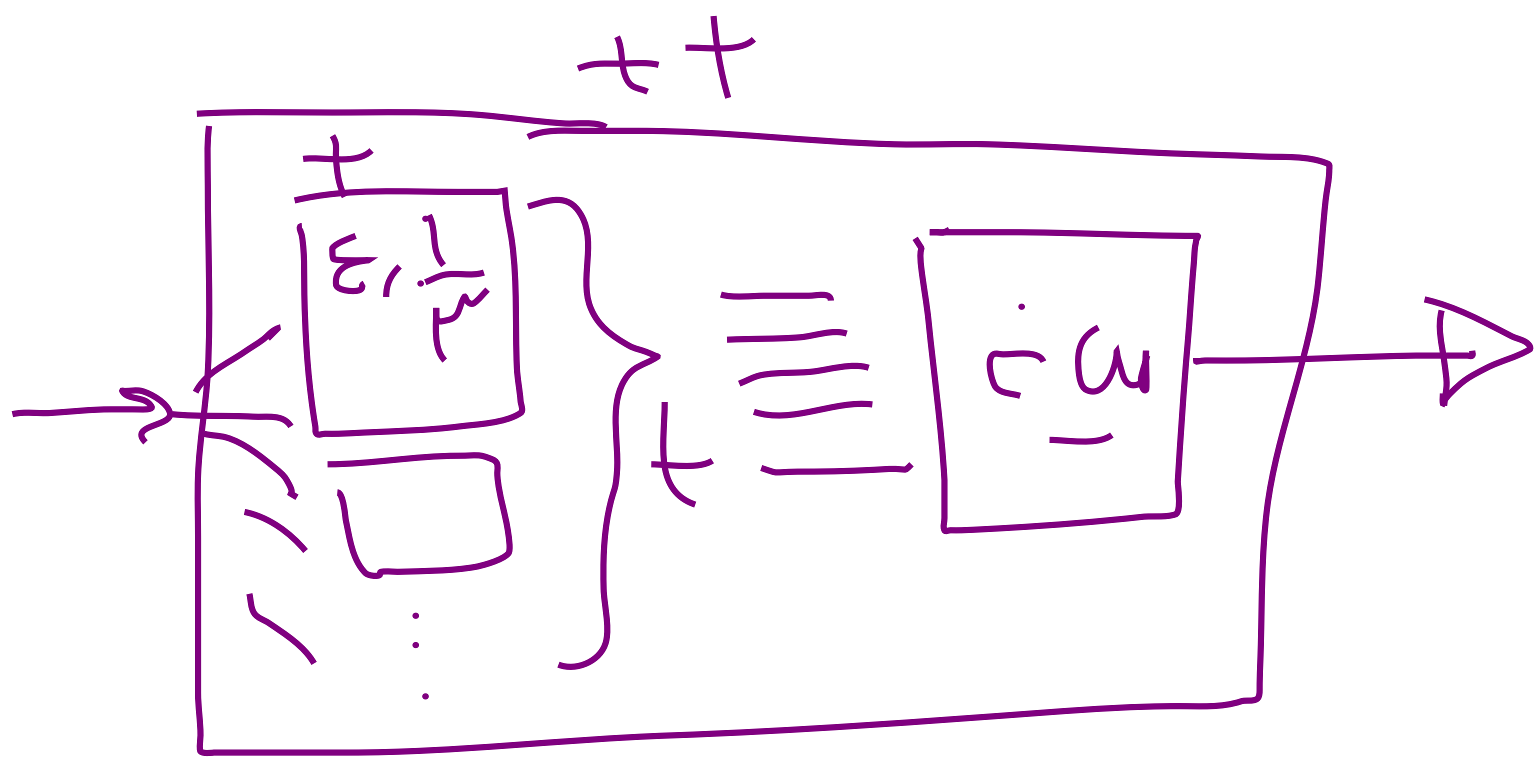
- کی میانه خوب است؟

for $t \in \Theta(\lg(1/\delta))$.

الگوریتم موریس++

الگوریتم موریس++

- اجرای t تا موریس+، با $\delta = 1/3$
- جواب: میانه موریس+ها



الگوریتم موریس++

الگوریتم موریس++

- اجرای t تا موریس+، با $\delta = 1/3$
- جواب: میانه موریس+ها

$E\sum Y_i - t/2 \geq \frac{1}{4} E\sum Y_i$

$E\sum Y_i \geq \frac{3}{4} t$

$E Y_i \geq \frac{3}{4}$

$\frac{2}{3}$

$n(1-\varepsilon)$ $n(1+\varepsilon)$

n

Y_i: موریس+ ام خوب

تحلیل الگوریتم:

- کی میانه خوب است؟

$$\text{احتمال اوضاع بد} \leq P\left[\sum Y_i \leq \frac{t}{2}\right]$$

$$= P\left[\sum Y_i - E\sum Y_i \leq \frac{t}{2} - E\sum Y_i\right]$$

$$= P\left[E\sum Y_i - \sum Y_i \geq E\sum Y_i - \frac{t}{2}\right]$$

$$\leq P\left[1 \geq \frac{3}{4}t - \frac{t}{2}\right] \leq 2 \cdot e^{-\left(\frac{1}{3}\right)^2 \frac{3}{4}t/3}$$

تحلیل الگوریتم موریس++، در نهایت

- موریس++:   تا موریس+
  - موریس+: \delta=⅓ پس s =   تا
  - موریس: بیت
- در مجموع:

تحلیل الگوریتم موریس++، در نهایت

- موریس++: $O(\lg \frac{1}{\delta})$ تا موریس+
  - موریس+: \delta=⅓ پس s = $\Theta(\frac{1}{\varepsilon^2})$ تا
  - موریس: $\lg\lg n$ بیت
- در مجموع:

$$\Theta\left(\lg \frac{1}{\delta} \cdot \frac{1}{\varepsilon^2} \cdot \lg\lg n\right)$$

تحلیل دقیق‌تر حافظه

- پس از X=a:
  - بزرگ‌تر شدن یک متغیر در یک درخواست
  - بزرگ‌تر شدن یک متغیر در n درخواست
  - بزرگ‌تر شدن حداقل یک متغیر از k متغیر در n درخواست

- با احتمال $1-\delta$ حافظه = $O(\varepsilon^{-2}\lg(1/\delta)(\lg\lg(n/(\varepsilon\delta))))$

وضعیت کنونی

- کران پایین $O(\lg(\lg_{1+\varepsilon} n)) = O(\lg(1/\varepsilon) + \lg\lg n)$

- الگوریتم بهتر برای کران پایین:
  - با احتمال $1/(1+a)^X$ اضافه کن
  - حافظه = $O(\log(1/\varepsilon) + \log\log n + \log\log(1/\delta))$