INF554: Utilisation de l'aléatoire en algorithmique **Automne 2014**

Cours 5 — 13 octobre

Enseignant : Frédéric Magniez *Rédacteur : Alexandros Hollender*

5.1 Algorithmes de streaming

5.1.1 Modèle et motivation

Définition 5.1 (Massive data). *L'entrée x (Data stream) est trop grosse pour être écrite en mémoire RAM (Random Access Memory). Sa taille est* $o(n)$ *(sous-linéaire), idéalement* $O(\log(n))$.

- *Nécessite un accès séquentiel à l'entrée : on lit x par morceaux (un bit, un entier, une lettre, ... un élément de taille fixée petite).*
- *Une seule passe ou plusieurs possibles selon les cas (mais toujours un nombre constant de passes).*
- *A la fin des passes, l'algorithme doit calculer ou approcher une fonction.*

Exemple: Analyse de l'ADN, du graphe du WEB, du trafic sur un routeur ... les données sont trop grandes pour être stockées en mémoire RAM. Nombre constant de passes.

Exemple (Missing number):

Stream: suite de n entiers distincts de $[[1; n+1]]$

Sortie: trouver l'entier manquant

Contrainte: mémoire en $O(\log(n))$ bits, 1 passe

Algorithme

$s \leftarrow 0$

Tant que Stream non vide

    $x \leftarrow$ lire Stream

    $s \leftarrow s + x$

retourner $\frac{(n+1)(n+2)}{2} - s$

Définition 5.2 (Paramètres d'un algorithme de streaming). *Un algorithme de streaming à* $p(n)$ *passes, mémoire* $s(n)$*, temps* $t(n)$*, lit une entrée* x *de taille* n *en :*

- *effectuant* $p(n)$ *passes sur l'entrée* x *vue comme un stream*
- *maintenant une mémoire interne (RAM) de* $s(n)$ *bits*
- *ayant une complexité en temps* $t(n)$ *après chaque symbole lu sur l'entrée* x.

paramètre	*valeur idéale*
nombre de passes $p(n)$	1 *ou* $O(1)$ *passes*
mémoire RAM $s(n)$	$O(\log^{O(1)}(n))$ *bits*
temps par morceaux $t(n)$	$O(\log^{O(1)}(n))$ *opérations*
temps de post-processing *(pour donner la réponse après passage du stream)*	$O(\log^{O(1)}(n))$ *opérations*

5.1.2 Moments de Fréquence

Définition 5.3.

Stream: $a_1, a_2, ..., a_n \in [[1, m]]$. *n est inconnu et m est connu*

Fréquences: $f_j = |\{i \in [[1; n]], a_i = j\}|, j \in [[1; m]]$

Moments d'ordre k: $F_k = \sum_{j=1}^{m} (f_j)^k$

- $F_0 = |\{j \in [[1; m]], f_j \neq 0\}|$ - *nombre d'éléments distincts*
- $F_1 = n$ - *nombre d'éléments*
- F_2 = *"repeat rate" ou "surprise index". F_2 grand $\Rightarrow$ f_j anormalement grand (possibilité d'attaque).*
- $F_\infty = \max_j f_j$
- $F_k = o(n^{1-2/k})(\log n + \log m)$

Définition 5.4. *μ est un $(\xi, \delta) -$ estimateur d'une valeur v si $\mathbb{P}[|\mu - v| > \xi|v|] \leq \delta$*

Remarque : En pratique, si on a un $(\xi, \delta) -$ *estimateur* avec $\delta < 1$ et $\xi < 1$ constantes, alors il est possible de construire un estimateur pour (ξ, δ) quelconque. Complexité typique : $O(\frac{1}{\delta}\log(\frac{1}{\xi}))$

Algorithme pour estimer F_1 :

Déterministe: espace en $O(\log(n))$ bits

Probabiliste: espace en $O(\log(\log(n)))$ bits

ALGORITHME

$a \leftarrow 0$

Tant que Stream non vide

    lire Stream

    $a \leftarrow a + 1$ avec probabilité $1/2^a$

retourner $2^a - 1$

Théorème 5.5. *Soit X_i la valeur de 2^a après i éléments ($X_0 = 1, X_1 = 2$, ...)*

$$\mathbb{E}(X_i) = i + 1 \tag{5.1}$$

Preuve: On a

$$\mathbb{E}(X_{i+1}) = \sum_{j=0}^{\infty} \mathbb{E}(X_{i+1}|X_i = 2^j)\mathbb{P}(X_i = 2^j)$$

mais

$$\mathbb{E}(X_{i+1}|X_i = 2^j) = \frac{1}{2^j}2^{j+1} + (1 - \frac{1}{2^j})2^j = 2 + 2^j - 1 = 2^j + 1$$

Donc

$$\begin{aligned} \mathbb{E}(X_{i+1}) &= \sum_{j=0}^{\infty}(2^j + 1)\mathbb{P}(X_i = 2^j) \\ &= \sum_{j=0}^{\infty} 2^j \mathbb{P}(X_i = 2^j) + \sum_{j=0}^{\infty} \mathbb{P}(X_i = 2^j) \\ &= \mathbb{E}(X_i) + 1 \end{aligned}$$

□

Inégalité de Markov

Soit X variable aléatoire positive, et soit $\mu = \mathbb{E}(X)$. $\forall a > 0, \mathbb{P}(X \geq a\mu) \leq 1/a$.
Application à notre problème : $\mathbb{P}(v \geq 2n) \leq 1/2$

Inégalité de Tchebychev

Soit X variable aléatoire telle que

$$\begin{aligned} \mathbb{E}(X) &= \mu \\ \mathrm{Var}(X) &= \mathbb{E}(X^2) - (\mathbb{E}(X))^2 = \sigma^2 \end{aligned}$$

Alors

$$\forall a > 0, \mathbb{P}(|X - \mu| > a\sigma) < 1/a^2 \tag{5.2}$$

Dans notre problème, si l'on veut $\mathbb{P}(|v - n| \geq \frac{n}{2}) \leq \frac{1}{4}$ donc $\sigma = \frac{n}{4} \Rightarrow \sigma^2 = \frac{n^2}{16}$, ce qui n'est pas très mauvais. Nous cherchons donc les petites variances

Lemme:
$\mathrm{Var}(v) = \frac{n(n+1)}{2} \leq \frac{(\mathbb{E}(v))^2}{2}$

Ce n'est pas très bon... Il faudrait diminuer la variance, et pour ce faire on utilisera l'astuce suivante.

Mean trick

ALGORITHME

$c_t \leftarrow 0$, $t = 1, 2, \ldots, k$, k paramètre.

tant que Stream non vide

lire élément suivant

$\forall t$, avec probabilité $\frac{1}{2^{c_t}}$: $c_t \leftarrow c_t + 1$

retourner la moyenne w des $v_t = 2^{c_t} - 1$

Remarque:

$\mathbb{E}(w) = \mathbb{E}(v_t) = n$

$\mathrm{Var}(w) = \frac{1}{k^2} k \,\mathrm{Var}(v_t) = \frac{1}{k}\,\mathrm{Var}(v_t) \leq \frac{1}{2k}(\mathbb{E}(v_t))^2 \leq \frac{n^2}{2k}$

Remarque:

Soit $\epsilon > 0$. Si $a = \epsilon\sqrt{2k}$, alors

$\mathbb{P}(|w - n| \geq \epsilon n) = \mathbb{P}(|w - n| \geq \sqrt{\frac{n^2}{2k}}a) \leq \mathbb{P}(|w - n| \geq \sigma a)$, cette dernière inégalité, parce qu'on autorise davantage d'évènements puisque σ est plus petit que l'autre quantité.

$\leq \frac{1}{a^2} = \frac{1}{2\epsilon^2 k}$ d'après Markov

$\leq 1/4$ si $k = \frac{2}{\epsilon^2}$

On a obtenu ainsi que si $k = \frac{2}{\epsilon^2}$, $\mathbb{P}(|w - n| \geq \epsilon n) \leq 1/4$.
Maintenant on voudrait $\mathbb{P}(|w - n| \geq \epsilon n) \leq \delta$, quel que soit $\delta > 0$.

Median trick

$v_1, v_2, ..., v_l$ des $(\epsilon, 1/4)$ estimateurs indépendants (i.e. $\forall i, \mathbb{P}(|v_i - \mu| \geq \epsilon\mu) \leq 1/4$). $\mathbb{E}(v_i) = \mu$. Soit w leur médiane. Alors, w satisfait

$$\mathbb{P}(|w - \mu| \geq \epsilon\mu) \leq e^{-l/24} \tag{5.3}$$

Pour nous, si $l \sim \log(1/\delta)$ alors $\mathbb{P}(|w - \mu| \geq \epsilon\mu) \leq \delta$.

Théorème 5.6. *Il existe un (ϵ, δ) estimateur de F_1 en*
- *1 passe*
- *mémoire* $\mathrm{O}(\frac{\log(1/\delta)}{\epsilon^2}\log(\log(n)))$

5.1.3 Estimer F_0

Définition 5.7 (2-universal family).
$\exists H \subseteq \{h : [[1;m]] \to [[1;M]]\}$ *tel que* $\left\{ \begin{array}{r} \forall x \neq y \in [[1;m]] \\ \forall u, v \in [[1;M]] \end{array} \right. , \mathop{\mathbb{P}}_{h} \left(\left\{ \begin{array}{l} h(x)=u \\ h(y)=v \end{array} \right. \right) = 1/M^2$

Si les valeurs sont uniformément réparties, $(\min_i(a_i)) = m/F_0$

Conséquences: $\forall x \in [[1;m]], \forall u \in [[1;M]], \mathbb{P}(h(x) = u) = 1/M$

interprétation: Si h uniformément choisi dans H
- $\forall x \in [[1;m]], h(x)$ uniformément réparti sur $[[1;M]]$
- $\forall x \neq y \in [[1;m]]$, $h(x)$ et $h(y)$ indépendants

Théorème 5.8 (Construction). *Soit $m \leq p < 2m$ premier. $M = p$*

$$\forall a, b \in [[0;p-1]], h_{a,b} : \begin{array}{rcl} [[1;m]] & \to & [[1;p]] \\ x & \mapsto & a \cdot x + b \mod p \end{array} \tag{5.4}$$

$\{h_{a,b}, a, b \in [[0;p-1]]\}$ *est une 2-universal family*

Preuve: $\left\{ \begin{array}{r} \forall x \neq y \in [[1;m]] \\ \forall u, v \in [[1;M]] \end{array} \right.$ $\exists! a, b \in [[0;p-1]]$ tq $\left\{ \begin{array}{l} a \cdot x + b = u \mod p \\ a \cdot y + b = v \mod p \end{array} \right.$ (système d'équations linéaires non dégénéré, car $x \neq y$)
Par conséquent $\mathop{\mathbb{P}}_{a,b} \left(\left\{ \begin{array}{l} h_{a,b}(x)=u \\ h_{a,b}(y)=v \end{array} \right. \right) = 1/p^2$ □

Algorithme MinHash

$m \leq p < 2m$ premier, $min \leftarrow p$
$a, b \in [[0;p-1]]$
Tant que Stream non vide
    $x \leftarrow$ lire Stream
    $min \leftarrow \min\{a \cdot x + b \mod p; min\}$
retourner p/min

---

analyse:
- 1 passe
- mémoire en $O(\log(m))$ bit
- temps par élément en $O(1)$ opérations arithmétiques
- temps post-processing en $O(1)$ opérations arithmétiques

Théorème 5.9.

$$\mathbb{P}(F_0/6 \leq p/min \leq 6F_0) \geq 2/3 \tag{5.5}$$

Améliorable avec les même techniques que F_1

Preuve:

$$\begin{aligned}
\mathbb{P}(p/min > 6F_0) &= \mathbb{P}(\exists k, h(a_k) < \frac{p}{6F_0}) \\
&\leq \sum_k \mathbb{P}(h(a_k) < \frac{p}{6F_0}) \\
&\leq F_0 \max_k \mathbb{P}(h(a_k) < \frac{p}{6F_0}) \\
&\leq F_0 \frac{p}{6F_0}\frac{1}{p} \\
&\leq 1/6
\end{aligned}$$

$$\mathbb{P}(p/min < F_0/6) = \mathbb{P}(\forall k, h(a_k) > \frac{6p}{F_0})$$

Soit $Y_k = \begin{cases} 1 \text{ si } h(a_k) \leq \frac{6p}{F_0} \\ 0 \text{ sinon} \end{cases}$ et $Y = \sum_k Y_k$

On a $\mathbb{E}(Y_k) = \frac{6}{F_0}$ et $\mathrm{Var}(Y_k) = \frac{6}{F_0}(1 - \frac{6}{F_0})$

Donc $\mathbb{E}(Y) = 6$ et $\mathrm{Var}(Y) = 6(1 - \frac{6}{F_0}) < 6$

Finalement

$$\begin{aligned}
\mathbb{P}(p/min < F_0/6) &= \mathbb{P}(Y = 0) \\
&\leq \mathbb{P}(|Y - \mathbb{E}(Y)| \geq 6) \\
&\leq 1/6
\end{aligned}$$

□

Bilan pour l'estimateur de F_k :

k	Mémoire
0	$O(\log m)$
1	$O(\log \log n)$
2	$O(\log n + \log m)$
> 2	$O(n^{1-2/k}(\log n + \log m))$