Auxiliar #7 - Algoritmos aleatorizados y/o probabilistas

30 de noviembre de 2020 - Bernardo Subercaseaux

Problema 1. (★★) Dadas tres matrices A, B y C, todas de $n \times n$ a coeficientes reales, se desea chequear si $AB = C$. El algoritmo ingenuo requiere computar AB en tiempo $O(n^3)$, y comparar el resultado con C. Diseñe un algoritmo aleatorizado de complejidad $O(n^2)$.

Solución 1 . Lo que haremos será samplear un vector aleatorio x de tamaño $\{0,1\}^n$, y luego ver si se cumple que $A(Bx) = Cx$, en cuyo caso respondemos afirmativamente que $AB = C$, y no en caso contrario. Notemos primero que calcular el producto Bx toma tiempo $O(n^2)$, y resulta en un vector de tamaño n, por lo que luego multiplicarlo con A también toma tiempo $O(n^2)$. Así, el chequeo toma tiempo $O(n^2)$. Si el chequeo falla sabemos que $A(Bx) \neq Cx$, pero $(AB)x = A(Bx)$, así que AB debe ser distinto de C. Eso quiere decir que si el chequeo falla, es seguro responder que $AB \neq C$. Si el chequeo funciona responderemos afirmativamente al problema, sin embargo, no podemos garantizar que $AB = C$. La probabilidad de error está dada por el caso en que $AB \neq C$ y sin embargo $A(Bx) = Cx$. ¿Cuál es la probabilidad de que esto ocurra? Consideremos $D = C - AB$ y $P = Dx$. Debe haber una componente $D_{ij} \neq 0$. La probabilidad de que $P_i = 0$ está dada por:

$$
\begin{aligned}
Pr(P_i = 0) =& Pr\left(\sum_{k=1}^{n} D_{ik}x_k = 0\right) && (1)\\
=& Pr\left(D_{ij}x_j + R = 0\right) && (2)\\
=& Pr\left(D_{ij}x_j = 0\right) Pr(R = 0) + Pr\left(D_{ij}x_j + R = 0 \mid R \neq 0\right) Pr(R \neq 0) && (3)\\
=& Pr\left(x_j = 0\right) Pr(R = 0) + Pr\left(x_j = 1 \wedge R = -D_{ij}\right) Pr(R \neq 0) && (4)\\
\leq& Pr\left(x_j = 0\right) Pr(R = 0) + Pr\left(x_j = 1\right) Pr(R \neq 0) && (5)\\
=& \frac{1}{2} Pr(R = 0) + \frac{1}{2} Pr(R \neq 0) = 1/2 && (6)
\end{aligned}
$$

Repetir el chequeo k veces mantiene la complejidad en $O(n^2)$, pero baja la probabilidad de error a $(\frac{1}{2})^k$.

Problema 2. (★★★★) Suponga que tenemos una tabla de hash con n registros (*slots*), y que la utilizamos para insertar n llaves resolviendo los conflictos por encadenamiento. Cada llave tiene la misma probabilidad de ser insertada en cada registro. Sea M el número máximo de llaves en algún registro después de que todas las llaves han sido insertadas.

1. ¿Cuál es la probabilidad Q_k de que exactamente k llaves sean insertadas en un registro dado?
2. Demuestre que $Q_k < (e/k)^k$. (Recuerde que $k! \geq (k/e)^k$ debido a la aproximación de Stirling).
3. Sea P_k la probabilidad de que $M = k$, es decir, de que el registro que contiene más llaves contenga exactamente k llaves. Demuestre que $P_k \leq nQ_k$.

4. Demuestre que existe una constante $c > 1$ tal que $Q_{k_0} < 1/n^3$, donde $k_0 = c \ln n / \ln\ln n$. Concluya que $P_k < 1/n^2$ para todo $k \geq k_0$.

5. Explique por qué se cumple que el valor esperado $E(M)$ de M está acotado por la siguiente expresión
$$Pr(M > k_0) \cdot n \; + \; Pr(M \leq k_0) \cdot k_0.$$
Concluya que $E(M) = O(\ln n / \ln\ln n)$.

Hint: Exprese $E(M)$ como $\sum_{i=1}^{k_0} i \cdot Pr(M = i) + \sum_{i=k_0+1}^{n} i \cdot Pr(M = i)$.

Solución 2 .

1. La probabilidad de que una llave dada quede en un registro dado es $1/n$. Hay $\binom{n}{k}$ formas de elegir las k llaves a insertar en el registro dado. Una vez elegidas, requerimos que aquellas k llaves efectivamente se inserten en el registro dado, y las $n-k$ restantes no. Esto último ocurre con probabilidad $(\frac{1}{n})^k(1-\frac{1}{n})^{n-k}$. En total, considerando que podría ser cualquier conjunto de k llaves, $Q_k = \binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k}$.

2. Dado que $(1-\frac{1}{n}) < 1$, se cumple que $Q_k < \binom{n}{k}(\frac{1}{n})^k$. Recordando que $\binom{n}{k} = \frac{n!}{k!(n-k)!} < n^k/k!$ obtenemos que $Q_k < 1/k!$. Luego, la aproximación de Stirling nos dice que $k! > (k/e)^k$, de donde $Q_k < (e/k)^k$.

3. Sea e_i el evento en que el registro i tiene k llaves y es además el que más llaves tiene. Sabemos entonces que $Pr(e_i) \leq Q_k$. Además, $M = k$ significa que alguno de los e_i se cumple, así que $Pr(M = k) = Pr(\cup_i e_i) \leq \sum_i Pr(e_i) = nPr(e_i) \leq nQ_k$.

4. Basta un c tal que $(e/k_0)^{k_0} < 1/n^3$. Esto último es equivalente, tomando logaritmos, a
$$c\frac{\ln n}{\ln\ln n}(1 - \ln\frac{c\ln n}{\ln\ln n}) < -3\ln n$$
que es a su vez equivalente a
$$\frac{c}{\ln\ln n}(\ln c + \ln\ln n - \ln\ln\ln n - 1) > 3$$
Si forzamos $c > e$, se cumple que $\ln c - 1 > 0$, y por lo tanto nos basta que además
$$c\left(1 - \frac{\ln\ln\ln n}{\ln\ln n}\right) > 3 \iff c > \frac{3}{1 - \frac{\ln\ln\ln n}{\ln\ln n}}$$
La expresión $\frac{\ln\ln\ln n}{\ln\ln n}$ es siempre menor o igual que $1/2$ (esto se puede ver estudiando el máximo de $f(x) = \frac{\ln x}{x}$). Por lo que cualquier $c > 6$ satisface la restricción. Dado que $Q_{k_0} < 1/n^3$, y usando que $(e/k)^k$ es decreciente para $k > e$ y que $k_0 > e$, tenemos que $Q_k < 1/n^3$ para todo $k \geq k_0$ y por el apartado anterior obtenemos que $P_k < 1/n^2$ para todo $k \geq k_0$.

5. Usando el hint:

$$
\begin{aligned}
E(M) &= \sum_{i=1}^{k_0} i \cdot Pr(M=i) + \sum_{i=k_0+1}^{n} i \cdot Pr(M=1) && (7)\\
&\leq k_0 \sum_{i=1}^{k_0} Pr(M=i) + n \sum_{i=k_0+1}^{n} Pr(M=1) && (8)\\
&= k_0 \cdot Pr(M \leq k_0) + n \cdot Pr(M > k_0) && (9)\\
&\leq k_0 + n \cdot (n-k_0) \cdot 1/n^2 && (10)\\
&= k_0 + 1 = 1 + c\frac{\ln n}{\ln\ln n} = O\left(\frac{\ln n}{\ln\ln n}\right). && (11)
\end{aligned}
$$