Algoritmos aleatorizados

IIC2283

Algoritmos aleatorizados

Vamos a permitir a los algoritmos tener una componente aleatoria

- En general esto significa que un algoritmo toma algunas decisiones dependiendo de valores escogidos al azar (según una distribución de probabilidades)

Hablamos entonces de algoritmos aleatorizados

Algoritmos aleatorizados

La ejecución de un algoritmo aleatorizado depende entonces de valores escogidos al azar

- Distintas ejecuciones pueden dar resultados distintos

Vamos a considerar dos tipos de algoritmos aleatorizados:

- **Monte Carlo:** el algoritmo siempre entrega un resultado, pero hay una probabilidad de que sea incorrecto
- **Las Vegas:** si el algoritmo entrega un resultado es correcto, pero hay una probabilidad de que no entregue resultado

¿Cuáles son las ventajas de los algoritmos aleatorizados?

Existen problemas para los cuales los algoritmos aleatorizados son más eficientes que los algoritmos usuales (sin una componente aleatoria)

- Por ejemplo, el problema de verificar si un número es primo

Existen problemas para los cuales los únicos algoritmos eficientes conocidos son aleatorizados

- Por ejemplo, el problema de verificar si dos polinomios en varias variables son equivalentes

Vamos a ver en detalle estos ejemplos . . .

Un primer ejemplo: equivalencia de polinomios

Consideramos polinomios en $\mathbb{Q}$

Suponemos inicialmente que un polinomio es una expresión de la forma:

$$p(x) \quad = \quad \sum_{i=1}^{k} \prod_{j=1}^{\ell_i} (x + a_{i,j})$$

donde cada $a_{i,j} \in \mathbb{Q}$

La forma canónica de $p(x)$ es una expresión de la forma:

$$p(x) \quad = \quad \sum_{i=0}^{\ell} c_i x^i$$

donde cada $c_i \in \mathbb{Q}$ y $\ell = \text{máx}\{\ell_1, \ldots, \ell_k\}$

El grado de $p(x)$ es ℓ.

Un primer ejemplo: equivalencia de polinomios

Evaluamos los polinomios sobre elementos de $\mathbb{Q}$

- La operación más cara en este proceso es la multiplicación de números en $\mathbb{Q}$, por eso nos concentramos en ella

Ejercicio

1. De un algoritmo que, dado un polinomio $p(x)$ y $a \in \mathbb{Q}$, calcule $p(a)$
2. De un algoritmo que, dado un polinomio $p(x)$ en su forma canónica y $a \in \mathbb{Q}$, calcule $p(a)$

Los dos algoritmos deben realizar $O(|p(x)|)$ multiplicaciones, donde $|p(x)|$ es el largo de $p(x)$ considerado como una palabra sobre un cierto alfabeto.

Un primer ejemplo: equivalencia de polinomios

Dados dos polinomios $p(x)$ y $q(x)$, queremos verificar si son idénticos.

- Para cada $a \in \mathbb{Q}$, se tiene que $p(a) = q(a)$

¿Cómo podemos resolver este problema?

Un algoritmo para la equivalencia de polinomios

EquivPol($p(x)$, $q(x)$)
- determine el grado k de $p(x)$
- determine el grado ℓ de $q(x)$
- **if** $k \neq \ell$ **then return** no
- **else**
  - transforme $p(x)$ es su forma canónica $\sum_{i=0}^{k} c_i x^i$
  - transforme $q(x)$ es su forma canónica $\sum_{i=0}^{k} d_i x^i$
  - **for** $i := 0$ **to** k **do**
    - **if** $c_i \neq d_i$ **then return** no
  - **return** sí

Un algoritmo para la equivalencia de polinomios

Ejercicio

Muestre que el algoritmo anterior en el peor caso es $O(n^2)$, donde $n = |p(x)| + |q(x)|$

- Recuerde que la operación básica a contar es la multiplicación de números racionales

¿Es posible resolver este problema utilizando un menor número de multiplicaciones?

Un algoritmo aleatorizado para la equivalencia de polinomios

EquivPolAleatorizado($p(x)$, $q(x)$)
    determine el grado k de $p(x)$
    determine el grado ℓ de $q(x)$
    if $k \neq \ell$ **then return** no
    else
        escoja al azar y con distribución uniforme un elemento a
                del conjunto de números naturales $\{1, ..., 100 \cdot k\}$
        if $p(a) = q(a)$ **then return** sí
        else return no

Un algoritmo aleatorizado para la equivalencia de polinomios

El algoritmo sólo necesita realizar $O(n)$ multiplicaciones, donde $n = |p(x)| + |q(x)|$

- Ya que necesita calcular $p(a)$ y $q(a)$

Pero el algoritmo puede dar una respuesta equivocada

- ¿Cuál es la probabilidad de error?

Calculando la probabilidad de error

Sean $p(x)$ y $q(x)$ dos polinomios de grado k

- Si los polinomios $p(x)$ y $q(x)$ son equivalentes, entonces el algoritmo responde **sí** sin cometer error

- Si los polinomios $p(x)$ y $q(x)$ no son equivalentes, el algoritmo puede responder **sí** al sacar al azar un elemento $a \in \{1, \ldots, 100 \cdot k\}$ tal que $p(a) = q(a)$

  Esto significa que a es una raíz del polinomio $r(x) = p(x) - q(x)$

Calculando la probabilidad de error

$r(x)$ no es el polinomio nulo y es de grado a lo más k

- Por lo tanto $r(x)$ tiene a lo más k raíces en $\mathbb{Q}$

Concluimos que:

$$\begin{aligned} \Pr(a \text{ sea una raíz de } r(x)) &\leq \frac{k}{100 \cdot k} \\ &= \frac{1}{100} \end{aligned}$$

Un mejor algoritmo aleatorizado

La probabilidad de error del algoritmo está acotada por $\frac{1}{100}$

- ¿Es aceptable esta probabilidad?

Ejercicio

De un algoritmo que resuelva el problema de equivalencia de polinomios, que en el peor caso sea $O(n)$ y que tenga una probabilidad de error acotada por $\frac{1}{100^{10}}$

¿Confiaría en este algoritmo lineal?

- ¿Para qué probabilidad estaría dispuesto a confiar?

Un segundo ejemplo: una definición general de polinomios

Consideramos polinomios en varias variables en $\mathbb{Q}$

Un monomio es una expresión de la forma $cx_1^{\ell_1} \cdots x_n^{\ell_n}$, donde $c \in \mathbb{Q}$ y cada $\ell_i \in \mathbb{N}$

Un monomio $cx_1^{\ell_1} \cdots x_n^{\ell_n}$ es nulo si $c = 0$

- No es nulo si $c \neq 0$

El grado de un monomio $cx_1^{\ell_1} \cdots x_n^{\ell_n}$ no nulo es $\ell_1 + \cdots + \ell_n$

Un segundo ejemplo: una definición general de polinomios

Un polinomio es una expresión de la forma:

$$p(x_1, \ldots, x_n) \quad = \quad \sum_{i=1}^{\ell} \prod_{j=1}^{m_i} \left(\sum_{k=1}^{n} a_{i,j,k} x_k + a_{i,j,n+1} \right)$$

donde cada $a_{i,j,k} \in \mathbb{Q}$ y cada $a_{i,j,n+1} \in \mathbb{Q}$

Un segundo ejemplo: una definición general de polinomios

La forma canónica de un polinomio $p(x_1, \ldots, x_n)$ es única, y es igual a 0 o a una suma de monomios que satisface las siguiente propiedades:

- cada monomio en la forma canónica es de la forma $cx_1^{\ell_1} \cdots x_n^{\ell_n}$ con $c \neq 0$
- si $cx_1^{\ell_1} \cdots x_n^{\ell_n}$ y $dx_1^{m_1} \cdots x_n^{m_n}$ son dos monomios distintos en la forma canónica, entonces $\ell_i \neq m_i$ para algún $i \in \{1, \ldots, n\}$

Un polinomio $p(x_1, \ldots, x_n)$ es nulo si su forma canónica es 0

El grado de un polinomio $p(x_1, \ldots, x_n)$ no nulo es el mayor grado de los monomios en su forma canónica.

Equivalencia de polinomios en varias variables

Dos polinomios $p(x_1, \ldots, x_n)$ y $q(x_1, \ldots, x_n)$ son idénticos si para cada secuencia $a_1, \ldots, a_n \in \mathbb{Q}$ se tiene que:

$$p(a_1, \ldots, a_n) \quad = \quad q(a_1, \ldots, a_n)$$

Nuevamente queremos verificar si dos polinomios son idénticos.

Equivalencia de polinomios en varias variables

¿Podemos verificar en tiempo polinomial si dos polinomios en varias variables son equivalentes?

- Consideramos ahora todas las operaciones, no sólo la multiplicación de números racionales

Tenemos un problema: calcular la forma canónica de un polinomio toma tiempo exponencial

- Incluso si sólo consideramos la multiplicación de números racionales

Pero existe un algoritmo aleatorizado eficiente para este problema.

- Esto no es trivial ya que un polinomio $p(x_1, \ldots, x_n)$ puede tener una cantidad infinita de raíces
  - Por ejemplo: $p(x_1, x_2) = (x_1 - 1)(x_2 - 3)$
- El ingrediente principal es el lema de Schwartz-Zippel

El ingrediente principal

Lema de Schwartz-Zippel

Sea $p(x_1, \ldots, x_n)$ un polinomio no nulo de grado k, y sea A un subconjunto finito y no vacío de $\mathbb{Q}$. Si $a_1, \ldots, a_n$ son elegidos de manera uniforme e independiente desde A, entonces

$$\Pr(p(a_1, \ldots, a_n) = 0) \quad \leq \quad \frac{k}{|A|}$$

Demostración: Por inducción en n

Ya habíamos considerado el caso $n = 1$

Suponemos que la propiedad se cumple para todo polinomio en n variables, y consideramos un polinomio $p(x_1, x_2, \ldots, x_{n+1})$ de grado k

Demostración del lema de Schwartz-Zippel

Si $p(x_1, x_2, \ldots, x_{n+1})$ en su forma canónica es igual a $c \in (\mathbb{Q} \setminus \{0\})$, entonces el lema se cumple trivialmente ya que

$$\Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0) \quad = \quad 0$$

Suponemos entonces que $p(x_1, x_2, \ldots, x_{n+1})$ en su forma canónica no es igual a $c \in \mathbb{Q}$.

- Puesto que además sabemos que $p(x_1, x_2, \ldots, x_{n+1})$ no es nulo

Demostración del lema de Schwartz-Zippel

Tenemos que $p(x_1, x_2, \ldots, x_{n+1})$ en su forma canónica contiene un monomio de la forma:

$$c x_1^{\ell_1} x_2^{\ell_2} \cdots x_{n+1}^{\ell_{n+1}}$$

donde $c \neq 0$ y $\ell_i > 0$ para algún $i \in \{1, \ldots, n+1\}$

Sin perdida de generalidad suponemos que en el monomio anterior $\ell_1 > 0$

Tenemos que:

$$p(x_1, x_2, \ldots, x_{n+1}) = \sum_{i=0}^{k} x_1^i p_i(x_2, \ldots, x_{n+1})$$

donde cada $p_i(x_2, \ldots, x_{n+1})$ es un polinomio y al menos uno de ellos no es nulo

Demostración del lema de Schwartz-Zippel

Sea $\ell = \text{máx}\{i \in \{0, \ldots, k\} \mid p_i(x_2, \ldots, x_{n+1}) \text{ no es nulo}\}$

- Tenemos que $\ell > 0$ ya que supusimos que $\ell_1 > 0$

Dado que el grado de $p(x_1, x_2, \ldots, x_{n+1})$ es k, tenemos que el grado de $p_\ell(x_2, \ldots, x_{n+1})$ es m con $m \leq k - \ell$

Sea A un subconjunto finito y no vacío de $\mathbb{Q}$, y sea $a_1, \ldots, a_{n+1}$ una secuencia de números elegidos de manera uniforme e independiente desde A

Demostración del lema de Schwartz-Zippel

Por hipótesis de inducción tenemos que:

$$\begin{aligned} \Pr(p_\ell(a_2, \ldots, a_{n+1}) = 0) &\leq \frac{m}{|A|} \\ &\leq \frac{k-\ell}{|A|} \end{aligned}$$

Si $p_\ell(a_2, \ldots, a_{n+1}) \neq 0$, entonces por definición de ℓ tenemos que $q(x_1) = p(x_1, a_2, \ldots, a_{n+1})$ es un polinomio de grado ℓ.

Por lo tanto:

$$\Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0 \mid p_\ell(a_2, \ldots, a_{n+1}) \neq 0) \leq \frac{\ell}{|A|}$$

Demostración del lema de Schwartz-Zippel

Concluimos que:

$$
\begin{aligned}
&\Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0) \; = \\
&\quad \Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0 \mid p_\ell(a_2, \ldots, a_{n+1}) = 0) \cdot \Pr(p_\ell(a_2, \ldots, a_{n+1}) = 0) \; + \\
&\quad \Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0 \mid p_\ell(a_2, \ldots, a_{n+1}) \neq 0) \cdot \Pr(p_\ell(a_2, \ldots, a_{n+1}) \neq 0) \; \leq \\
&\quad \Pr(p_\ell(a_2, \ldots, a_{n+1}) = 0) + \Pr(p(a_1, a_2, \ldots, a_{n+1}) = 0 \mid p_\ell(a_2, \ldots, a_{n+1}) \neq 0) \; \leq \\
&\quad \frac{k - \ell}{|A|} + \frac{\ell}{|A|} \; = \; \frac{k}{|A|}
\end{aligned}
$$

□

Un algoritmo aleatorizado para la equivalencia de polinomios en varias variables

Vamos a dar un algoritmo aleatorizado eficiente para el problema de verificar si dos polinomios en varias variables son equivalentes

Suponga que la entrada del algoritmo está dada por los siguientes polinomios:

$$p(x_1, \ldots, x_n) = \sum_{i=1}^{\ell} \prod_{j=1}^{m_i} \left(\sum_{k=1}^{n} a_{i,j,k} x_k + a_{i,j,n+1} \right)$$

$$q(x_1, \ldots, x_n) = \sum_{i=1}^{r} \prod_{j=1}^{s_i} \left(\sum_{k=1}^{n} b_{i,j,k} x_k + b_{i,j,n+1} \right)$$

Un algoritmo aleatorizado para la equivalencia de polinomios en varias variables

EquivPolAleatorizado$(p(x_1, \ldots, x_n),\ q(x_1, \ldots, x_n))$

$k := \max\{m_1, \ldots, m_\ell, s_1, \ldots, s_r\}$

$A := \{1, ..., 100 \cdot k\}$

sea $a_1, \ldots, a_n$ una secuencia de números elegidos de manera uniforme e independiente desde A

if $p(a_1, \ldots, a_n) = q(a_1, \ldots, a_n)$ **then return** sí

else return no

Utilizando el lema de Schwartz-Zippel

Antes de analizar la complejidad del algoritmo vamos a calcular su probabilidad de error.

- Si los polinomios $p(x_1, \ldots, x_n)$ y $q(x_1, \ldots, x_n)$ son equivalentes, entonces el algoritmo responde **sí** sin cometer error

- Si los polinomios $p(x_1, \ldots, x_n)$ y $q(x_1, \ldots, x_n)$ no son equivalentes, el algoritmo puede responder **sí** al escoger una secuencia de números $a_1, \ldots, a_n$ desde A tales que $p(a_1, \ldots, a_n) = q(a_1, \ldots, a_n)$
  - Donde $A = \{1, \ldots, 100 \cdot k\}$

  Esto significa que $(a_1, \ldots, a_n)$ es una raíz del polinomio $r(x_1, \ldots, x_n) = p(x_1, \ldots, x_n) - q(x_1, \ldots, x_n)$

Utilizando el lema de Schwartz-Zippel

$r(x_1, \ldots, x_n)$ no es el polinomio nulo y es de grado t con $t \leq k$

- Dado que $k = \text{máx}\{m_1, \ldots, m_\ell, s_1, \ldots, s_r\}$

Utilizando el lema de Schwartz-Zippel obtenemos:

$$\Pr(r(a_1, \ldots, a_n) = 0) \leq \frac{t}{|A|} \leq \frac{k}{|A|} = \frac{k}{100 \cdot k} = \frac{1}{100}$$

La probabilidad de error del algoritmo está entonces acotada por $\frac{1}{100}$

Un mejor algoritmo aleatorizado para el problema general

Ejercicio

De un algoritmo aleatorizado que resuelva el problema de equivalencia de polinomios en varias variables.

- La probabilidad de error del algoritmo debe estar acotada por $\frac{1}{100^{10}}$
- Debe existir una constante c tal que el algoritmo en el peor caso es $O(m^c)$, donde m es el tamaño de la entrada
  - Si consideramos $p(x_1, \ldots, x_n)$ y $q(x_1, \ldots, x_n)$ como palabras sobre un cierto alfabeto, entonces $m = |p(x_1, \ldots, x_n)| + |q(x_1, \ldots, x_n)|$

Una solución para el ejercicio

EquivPolAleatorizado$(p(x_1, \ldots, x_n),\ q(x_1, \ldots, x_n))$

$k := \max\{m_1, \ldots, m_\ell, s_1, \ldots, s_r\}$

$A := \{1, ..., 100 \cdot k\}$

for $i := 1$ **to** 10 **do**

sea $a_1, \ldots, a_n$ una secuencia de números elegidos de manera uniforme e independiente desde A

if $p(a_1, \ldots, a_n) \neq q(a_1, \ldots, a_n)$ **then return** no

else return sí