Algoritmos Aleatorizados

Parte I

Segundo semestre 2022

IIC2283

Prof. Nicolás Van Sint Jan

Outline

Outline

Introducción

Equivalencia de polinomios

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**.

¿ Por qué nos puede interesar un algoritmo que posea un **error** asociado al entregar su respuesta ?

¿Cuáles son las ventajas de los algoritmos aleatorizados?

1. Existen problemas para los cuales los algoritmos aleatorizados **son más eficientes** que los algoritmos deterministas (i.e. sin una componente aleatoria).
   - Por ejemplo, el problema de verificar si un número es primo.

2. 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 ...

Outline

Algoritmos de Monte Carlo: equivalencia de polinomios

Consideramos polinomios en $\mathbb{Q}$.

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

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

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

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 $\ell \leq \max\{\ell_1, \ldots, \ell_k\}$.

Si $c_\ell \neq 0$, entonces $p(x)$ no es el polinomio nulo y su grado es ℓ

Algoritmos de Monte Carlo: 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 $p(a) = q(a)$.

Suponga que la operación básica a contar es la **suma** y **multiplicación** de números racionales.

¿ Cómo podemos resolver este problema ?

Un algoritmo para la equivalencia de polinomios

EquivPol($p(x)$, $q(x)$)
  transforme $p(x)$ es su forma canónica $\sum_{i=0}^{k} c_i x^i$
  transforme $q(x)$ es su forma canónica $\sum_{i=0}^{\ell} d_i x^i$
  if $k \neq \ell$ **then return** no
  else
    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 $\mathcal{O}(n^2)$, donde $n = |p(x)| + |q(x)|$

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

Un algoritmo aleatorizado para la equivalencia de polinomios

Suponga que:

$$\begin{aligned} p(x) &= \sum_{i=1}^{k} \prod_{j=1}^{r_i} (a_{i,j} x + b_{i,j}) \\ q(x) &= \sum_{i=1}^{\ell} \prod_{j=1}^{s_i} (c_{i,j} x + d_{i,j}) \end{aligned}$$

Utilizamos el siguiente algoritmo aleatorizado:

EquivPolAleatorizado$(p(x),\ q(x))$

$K := 1 + \max\{r_1, \ldots, r_k, s_1, \ldots, s_\ell\}$

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 $\mathcal{O}(n)$ operaciones, 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 dados como entrada a **EquivPolAleatorizado**.

- 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

Sabemos que $r(x)$ no es el polinomio nulo y que es de grado a lo más K.

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

Concluimos que:

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

¿ Es **aceptable** esta probabilidad ?

Un mejor algoritmo aleatorizado

Ejercicio

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

$$\Pr(\text{obtener una respuesta incorrecta}) \leq \frac{1}{100^{10}}$$

¿ **Confiaría** en este algoritmo lineal ? ¿ Para qué probabilidad estaría dispuesto a confiar ?

Una solución para el ejercicio

Suponga que $p(x)$ y $q(x)$ son de la forma definida en la versión anterior de **EquivPolAleatorizado**.

EquivPolAleatorizado($p(x)$, $q(x)$)

$K := 1 + \max\{r_1, \ldots, r_k, s_1, \ldots, s_\ell\}$

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

$total := 0$

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

escoja al azar y con distribución uniforme un elemento a en A

if $p(a) = q(a)$ **then** $total = total + 1$

if $total = 10$ **then return** sí

return no