

RESUMEN: ALGORITMOS PROBABILISTAS, TIEMPO ESPERADO Y ALGORITMOS NUMÉRICOS

1. Algoritmos Probabilistas

Los algoritmos probabilistas son aquellos que incorporan alguna forma de aleatoriedad en su lógica para tomar decisiones. En lugar de producir siempre la misma salida para una misma entrada, pueden dar diferentes resultados dependiendo de valores aleatorios generados durante la ejecución. Se dividen en dos tipos principales:

- **Algoritmos de Monte Carlo:** siempre se ejecutan en tiempo determinado, pero pueden cometer errores con una pequeña probabilidad.
- **Algoritmos de Las Vegas:** siempre producen la salida correcta, pero el tiempo de ejecución puede variar.
- **Ventajas:** suelen ser más rápidos y simples que sus contrapartes deterministas.
- **Desventajas:** pueden no ser adecuados para aplicaciones donde se requiere precisión absoluta.

Ejercicio:

- Supongamos que tienes un algoritmo de Monte Carlo que resuelve un problema de decisión con un 95% de probabilidad de éxito. ¿Cuál es la probabilidad de que falle 2 veces consecutivas?
Solución: La probabilidad de fallo es $1 - 0.95 = 0.05$. Entonces, la probabilidad de fallar dos veces es:
 $P(\text{fallo dos veces}) = 0.05 * 0.05 = 0.0025$ (0.25%)

2. Tiempo Esperado

El tiempo esperado es una medida estadística que indica cuánto tiempo, en promedio, tarda un algoritmo en completarse. Se utiliza comúnmente en el análisis de algoritmos probabilistas o en aquellos cuyo comportamiento depende de la entrada.

Se calcula como el valor esperado (esperanza matemática) sobre todas las posibles ejecuciones del algoritmo:

$$E[T] = \sum (\text{tiempo}_i * \text{probabilidad}_i)$$

Donde tiempo_i es el tiempo que toma una ejecución particular y probabilidad_i es la probabilidad de esa ejecución.

Ejercicio:

- Un algoritmo tiene 3 posibles tiempos de ejecución: 1s (50% de probabilidad), 2s (30%) y 4s (20%).
Calcula el tiempo esperado.
 $E[T] = (1 * 0.5) + (2 * 0.3) + (4 * 0.2) = 0.5 + 0.6 + 0.8 = 1.9$ segundos.

3. Algoritmos Numéricos

Los algoritmos numéricos son procedimientos diseñados para resolver problemas matemáticos mediante operaciones aritméticas. Son esenciales en áreas como la ingeniería, física, estadística y ciencias de la computación.

Ejemplos de algoritmos numéricos incluyen:
-Método de Newton-Raphson para encontrar raíces.
-Eliminación de Gauss para resolver sistemas lineales.
-Métodos de integración numérica como Simpson o trapecios.

Consideraciones clave: estabilidad numérica, precisión, complejidad computacional y errores de redondeo.

Ejercicio:

- Utiliza el método de Newton-Raphson para encontrar una raíz de la función $f(x)$
 $f(x) = x^2 - 2$, con $x_0 = 1$:
 $f'(x) = 2x$
 $x_1 = x_0 - f(x_0)/f'(x_0) = 1 - (1^2 - 2)/(2 \cdot 1) = 1 + 0.5 = 1.5$
 $x_2 = 1.5 - (1.5^2 - 2)/(2 \cdot 1.5) \approx 1.4167$
Aproximación $\sqrt{2} \approx 1.4167$ en solo dos iteraciones.