Tema 6 Métodos Monte Carlo (I) Introducción y Generadores

1. Introducción
2. Ejemplos. La aguja del conde de Buffon.
3. Ventajas e inconvenientes.
4. Generadores. Introducción.
5. Números aleatorios verdaderos.
6. Números pseudoaleatorios
7. Algunos generadores conocidos.

1. Introducción. Definición

¿Qué es el método de Monte Carlo?

¿Método numérico que permite resolver problemas matemáticos mediante la simulación de variables aleatorias?

¿Experimentos matemáticos con números aleatorios?

Definición más precisa (Halton 1970):

Método de Monte Carlo es un método que intenta encontrar la solución de un problema de estimación de un parámetro de una población hipotética mediante la utilización de números aleatorios para construir una muestra de la población de la cual deducimos el parámetro.

Tipo de problemas

Probabilísticos:

Simulación de procesos físicos. Ejemplos:

- Reactores nucleares. Evolución neutrones.
- Poblaciones de insectos. Crecimiento.

Deterministas:

Problemas teóricos sin solución analítica.

- Ecuaciones diferenciales.
- Resolución de integrales.

A veces la separación entre un tipo y otro de problemas no es tan evidente

Método de Monte Carlo $\rightarrow$ Cualquier método que utilice números aleatorios

1. Introducción. Antecedentes

Antecedentes:

• En 1777 el conde de Buffon plantea su problema de la aguja sobre un fondo de líneas paralelas.

• Posteriormente, Laplace propone utilizar dicho método para estimar el número π.

• Lord Kelvin utilizó el muestreo aleatorio para resolver integrales de Teoría cinética de los gases.

• En 1908 Student utiliza técnicas similares para descubrir la distribución que lleva su nombre.

• En 1930 Fermi utiliza experimentos numéricos para estudiar el comportamiento de los neutrones al interaccionar con la materia.

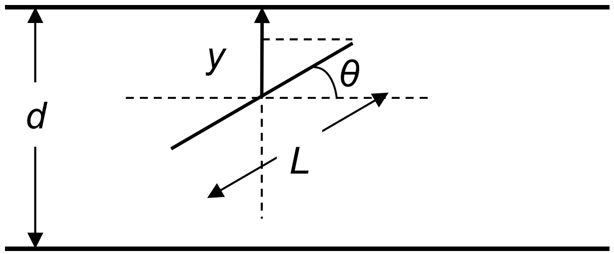
Nacimiento

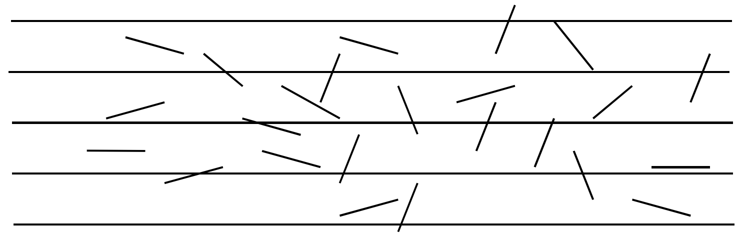
• En 1949 Fermi, Ulam, Neumann y Metropolis sientan las bases de los Métodos de Monte Carlo.

• El nacimiento de las calculadoras digitales permite su aplicación a diferentes campos.

2. Ejemplos. El agua del conde de Buffon

Si arrojamos una aguja de longitud *L* sobre una superficie plana sobre la que hay dibujadas una serie de líneas paralelas separadas una distancia *d* > *L* ¿Cuál es la probabilidad de que la aguja corte a las líneas?





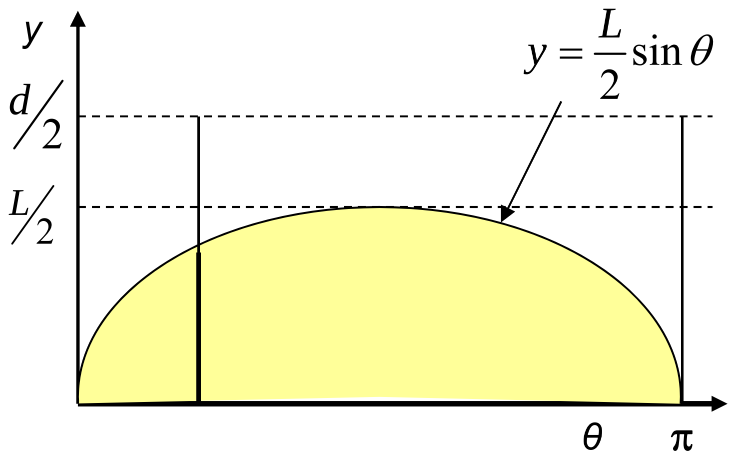
$y \rightarrow$ distancia del centro de la aguja a la línea superior.
$\theta \rightarrow$ ángulo respecto a la horizontal

¿Cuándo toca la aguja a la línea? $y < \frac{L}{2}\sin\theta$

Para un ángulo θ dado:

$$P = \frac{\frac{L}{2}\sin\theta}{\frac{d}{2}}$$

Para todos los ángulos será el cociente entre las áreas:



$$P = \frac{\int_0^{\pi} \frac{L}{2}\sin\theta d\theta}{\pi\frac{d}{2}} = \frac{2L}{\pi d} = \frac{\#(exitos)}{\#(exitos + fracasos)}$$

$$\pi = \frac{2L\#(éxitos + fracasos)}{d\#(éxitos)}$$

2. Ejemplos. El agua del conde de Buffon

¿Cuál es el error tras un número *n* de intentos?

Supongamos que de *n* intentos *r* son éxitos → distribución binomial

$$\begin{cases} \mu = np \\ \sigma^2 = np(1-p) \end{cases}$$

Si tomamos como variable $p = \frac{r}{n}$

$$\begin{cases} \mu = E\left[\frac{r}{n}\right] = \frac{1}{n} np = p \\ \sigma^2 = V\left[\frac{r}{n}\right] = \frac{1}{n^2} np(1-p) = \frac{p(1-p)}{n} \end{cases}$$

Como valor de *p* tomamos una estimación:

$$\hat{p} = \frac{r}{n} \quad ; \quad \sigma(\hat{p}) = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

$$\hat{p} \simeq \frac{2}{\pi} = 0.6366$$

El valor de π vendrá dado por:

Si $L = d$

$$\pi = \frac{2L\#(\acute{e}xitos + fracasos)}{d\#(\acute{e}xitos)} = \frac{2Ln}{dr} = 2\frac{n}{r} = \frac{2}{\hat{p}}$$

El error de π vendrá dado por:

$$\sigma(\pi) = \frac{2}{\hat{p}^2}\sigma(\hat{p}) = \frac{2}{\hat{p}^2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} = \frac{2.37}{\sqrt{n}}$$

El error de una magnitud calculada por Monte Carlo disminuye, en general, con la raíz cuadrada del número de sucesos

$$\sigma = \frac{\sigma_0}{\sqrt{n}}$$

Reducir el error en un factor ***k*** significa aumentar en ***k*²** el número de intentos!!!

$$\frac{\sigma}{k} = \frac{\sigma_0}{\sqrt{k^2 n}} = \frac{\sigma_0}{\sqrt{N}}$$

$$N = nk^2$$

$\sigma(\pi)$	*n*
0.2374	100
0.0237	10000
0.0024	1000000

2. Ejemplo. Integración Monte Carlo

Ley grandes números → Los valores medios tienden a sus valores esperados

Valor medio ⇒ $\bar{g} = \frac{1}{n}\sum g(x_i)$ / x_i variable aleatoria uniforme $f(x) = \frac{1}{(b-a)}$

Valor esperado ⇒ $E[\bar{g}(x)] = E[g(x)] = \int_a^b g(x) f(x)\, dx = \frac{1}{(b-a)}\int_a^b g(x)\, dx$

$$\bar{g} = \frac{1}{n}\sum g(x_i) \Rightarrow \frac{1}{(b-a)}\int_a^b g(x)\, dx$$

Varianza

$$V[\bar{g}] = V\left[\frac{1}{n}\sum g(x_i)\right] = \frac{1}{n^2}\sum V[g(x_i)] = \frac{1}{n^2}\sum V[g] = \frac{nV[g]}{n^2} = \frac{n\sigma^2(g)}{n^2} = \frac{\sigma^2(g)}{n}$$

Error en la integral ⇒ $\boxed{\sigma(\bar{g}) = \frac{\sigma(g)}{\sqrt{n}}}$ **Monte Carlo crudo**

3. Ventajas e inconvenientes

• Única forma de abordar muchos problemas complejos. Simulaciones.

• Simulado un problema podemos contestar a más de una pregunta (medias, fluctuaciones,…)

• Resolución de integrales con condiciones de contorno complicadas.

• En integración la convergencia es lenta pero en muchas dimensiones puede ser competitivo.

Incertidumbre en la integral calculada con n puntos		
Método	**1 dimensión**	**d dimensiones**
Monte Carlo	$n^{-1/2}$	$n^{-1/2}$
Regla trapezoidal	n^{-2}	$n^{-2/d}$
Regla de Simpson	n^{-4}	$n^{-4/d}$
Gauss	$n^{-(2m-1)}$	$n^{-(2m-1)/d}$

4. Generadores. Introducción

- No podemos decir si un número dado es aleatorio o no. Carece de sentido.
- Hablamos de secuencias de números aleatorios independientes con una distribución determinada.
- Cada número de la secuencia se obtuvo por azar sin tener nada que ver con cualquier otro número de la secuencia.
- Solo consideraremos generadores de números aleatorios uniformes.

Tipos de generadores.

Números aleatorios verdaderos.

- Producidos por un proceso físico aleatorio.
- Totalmente impredecibles.

Números pseudoaleatorios.

- Generados por un ordenador mediante un algoritmo numérico.
- No son realmente aleatorios pero son indistinguibles de los números aleatorios verdaderos.

Números quasialeatorios.

- No cumplen todos los requisitos de los números aleatorios.
- Distribuidos lo más uniformemente posible

4. Generadores. Propiedades

Propiedades de los generadores de números aleatorios.

Aleatoriedad .- Deben ser verdaderamente aleatorios en el sentido de que han de pasar una serie de pruebas o tests que garanticen su aleatoriedad.

Periodo .- Los números pseudoaleatorios y quasialeatorios presentan un periodo repitiéndose la secuencia al cabo de un cierto número de generaciones. El periodo ha de ser mayor que la secuencia de números que necesitemos.

Eficiencia.- En relación a la rapidez con que son generados los números aleatorios. Bajo coste de CPU y memoria.

Portabilidad.- La misma secuencia de números ha de ser generada por diferentes ordenadores. Propiedad asociada a la idea de repetibilidad.

5. Números aleatorios verdaderos.

Son realmente impredecibles e irrepetibles.

Son producidos por procesos físicos: desintegración radiactiva, ruido térmico en dispositivos electrónicos, tiempos de llegada de rayos cósmicos.

Inconvenientes

- Dificultades técnicas.
- Garantizar que no se produzca ningún sesgo.
- Hay que almacenarlos físicamente.

Ejemplo

(Frigerio 1978)

Número de partículas α registradas en un contador cada 10 ms:

Si resulta un número par → Tomamos un 0

Si resulta un número impar → Tomamos un 1

Posible sesgo → La probabilidad de obtener par o impar no son iguales:

$$P(\text{par}) = \sum_{k=0}^{\infty} \frac{\mu^{2k}}{(2k)!} e^{-\mu} = e^{-\mu} \cosh \mu = \frac{e^{-\mu}\left(e^{\mu} + e^{-\mu}\right)}{2} = \frac{1 + e^{-2\mu}}{2} \neq \frac{1}{2}$$

5. Números aleatorios verdaderos.

¿Cómo evitar este sesgo?

Consideramos pares de bits consecutivos de manera que:

- Si los dos bits son iguales se rechazan ambos.
- Si son diferentes solo se rechaza el primero

$$\cancel{0}0\,1\,0\,\cancel{1}1\,1\,0\,\cancel{0}0\,\cancel{1}1\,0\,1\,1\,0\ldots \;\rightarrow\; 0010$$

Llamamos:

- Probabilidades de obtener 0 y 1 en la primera secuencia: $P(0), P(1)$
- Probabilidades de obtener 0 y 1 en la segunda secuencia: $P^{'}(0), P^{'}(1)$

$$\left.\begin{array}{l} P^{'}(0) = P(1)\cdot P(0) \\ P^{'}(1) = P(0)\cdot P(1) \end{array}\right\} \Rightarrow P^{'}(0) = P^{'}(1)$$

Sin embargo la eficiencia es como máximo del 25 % !!!

Problema fundamental → Determinar la distribución exacta del mecanismo de generación (sesgo del dispositivo).

En general se trata de mecanismos complicados. Actualmente apenas se utilizan.

6. Números pseudoaleatorios

- Generados por ordenador mediante un algoritmo numérico.
- Son predecibles, no son realmente aleatorios.
- Aleatorios en el sentido de que son indistinguibles de los números aleatorios verdaderos.
- Misma probabilidad de pasar un test estadístico que los números aleatorios verdaderos.

Primeros intentos → Método de los medios cuadrados (Newman, 1946)

Partir de un número, elevarlo al cuadrado y quedarnos con las cifras centrales

Ejemplo →

$$(5232)^2 \to 27373824 \to 3738$$
$$(3738)^2 \to 13972644 \to 9726$$
$$(9726)^2 \to 94595076 \to 5950$$
$$\ldots$$

Inconvenientes:

- Ciclo pequeño.
- Números bajos.
- Si aparece el cero se acaba el proceso.

6. Números pseudoaleatorios

En general:

- Un generador consiste en una expresión matemática donde el resultado de cada generación depende de los números anteriores.
- El proceso se inicializa con un primer número denominado semilla.

$$x_n = f\left(x_{n-1}, \cdots, x_0\right) \qquad x_0 \rightarrow \text{ semilla (seed)}$$

- Se utilizan números enteros entre 0 y *m.*
- En general $m = 2^t - 1$ siendo t el número de bits.
- Para que el número aleatorio esté comprendido entre 0 y 1 se toma: $u_n = \dfrac{x_n}{m}$
- Como mucho el periodo será como máximo *m*.
- Ejemplo: con *t* = *32* podemos representar enteros entre 0 y $2^{32} - 1 \simeq 10^{10}$

6. Números pseudoaleatorios

Métodos congruentes

(Lehmer, 1949) Cálculo de residuos de una transformación lineal.

MLC → ***Multiplicative linear congruential***

$$X_{n+1} = (aX_n + c) \bmod m \quad ; \quad n \geq 0$$

$m \rightarrow$ módulo

$a \rightarrow$ multiplicador

$c \rightarrow$ incremento

$X_0 \rightarrow$ Valor inicial

Ejemplo

$m = 10, X_0 = a = c = 7$

$X_{n+1} = (7X_n + 7) \bmod 10$

n	X_n	$(7X_n + 7)$	$(7X_n + 7) \bmod 10$
0	7	56	6
1	6	49	9
2	9	70	0
3	0	7	7
4	7	56	6

Teoría de números

La secuencia original tiene periodo máximo si se cumplen las condiciones:

- ***Mixed congruential*** (*c ≠ 0*)
  - *c* y *m* son primos entre si.
  - *(a -1)/p* es un entero para cada *p* primo que divida a *m.*
  - Si *m* es múltiplo de 4 entonces *a -1* también lo es.
- ***Multiplicative*** (*c = 0*)
  - *m* ha de ser un número primo.
  - *a* ha de ser tal que $a^{(m-1)/p} \bmod m \neq 1 \; \forall$ los divisores primos de *(m -1).*

6. Números pseudoaleatorios

Métodos congruentes

Uno de los requisitos de este tipo de generadores es que no tenga correlaciones, sin embargo si se representan los pares de números consecutivos:

$$\left(\xi_i, \xi_{i+1}\right) i = 0, 1, \cdots$$

se obtienen patrones regulares difícilmente compatibles con una secuencia aleatoria verdadera

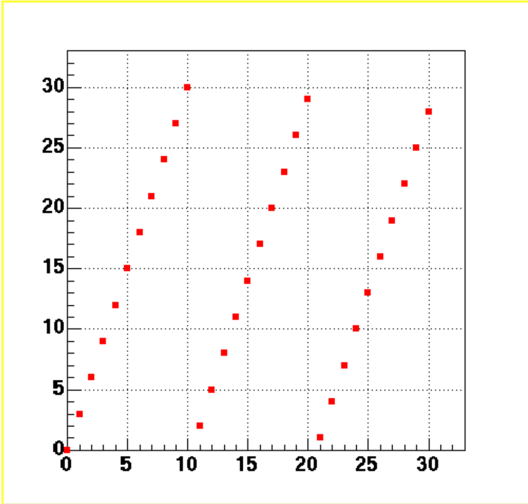
Ejemplo

$$m = 31, c = 0, a = 3, X_0 = 1$$

$$X_{n+1} = \left(3X_n + 0\right) \operatorname{mod} 31$$

$$1, 3, 9, 27, 19, 26, 16, 17, 20, 29, \cdots$$

¡¡ Fuerte correlación !!



Efecto Marsaglia

- Cuando se representan d-uplas de números generados congruentemente en un espacio d-dimensional, éstos se sitúan en determinados hiperplanos.
- Se buscan aquellos generadores que presentan mayor granularidad o número de hiperplanos

6. Números pseudoaleatorios

Otros métodos

Generadores de Fibonacci

- Serie de Fibonacci; cada número es la suma de los dos anteriores:1,1,2,3,5,8,…
- Generalización.

$$x_i = f\left(x_{i-1}, x_{i-2}\right)$$

- No son muy buenos. Se utilizan más los generadores de Fibonacci decalados:

$$x_i = \left(x_{i-p} \odot x_{i-q}\right) \bmod m$$

donde $\odot$ es una operación lógica binaria (OR exclusivo, suma, adición,…) y p y q son desfases

- Si $p > q$ los primeros p números se generan mediante un método congruente.

Generadores *shift-register* o de Tauseworth

- Están basados en la misma fórmula que los generadores de Fibonacci decalados pero con *m=2.*
- Se generan bits individuales que se agrupan en palabras.

7. Algunos generadores conocidos

RANMAR

Combinación de dos generadores:

1) Generador de Fibonacci decalado $\rightarrow$ $r_i = \left(r_{i-97} - r_{i-33}\right) \bmod 2^{24}$

2) $$t_i = \begin{cases} t_{i-1} - 7654321 & \text{si} \quad t_{i-1} - 7654321 \geq 0 \\ t_{i-1} - 7654321 + 2^{24} - 3 & \text{otros} \end{cases}$$

3) Finalmente $s_i = \left(r_i - t_i\right) \bmod 2^{24}$

G. Marsaglia, A. Zaman and W.W. Tsang *Stat. Prob. Lett.* 9 (1990) 35

ACARRY/RCARRY/RANLUX

Generador de Fibonacci decalado con un bit de añadido

$$\Delta_n = s_{n-i} - s_{n-j} - c_{n-1} \qquad j > i \geq 1$$

$$s_n = \Delta_n \qquad \rightarrow c_n = 0 \qquad \text{si} \quad \Delta_n \geq 0$$

$$s_n = \Delta_n + b \qquad \rightarrow c_n = 1 \qquad \text{otro}$$

Con $j = 24,\ i = 10$ y $b = 2^{24} \rightarrow$ ***RCARRY***

RANLUX $\rightarrow$ ***RCARRY modificado;*** *aceptamos j números y se rechazan los p-j siguientes*
p óptimo es 223