



Universidad de los Andes — Vigilada Mineducación. Reconocimiento como Universidad: Decreto 1297 del 30 de mayo de 1964.

Reconocimiento personería jurídica: Resolución 28 del 23 de febrero de 1949 Minjusticia.



# Métodos Computacionales

PhD. Alejandro Segura

2021



# Tabla de Contenido

<b>Lista de figuras</b>	<b>IV</b>
<b>1. Derivación e integración</b>	<b>1</b>
1.1. Error de redondeo . . . . .	1
1.2. Error de truncamiento . . . . .	1
1.3. Derivación . . . . .	1
1.3.1. Derivada Progresiva . . . . .	2
1.3.2. Derivada Regresiva . . . . .	2
1.3.3. Derivada Central . . . . .	2
1.3.4. Error Local . . . . .	3
1.3.5. Error global . . . . .	3
1.3.6. Segunda Derivada . . . . .	3
1.3.7. Ejercicios . . . . .	3
1.4. Método de Newton-Raphson . . . . .	4
1.4.1. Criterio de parada . . . . .	4
1.4.2. Ejercicios . . . . .	4
1.5. Interpolación de Lagrange . . . . .	5
1.5.1. Error . . . . .	5
1.5.2. Ejercicios . . . . .	6
1.6. Integración . . . . .	6
1.6.1. Método de trapecio simple . . . . .	6
1.6.2. Método de trapecio compuesto . . . . .	7
1.6.3. Método de Simpson simple 1/3 . . . . .	8
1.6.4. Método de Simpson compuesto . . . . .	9
1.6.5. Cuadratura Gaussiana . . . . .	11
1.6.6. Cuadratura de Gaus-Legendre . . . . .	12
1.6.7. Ejercicios . . . . .	15
<b>2. Método de MonteCarlo</b>	<b>17</b>
2.1. Generación pseudo-aleatoria de números . . . . .	17
2.2. Integración MonteCarlo . . . . .	18
2.2.1. Ley de grandes números . . . . .	18
2.2.2. Método de la transformada inversa . . . . .	19
2.2.3. Método de aceptación y rechazo . . . . .	20
2.2.4. Incertidumbre del método de MonteCarlo . . . . .	20
2.2.5. Metropolis-Hasting algorithm . . . . .	21

2.2.6. Estimación por máxima verosimilitud (Likelihood) . . . . .	22
2.2.7. Varianza de los estimadores . . . . .	23
2.3. Ejercicios . . . . .	24
<b>References</b>	<b>27</b>

# Índice de figuras

2.1. Correlaciones en los primeros k-vecinos para un mal generador (izquierda) y un buen generador (drand48) de números aleatorios. . . . .	18
---	----





# Capítulo 1

## Derivación e integración

### 1.1. Error de redondeo

Es la diferencia entre el valor exacto de un número y la aproximación calculada debida al redondeo. Por ejemplo,  $\pi = 3,1415926535\dots$  si se aproxima a 3,1416 el error es  $7,3464 \times 10^{-6}$ , entonces la pregunta natural es: ¿cuál es el número más pequeño que podemos aproximar usando el computador? en otras palabras ¿cuál es el valor de  $\epsilon$  para que se cumpla  $1 + \epsilon = 1$ .

### 1.2. Error de truncamiento

El error de truncamiento aparece cuando se usan aproximaciones en lugar de las expresiones exactas, en general, este tipo de error depende del tipo de aproximación que se realiza. Por ejemplo, cuando expandimos una cierta función alrededor de un punto y despreciamos términos de orden superior, se introducen error de truncamiento a nuestras estimaciones.

$$\begin{aligned} \sin(x) &\cong x + \mathcal{O}(x^3) \\ \sin(x) &\cong x - \frac{x^3}{3!} + \mathcal{O}(x^5) \end{aligned} \tag{1.1}$$

tiene diferente error de truncamiento para la estimación de la función  $\sin(x)$ .

### 1.3. Derivación

Para construir la derivada numérica se define la siguiente discretización para nodos equi-espaciados.

$$x_j = x_0 + jh, \tag{1.2}$$

donde  $h$  se denomina paso, que es en general una variación "pequeña" de la función.

### 1.3.1. Derivada Progresiva

Dada esta condición podemos hacer una expansión en series de Taylor de  $f(x)$  (el dominio son los puntos nodales).

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \dots \quad \text{para} \quad h \ll 1 \quad (1.3)$$

despejando la primera derivada tenemos:

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \underbrace{\frac{h}{2}f''(x)}_{\mathcal{O}(h)} \quad (1.4)$$

para algún punto de la partición:

$$f'(x_j) \cong \frac{f(x_{j+1}) - f(x_j)}{h} \quad (1.5)$$

La última expresión se denomina la derivada progresiva del punto  $x_j$ , la cuál tiene orden  $\mathcal{O}(h)$  en la estimación.

### 1.3.2. Derivada Regresiva

Para obtener la derivada regresiva se expande:

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) + \dots \quad \text{para} \quad h \ll 1 \quad (1.6)$$

despejando la primera derivada tenemos:

$$f'(x) = \frac{f(x) - f(x-h)}{h} + \underbrace{\frac{h}{2}f''(x)}_{\mathcal{O}(h)} \quad (1.7)$$

para algún punto de la partición:

$$f'(x_j) \cong \frac{f(x_j) - f(x_{j-1})}{h} \quad (1.8)$$

Para la derivada regresiva se tiene un orden de aproximación de orden  $\mathcal{O}(h)$ .

### 1.3.3. Derivada Central

Para estimar la derivada central se compara las expresiones de ambos desarrollos de Taylor.

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{3!}f'''(x) + \dots \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{3!}f'''(x) + \dots \end{aligned} \quad (1.9)$$

Restamos las dos expresiones tenemos:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \underbrace{\frac{h^2}{3}f'''(x)}_{\mathcal{O}(h^2)} \quad (1.10)$$

para algún punto de la partición:

$$f'(x_j) \cong \frac{f(x_{j+1}) - f(x_{j-1}))}{2h}. \quad (1.11)$$

Notar que la estimación central tiene un orden  $\mathcal{O}(h^2)$  en la estimación.

#### 1.3.4. Error Local

Una medida del error local es la distancia entre el valor estimado y el valor real.

$$\Delta_l(Df(x_j)) = f'(x_j) - \delta f_0(x_j) \quad (1.12)$$

#### 1.3.5. Error global

Se define el error global como la propagación de errores locales en todos los puntos de la discretización.

$$\Delta_g(Df(x_j)) = \sqrt{\frac{\sum_{j=1}^n (f'(x_j) - \delta f_0(x_j))^2}{\sum_{j=1}^n f'(x_j)^2}} \quad (1.13)$$

#### 1.3.6. Segunda Derivada

Para estimar la segunda derivada numérica, se suma los dos desarrollos de Taylor en la Ecuación (1.9).

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2) \quad (1.14)$$

para algún punto de la partición:

$$f''(x_j) \cong \frac{f(x_{j+1}) - 2f(x_j) + f(x_{j-1}))}{h^2} \quad (1.15)$$

Notar que la estimación tiene un orden  $\mathcal{O}(h^2)$  en la estimación.

#### 1.3.7. Ejercicios

1. Es posible construir una aproximación de orden  $\mathcal{O}(h^2)$  para la derivada progresiva y regresiva. Para tal propósito, escribir el polinomio de interpolación de grado 2, con  $x_1, x_2, x_3$ , siendo  $y_j = f(x_j)$  (ver sección de interpolación de Lagrange). Usar el polinomio interpolador para mostrar que la derivada progresiva de orden dos es:

$$f'(x_1) \approx p'(x_1) = \frac{1}{2h}(-3f(x_1) + 4f(x_2) - f(x_3)) \quad (1.16)$$

más generalmente se puede escribir como:

$$f'(x) \cong \frac{1}{2h}(-3f(x) + 4f(x+h) - f(x+2h)) \quad (1.17)$$

Para  $f(x) = \sqrt{\tan(x)}$ , estimar  $f'(x)$  en el intervalo  $[1, 2]$  con  $h = 0,05$ .

*Hint:* La derivada del polinomio interpolador es:

$$p'(x) = \frac{y_2 - y_1}{h} + \frac{1}{2h^2}(y_1 - 2y_2 + y_3)((x - x_1) + (x - x_2)) \quad (1.18)$$

2. Encuentre el operador  $D^4 f(x_j)$ . ¿Cuál es orden de la aproximación?

## 1.4. Método de Newton-Raphson

Es un método iterativo para encontrar las raíces reales polinomios usando conceptos de cálculo diferencial. Tomemos un punto cualquiera  $x_n$  y construimos la ecuación de la recta usando la derivada de  $f(x)$  en  $x_n$ .

$$Df(x_n) = \frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n} \quad (1.19)$$

Se pretende que el siguiente punto en la iteración sea una raíz de  $f(x_{n+1}) = 0$ , por tanto:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (1.20)$$

Este es conocida como la forma iterativa de Newton-Raphson. Otro camino para deducir esta formula consiste en expandir  $f(x)$  alrededor de  $x_n$ .

$$f(x) = f(x_n) + f'(x_n)(x - x_n) + \frac{(x - x_n)^2}{2!}f''(x_n) + \dots \quad (1.21)$$

Si se trunca la función hasta orden  $\mathcal{O}(x^2)$  y se evalúa en el siguiente punto  $x_{n+1}$ , el cuál se considera una raíz de  $f(x)$ ; se llega a la formula deseada.

### 1.4.1. Criterio de parada

Podemos usar el error relativo en cada iteración para detener el método.

$$\epsilon = \frac{|x_{k+1} - x_k|}{|x_{k+1}|} \quad (1.22)$$

el cual detiene el método cuando sea menor a una tolerancia, i.e,  $\epsilon < 10^{-6}$ .

### 1.4.2. Ejercicios

1. ¿De qué tipo es el error asociado a la estimación de raíces usando el método de Newton-Raphson?
2. ¿Cómo ajustar la precisión para estimar raíces con este método?

## 1.5. Interpolación de Lagrange

Descubierto por Edwarg Waring en 1779 y redescubierto por Leonhard Euler en 1783, fue publicado por Lagrange en 1795. Se plantea como sigue: dado un conjunto de  $n+1$  puntos diferentes  $\Omega = \{(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)\}$ , existe un polinomio interpolador de grado  $n$ :

$$p_n(x) = \sum_{i=0}^n f(x_i) \mathcal{L}_i(x), \quad (1.23)$$

donde  $\mathcal{L}_i(x)$  es la base de Lagrange (también conocidas como funciones cardinales).

$$\mathcal{L}_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} \quad (1.24)$$

Este polinomio cumple que  $p(x_k) = y_k$  para todo  $k$  en  $\{0, \dots, n\}$ .

### Ejemplo:

Encontrar las funciones cardinales ( $\mathcal{L}_i(x)$ ) y el polinomio interpolador para el siguiente conjunto:  $\Omega = \{(5, 10), (10, 15)\}$ .

$$\mathcal{L}_0(x) = \frac{x - 10}{5 - 10} = -\frac{1}{5}(x - 10) \quad (1.25)$$

$$\mathcal{L}_1(x) = \frac{x - 5}{10 - 5} = \frac{1}{5}(x - 5) \quad (1.26)$$

Por tanto, el polinomio interpolador es:

$$\begin{aligned} p_1(x) &= 10\mathcal{L}_0(x) + 15\mathcal{L}_1(x) \\ p_1(x) &= x + 5 \end{aligned} \quad (1.27)$$

### 1.5.1. Error

Sea  $f : I \rightarrow \mathbb{R}$ ,  $\{x_i\}_{i=0}^n \subseteq I$ ,  $x_i \neq x_j$  para  $i \neq j$  y suponemos que  $f$  es derivable  $n+1$  veces. El error asociado a la interpolación está dado por:

$$E = f(x) - p(x) = \frac{f^{n+1}(\xi_x)}{(n+1)!} (x - x_0)(x - x_1) \dots (x - x_n) \quad (1.28)$$

donde  $p(x)$  es el polinomio interpolador en  $\{x_i\}_{i=0}^n$  y  $\xi_x \in$  al intervalo que contiene los puntos.

### proof:

Si  $x$  es un punto  $x_k$  la identidad se satisface para cualquier  $\xi$ . De lo contrario, si  $x$  es fijo y diferente  $x_k$  se considera una función auxiliar:

$$F(t) = f(t) - p(t) - cL(t), \quad \text{donde} \quad c = \frac{f(x) - p(x)}{L(x)} \quad (1.29)$$

$L(x) = \prod_{i=0}^n (x - x_i)$ . Si evaluamos la función auxiliar en los puntos  $x_k$ ,  $F(x_k) = y_k - y_k - 0 = 0$  para todo  $k$ . Por tanto,  $F$  tiene  $n+1$  ceros. Adicionalmente  $F(x) = f(x) - p(x) - cL(x) = 0$ , dada

la definición de  $c$ . entonces la función  $F$  tiene  $n + 2$  ceros en el intervalo  $I$ . Ahora, por el teorema de Rolle,  $F'$  debe tener al menos  $n+1$  ceros en el intervalo que tiene a los puntos  $x_k$ ; entonces la  $(n+1)$ -ésima derivada debe tener al menos un cero. Sea  $\xi_x$  ese cero. Entonces derivamos  $(n + 1)$  veces y evaluamos en  $\xi_x$ :

$$F^{n+1}(\xi_x) = f^{n+1}(\xi_x) - c(n+1)! = 0 \quad (1.30)$$

debido a que la  $(n+1)$ -ésima derivada de  $p(x)$  es cero. Entonces:

$$c = \frac{f^{n+1}(\xi_x)}{(n+1)!} \rightarrow cL(x) = f(x) - p(x) = \frac{1}{(n+1)!} f^{n+1}(\xi_x) L(x) \quad \blacksquare \quad (1.31)$$

### 1.5.2. Ejercicios

1. Demuestre que el polinomio interpolador es único.
2. Compruebe que las funciones cardinales son base (i.e,  $L_i(x) = \delta_{ij}$  para cada  $j \in \{0, 1, \dots, n\}$ ).
3. ¿Con qué grado de exactitud podemos calcular  $\sqrt{114}$  mediante la interpolación de de Lagrange para la función  $f(x) = \sqrt{x}$ , si elegimos los puntos  $x_0 = 100$ ,  $x_1 = 121$ ,  $x_2 = 144$ .  
Rpta:  $|E| \simeq 1,8 \times 10^{-3}$ .

## 1.6. Integración

Para el calculo de integrales definidas existe una familia de métodos denominados *Métodos de Newton-Côtes*, los cuales se basan en cambiar el integrando  $f(x)$  a un polinomio interpolador que aproxima a  $f(x)$  en el intervalo de integración. El grado del polinomio interpolador queda definido por el número de puntos a considerar, por ejemplo, en los casos más simples de interpolación lineal (Regla de trapecio) e interpolación cuadrática (Regla de simpson), se tiene expresiones sencillas que son fáciles de implementar computacionalmente.

### 1.6.1. Método de trapecio simple

Para la integral definida:

$$I = \int_a^b f(x)dx, \quad (1.32)$$

el método de trapecios simple cambia el integrando por un polinomio interpolador de grado uno. Este polinomio interpolador esta definido en el conjunto  $\Omega = \{(a, f(a)), (b, f(b))\}$  que finalmente conduce a la siguiente aproximación:

$$f(x) \approx p_1(x) = \frac{x-b}{a-b}f(a) + \frac{x-a}{b-a}f(b), \quad \forall x \in [a, b]. \quad (1.33)$$

De modo que la integral tiene un valor aproximado:

$$I = \int_a^b f(x)dx \cong \int_a^b p_1(x)dx = \frac{b-a}{2}(f(a) + f(b)) \quad (1.34)$$

El error en la estimación está asociado al procedimiento de interpolación. Suponiendo que  $f(x)$  es continua y derivable de clase  $C^2$  en el intervalo  $[a, b]$ :

$$f(x) = p_1(x) + \epsilon(x), \quad (1.35)$$

donde

$$\epsilon(x) = \frac{f''(\xi)}{2}(x-a)(x-b), \quad a \leq \xi \leq b. \quad (1.36)$$

Entonces el error asociado a la integración por el método del trapecio está dado por ( $h = b - a$ ):

$$E = \int_a^b \epsilon(x) dx = -\frac{h^3}{12} f''(\xi) \quad (1.37)$$

De esta forma el error alcanza un valor máximo para algún valor de  $\xi$ , donde la segunda derivada de  $f(x)$  se maximice; de modo que podemos acotar el error máximo en esta estimación.

$$|E| \leq \frac{h^3}{12} \underbrace{\max_{a \leq \xi \leq b}}_{|f''(\xi)|}. \quad (1.38)$$

Notar que el error es de orden  $\mathcal{O}(h^3)$ .

### 1.6.2. Método de trapecio compuesto

Para la integral definida:

$$I = \int_a^b f(x) dx, \quad (1.39)$$

el método de trapecios compuesto genera una partición *equi-espaciada* tal que:  $x_{i+1} - x_i = h$ ,  $\forall_i = [1, \dots, n]$  sobre el conjunto  $\Omega = \{(x_0, f(x_0)), (x_1, f(x_1)), \dots, (x_n, f(x_n))\}$ . Las condiciones de borde corresponden con los límites de integración ( $x_0 = a$  y  $x_n = b$ ), definiendo el paso de integración  $h = \frac{b-a}{n}$ . La integral se puede escribir como:

$$\int_a^b f(x) dx = \int_{x_0}^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx, \quad (1.40)$$

aplicando el método de trapecios simple se tiene:

$$\int_a^b f(x) dx \approx \frac{h}{2}(f(x_0) + f(x_1)) + \frac{h}{2}(f(x_1) + f(x_2)) + \dots + \frac{h}{2}(f(x_{n-1}) + f(x_n)), \quad (1.41)$$

tenemos la expresión para la regla de trapecio compuesta:

$$\int_a^b f(x) dx \approx \frac{h}{2}(f(a) + f(b)) + h \sum_{i=1}^{n-1} f(x_i). \quad (1.42)$$

La estimación del error se calcula sumando los errores en cada sub-intervalo.

$$E = \sum_{i=1}^n E_i = -\frac{h^2}{12}(f''(\xi_1) + f''(\xi_2) + \dots + f''(\xi_n)) \quad (1.43)$$

El error puede ser acotado por el valor máximo promedio que tome la segunda derivada en el intervalo  $[a, b]$ .

$$|E| \leq \frac{h^2(b-a)}{12} \underbrace{\max_{a \leq \xi \leq b}} |f''(\xi)|. \quad (1.44)$$

Notar que el error es de orden  $\mathcal{O}(h^2)$ .

### 1.6.3. Método de Simpson simple 1/3

Para la integral definida:

$$I = \int_a^b f(x)dx, \quad (1.45)$$

el método de Simpson simple cambia el integrando por un polinomio interpolador de grado dos. Este polinomio interpolador esta definido en el conjunto  $\Omega = \{(a, f(a)), (x_m, f(x_m)), (b, f(b))\}$ , donde  $x_m = \frac{a+b}{2}$  es el punto medio del intervalo. La interpolación conduce a la siguiente aproximación:

$$f(x) \approx p_2(x) = \frac{(x-b)(x-x_m)}{(a-b)(a-x_m)}f(a) + \frac{(x-a)(x-b)}{(x_m-a)(x_m-b)}f(x_m) + \frac{(x-a)(x-x_m)}{(b-a)(b-x_m)}f(b), \quad \forall x \in [a, b]. \quad (1.46)$$

De modo que la integral tiene un valor aproximado:

$$\int_a^b f(x)dx \cong \int_a^b p_2(x)dx = \frac{h}{3}(f(a) + 4f(x_m) + f(b)). \quad (1.47)$$

Adicionalmente, es importante mencionar que *la discretización debe ser par*. El error de la aproximación tiene una característica interesante, dado que el error de la interpolación genera un resultado idénticamente nulo. Suponiendo que  $f(x)$  es continua y derivable de clase  $C^3$  en el intervalo  $[a, b]$ :

$$f(x) = p_2(x) + \epsilon(x), \quad (1.48)$$

donde

$$E = \int_a^b \epsilon(x)dx = \int_a^b \frac{f'''(\xi)}{4!}(x-a)(x-b)(x-(a+b)/2)dx = 0, \quad a \leq \xi \leq b. \quad (1.49)$$

Esto significa que la regla de Simpson es exacta a orden  $\mathcal{O}(h^3)$ , entonces necesitamos otro camino para calcular el error de la estimación a orden  $\mathcal{O}(h^4)$ . Para tal propósito, se va a expandir en series de Taylor alrededor del punto medio  $x_m$  a  $f(x)$  y al polinomio interpolador  $p_2(x)$ . Suponiendo que  $f(x)$  es continua y derivable de clase  $C^4$  en el intervalo  $[a, b]$ , a orden  $\mathcal{O}(h^4)$  se tiene:



$$f(x) = f(x_m) + f'(x_m)(x - x_m) + \frac{f''(x_m)}{2!}(x - x_m)^2 + \frac{f'''(x_m)}{3!}(x - x_m)^3 + \epsilon(x), \quad (1.50)$$

donde el error es de orden  $\mathcal{O}(h^4)$ .

$$\epsilon(x) = \frac{f^{(4)}(\xi)}{4!}(x - x_m)^4. \quad (1.51)$$

Para reemplazar en la regla simple de Simpson, se debe encontrar  $f(a)$  y  $f(b)$ , entonces:

$$\begin{aligned} f(a) &= f(x_m - h) = f(x_m) + f'(x_m)(-h) + \frac{f''(x_m)}{2!}(-h)^2 + \frac{f'''(x_m)}{3!}(-h)^3 + \epsilon(x_m - h) \\ f(b) &= f(x_m + h) = f(x_m) + f'(x_m)(+h) + \frac{f''(x_m)}{2!}(+h)^2 + \frac{f'''(x_m)}{3!}(+h)^3 + \epsilon(x_m + h) \end{aligned} \quad (1.52)$$

Por tanto, la regla de Simpson queda expresada como (Notar como el término de orden  $\mathcal{O}(h^3)$  no contribuye):

$$\begin{aligned} \frac{h}{3}(f(a) + 4f(x_m) + f(b)) &= \frac{h}{3}(6f(x_m) + f''(x_m)h^2 + \epsilon(x_m + h) + \epsilon(x_m - h)) \\ &= \frac{h}{3}(6f(x_m) + f''(x_m)h^2 + \frac{1}{12}f^{(4)}(\xi)h^4) \end{aligned} \quad (1.53)$$

Ahora integrando el desarrollo de Taylor dado por la Ecuación (1.50):

$$\int_a^b f(x)dx = 2hf(x_m) + \frac{f''(x_m)}{3}h^3 + \frac{f^{(4)}(\xi)}{60}h^5. \quad (1.54)$$

De modo que es posible calcular el error como la diferencia en las dos integrales.

$$\begin{aligned} E &= \int_a^b f(x)dx - \int_a^b p_2(x)dx \\ &= \frac{f^{(4)}(\xi)}{60}h^5 - \frac{f^{(4)}(\xi)}{36}h^5 = -\frac{1}{90}f^{(4)}(\xi)h^5 \end{aligned} \quad (1.55)$$

El error puede ser acotado por el valor máximo promedio que tome la cuarta derivada en el intervalo  $[a, b]$ ; notar que el error es de orden  $\mathcal{O}(h^5)$ .

$$|E| \leq \frac{1}{90}h^5 \underbrace{\max_{a \leq \xi \leq b}}_{\text{max}} |f^{(4)}(\xi)|. \quad (1.56)$$

#### 1.6.4. Método de Simpson compuesto

Para la integral definida:

$$I = \int_a^b f(x)dx, \quad (1.57)$$

el método de Simpson compuesto genera una partición *equi-espaciada* tal que:  $x_{i+1} - x_i = h$ ,  $\forall_i = [1, \dots, n]$  sobre el conjunto  $\Omega = \{(x_0, f(x_0)), (x_1, f(x_1)), \dots, (x_n, f(x_n))\}$ . Las condiciones de borde corresponden con los límites de integración ( $x_0 = a$  y  $x_n = b$ ). Definiendo el paso de integración como  $h = \frac{b-a}{n}$ , la integral se puede escribir:

$$\int_a^b f(x)dx = \int_a^{x_2} f(x)dx + \int_{x_2}^{x_4} f(x)dx + \dots + \int_{x_{n-2}}^b f(x)dx \quad (1.58)$$

Notar que el número de puntos de la partición deber ser par y que los puntos medios serían el sub-conjunto  $\{x_{2n+1}\}$ . Aplicando el método simple a cada integral tenemos:

$$\int_a^b f(x)dx \approx \frac{h}{3} \left( f(a) + 4f(x_1) + f(x_2) + f(x_2) + 4f(x_3) + f(x_4) + \dots + f(x_{n-2}) + 4f(x_{n-1}) + f(b) \right) \quad (1.59)$$

En general, podemos escribir la expresión final para el método de Simpson compuesto:

$$\int_a^b f(x)dx \approx \frac{h}{3} \left( f(a) + 4 \sum_{i=1, \text{impares}}^{n-1} f(x_i) + 2 \sum_{i=2, \text{pares}}^{n-2} f(x_i) + f(b) \right) \quad (1.60)$$

Para la estimación del error total, se debe considerar el error cometido en cada una de las integrales. De este modo:

$$|E| = \frac{1}{90} h^5 \left( f^{(4)}(\xi_1) + f^{(4)}(\xi_2) + \dots + f^{(4)}(\xi_{n/2}) \right) \quad (1.61)$$

Finalmente, acotando superiormente la estimación tenemos el error de la regla de Simpson compuesta a orden  $\mathcal{O}(h^4)$ .

$$|E| \leq \frac{b-a}{180} h^4 \underbrace{\max_{a \leq \xi \leq b}}_{\text{max}} |f^{(4)}(\xi)|. \quad (1.62)$$

Cabe resaltar que la regla de Simpson compuesta que aproxima el integrando por un polinomio de orden tres, reduce en dos ordenes el error cometido por método de trapecio compuesto.

### Ejemplo:

Para la integral:

$$\int_{-1}^1 \sqrt{1 + e^{-x^2}} dx, \quad (1.63)$$

Usando la regla de Simpson compuesta, ¿cuál debe ser el número de puntos para aproximar la integral con un error menor a 0.001?

Usando el operador discreto  $D^4 f(x_j)$  se estima el máximo en  $M = 3,183$ . Entonces:

$$n \geq \sqrt[4]{\frac{(b-a)^5}{E180}} M \approx 4,87 = 6 \quad \text{Intervalos} \quad (1.64)$$

debido que el número de intervalos debe ser par!

### 1.6.5. Cuadratura Gaussiana

La idea es aproximar la integral a una suma ponderada de la función  $f(x)$ , que use los valores más óptimos posibles para su determinación. En general, la formula de cuadratura está dada por:

$$G_M(f) = \sum_{i=1}^n w_i^{(n)} f(x_i^{(n)}) \cong \int_{-1}^1 f(x) dx. \quad (1.65)$$

Donde  $\{w_i^{(n)}\}$  son los pesos de ponderación y los  $\{x_i^{(n)}\}$  son los puntos de Gauss, ambos cantidades dependen del número de puntos que se elijan. A diferencia de los métodos anteriores, queremos encontrar una regla que nos permita maximizar el grado de precisión en nuestras estimaciones. En general, para calcular los coeficientes se exige que la regla sea exacta para el polinomio de mayor grado que se tenga. Tenemos entonces:

$$\int_{-1}^1 x^k dx = \sum_{i=1}^n w_i^{(n)} (x_i^{(n)})^k \quad k = 0, 1, \dots, N \quad (1.66)$$

donde  $N$  es tan grande como sea posible. Necesitamos  $2n$  condiciones para integrar polinomios de grado  $2n - 1$ , esto se denomina la regla de  $n$  puntos, para el grado polinomial  $2n - 1$ .

**Lema 1:** Si  $N = 2n$ , entonces no existe el conjunto  $\{w_i^{(n)}\}$  tal que se verifique la igualdad.

Supongamos que existe dicho polinomio, si definimos  $L(x) = \prod_{j=1}^n (x - x_j^{(n)})^2$  entonces  $L(x) \geq 0$  y por tanto la integral  $\int_{-1}^1 L(x) dx > 0$ . Sin embargo,  $\sum w_i^{(n)} L(x_i^{(n)}) = 0$ , entonces hay una contradicción.

**Lema 2:** Sea  $\{w_i^{(n)}\}$  los pesos y  $\{x_i^{(n)}\}$  los puntos de Gauss tal que se cumpla la Ecuación (1.66) para  $k = 0, 1, 2, \dots, N = 2n - 1$ , los pesos deben satisfacer:

$$w_i^{(n)} = \int_{-1}^1 L_i^{(n)}(x) dx, \quad (1.67)$$

con

$$L_i^{(n)}(x) = \prod_{k=1, k \neq i}^n \frac{x - x_k^{(n)}}{x_i^{(n)} - x_k^{(n)}} \quad (1.68)$$

Las bases cardinales son de grado  $\leq 2n - 1$  y  $L_i^{(n)}(x_j) = \delta_{ij}$ , como la regla de cuadratura debe ser exacta para los polinomios de grado  $2n - 1$ , entonces:

$$\int_{-1}^1 L_i^{(n)}(x) dx = \sum_{j=1}^n w_j^{(n)} L_i^{(n)}(x_j) = \sum_{j=1}^n w_j^{(n)} \delta_{ij} = w_i^{(n)} \quad (1.69)$$

Ahora, tenemos un polinomio de grado  $n$ ,  $P_n(x) = \prod_{i=1}^n (x - x_i^{(n)})$  y supongamos que tenemos un polinomio cualquier  $Q(x)$  de grado  $\leq n-1$ , de modo que la multiplicación tiene grado  $\leq 2n-1$ . Por tanto la identidad de la cuadratura se debe satisfacer.

$$\int_{-1}^1 P_n(x)Q(x)dx = \sum_{i=1}^n w_i^{(n)} P_n(x_i^{(n)})Q(x_i^{(n)}). \quad (1.70)$$

Por la definición de  $P_n(x)$  la integral es nula y los polinomios deben formar una base ortonormal. Adicionalmente, el conjunto  $\{x_i^{(n)}\}$  deben ser las raíces de los  $P_n(x)$ .

### Ejemplo:

Obtener la regla de cuadratura para de  $f(x)$  en el intervalo  $[-3, 3]$  usando la siguiente partición  $P = \{-1, 0, 1\}$ .

Calculando las funciones de Lagrange:

$$\begin{aligned} L_1(x) &= \frac{1}{2}x(x-1) \\ L_2(x) &= -(x+1)(x-1) \\ L_3(x) &= \frac{1}{2}x(x+1) \end{aligned} \quad (1.71)$$

Integrando dichas funciones en el intervalo  $[-3, 3]$  obtenemos el conjunto de pesos:

$$w_i^3 = \{9, -12, 9\} \quad (1.72)$$

Entonces la regla de cuadratura es:

$$\int_{-3}^3 f(x)dx \approx 3[3f(-1) - 4f(0) + 3f(1)]. \quad (1.73)$$

Note que es exacta para  $f(x) = x^2$ .

### 1.6.6. Cuadratura de Gaus-Legendre

De la necesidad del uso de polinomios ortonormales y el cálculo de sus raíces, surge la cuadratura de Gaus-Legendre. Este calculo usa los polinomios de Legendre como conjunto ortonormal.

$$p_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n, \quad (1.74)$$

Los primeros polinomios son:

$$\begin{aligned} p_0(x) &= 1 \\ p_1(x) &= x \\ p_2(x) &= \frac{1}{2}(3x^2 - 1) \\ p_3(x) &= \frac{1}{2}(5x^3 - 3x) \end{aligned} \quad (1.75)$$

cumplen con la siguiente relación de completez:

$$\int_{-1}^1 p_n(x)p_m(x)dx = \begin{cases} 0 & m \neq n \\ \frac{2}{2n+1} & m = n \end{cases} \quad (1.76)$$

y las siguientes relaciones de recurrencia.

$$(n+1)p_{n+1}(x) = (2n+1)xp_n(x) - np_{n-1}(x). \quad (1.77)$$

$$(x^2 - 1)\frac{dp_n}{dx} = nxp_n - np_{n-1}. \quad (1.78)$$

Adicionalmente se tiene el siguiente coeficiente director.

$$a_n = \frac{(2n)!}{2^n(n!)^2} \quad (1.79)$$

Para calcular los pesos de la cuadratura de Gauss-Legendre, se requiere escribir las funciones cardinales en términos de los polinomios de Legendre. Si definimos las funciones base de Lagrange como:

$$L(x) = \frac{p_n(x)}{x - x_k} \quad (1.80)$$

Tenemos una forma indeterminada cuando  $x = x_k$ . Para conocer dicho límite, si existe, usemos la regla de L'Hôpital.

$$\lim_{x \rightarrow x_k} L(x) = \left. \frac{\frac{dp_n(x)}{dx}}{\frac{d(x-x_k)}{dx}} \right|_{x=x_k} = p'_n(x_k) \quad (1.81)$$

como las funciones de Lagrange deben ser base y valor 1 justo en cada raíz, se tiene la siguiente expresión.

$$L(x) = \frac{1}{p'_n(x_k)} \frac{p_n(x)}{x - x_k} \quad (1.82)$$

Por tanto, los pesos pueden ser evaluados por:

$$w_k = \frac{1}{p'_n(x_k)} \int_{-1}^1 \frac{p_n(x)}{x - x_k} dx \quad (1.83)$$

### Ejemplo:

Calcular los pesos de ponderación para dos puntos ( $n = 2$ ). El polinomio de Legendre sería  $p_2(x) = \frac{1}{2}(3x^2 - 1)$ , cuyas raíces son:  $x_0 = \frac{1}{\sqrt{3}}$ ,  $x_1 = -\frac{1}{\sqrt{3}}$ . Los pesos están dados por:

$$w_0 = \frac{1}{3(\frac{1}{\sqrt{3}})} \int_{-1}^1 \frac{\frac{1}{2}(3x^2 - 1)}{x - \frac{1}{\sqrt{3}}} dx = 1. \quad (1.84)$$

$$w_1 = \frac{1}{3(-\frac{1}{\sqrt{3}})} \int_{-1}^1 \frac{\frac{1}{2}(3x^2 - 1)}{x + \frac{1}{\sqrt{3}}} dx = 1. \quad (1.85)$$

Por tanto, la regla de cuadratura para dos puntos es:

$$\int_{-1}^1 f(x)dx = f\left(\frac{1}{\sqrt{3}}\right) + f\left(-\frac{1}{\sqrt{3}}\right) \quad (1.86)$$

para integrales que tiene un límite de integración diferente al de los polinomios de Legendre, es posible hacer un cambio de variable de modo que -1 coincida con el límite inferior y 1 con el límite superior de la integral. Planteando la siguiente proporción:

$$\frac{x-a}{b-a} = \frac{t-(-1)}{1-(-1)}, \quad (1.87)$$

de manera que:

$$\begin{aligned} x &= \frac{1}{2}[t(b-a) + a + b] \\ dx &= \frac{1}{2}(b-a) \end{aligned} \quad (1.88)$$

Usando este cambio de variable, podemos escribir la regla de cuadratura para cualquier intervalo de integración.

$$\int_a^b f(x)dx = \frac{1}{2}(b-a) \sum_{k=0}^n w_k f\left(\frac{1}{2}[t_k(b-a) + a + b]\right) \quad (1.89)$$

La Ecuación (1.83) aunque es correcta, resulta poco útil para calcular los pesos a alto orden computacionalmente. Es posible encontrar una versión integrada de dicha expresión usando el siguiente teorema:

**Theorem 1.** (*Identidad de Christoffel-Darboux*) Sea  $\{p_n(x)\}_{n=0}^{\infty}$  el sistema ortonormal de polinomios respecto a un peso  $\omega(x)$ . Entonces, para todo  $x, y \in R$ ,  $x \neq y$  y  $n \geq 1$  se cumple.

$$\sum_{k=0}^{n-1} p_k(x)p_k(y) = \frac{a_{n-1}}{a_n} \left[ \frac{p_n(x)p_{n-1}(y) - p_{n-1}(x)p_n(y)}{x-y} \right] \quad (1.90)$$

siendo  $p_n(x) = a_n x^n + \dots$  y  $a_n > 0$

Usando esta identidad y haciendo  $y = x_k$  las raíces de los polinomios tenemos.

$$\sum_{k=0}^{n-1} p_k(x)p_k(x_k) = \frac{a_{n-1}}{a_n} \frac{p_n(x)p_{n-1}(x_k)}{x-x_k} \quad (1.91)$$

Integrando, tenemos:

$$1 = \frac{a_{n-1}}{a_n} p_{n-1}(x_k) \int_{-1}^1 \frac{p_n(x)}{x-x_k} dx \quad (1.92)$$

Por tanto, podemos escribir la Ecuación (1.83) como:

$$w_k = \frac{a_n}{a_{n-1}p_{n-1}(x_k)p'_n(x_k)} \quad (1.93)$$

Un cálculo sencillo muestra que  $\frac{a_n}{a_{n-1}} = 2/n$  y usando la Ecuación (1.78).

$$w_k = \frac{2}{(1 - x_k^2)[p'_n(x_k)]^2}, \quad k = 1, \dots, n \quad (1.94)$$

Esta última expresión permite calcular los pesos usando librerías como **Simpy**.

### 1.6.7. Ejercicios

1. Hacer pasos intermedios para regla de trapecio simple, Ecuación (1.34).
2. Encontrar el error para regla de trapecio simple, Ecuación (1.37).
3. Hacer los pasos intermedios para encontrar la regla de Simpson simple, Ecuación (1.47).
4. Verificar el resultado presentado en la Ecuación (1.49).
5. La regla de Simpson 3/8 consiste en aproximar el integrando por un polinomio interpolador de orden 3. Use el paquete **Simpy** de **Python** para encontrar las funciones cardinales de dicha aproximación. Use los resultados para demostrar que:

$$\int_a^b f(x) \approx \frac{3h}{8} \left[ f(a) + 3f\left(\frac{2a+b}{3}\right) + 3f\left(\frac{a+2b}{3}\right) + f(b) \right]. \quad (1.95)$$

Es importante mencionar que *la discretización debe ser múltiplo de tres*.

6. Muestre que error asociado a la regla de Simpson 3/8 simple está dado por:

$$E = \frac{f^{(4)}(\xi)}{4!} \int_a^b (x - x_0)(x - x_1)(x - x_2)(x - x_3)dx = -\frac{3}{80}h^5 f^{(4)}(\xi). \quad (1.96)$$

*Hint:* Considere la siguiente integral:

$$\mathcal{I} = \int_0^{3h} (x)(x-h)(x-2h)(x-3h)dx \quad (1.97)$$

7. Muestre que para la regla de Simpson 3/8 compuesta, el error puede ser acotado por:

$$|E| \leq \frac{(b-a)}{80} h^4 \underbrace{\max}_{a \leq \xi \leq b} |f^{(4)}(\xi)|. \quad (1.98)$$

8. Evaluar:

$$\int_1^2 \frac{1}{x^2} dx, \quad (1.99)$$

por medio de la cuadratura de Gauss-Legendre con dos y tres puntos. Rpta:  $I_2 = 0,497041$ ,  $I_3 = 0,499874$ .

9. Escribir el polinomio  $p(x) = 3 + 5x + x^2$  en la base de Legendre. Rpta:  $p(x) = \frac{10}{3}p_0(x) + 5p_1(x) + \frac{2}{3}p_2(x)$
10. En el problema de la radiación de cuerpo negro aparece la siguiente integral.

$$\int_0^\infty \frac{x^3}{e^x - 1} dx = \frac{\pi^4}{15}. \quad (1.100)$$

- a) Calcular la integral usando el método de la cuadratura Gauss-Laguerre para  $n = 3$  puntos.
- b) Para esta estimación, gráfique el error relativo ( $\epsilon_r(n) = I_{estimated}(n)/I_{exact}$ ) como una función del número de puntos, con  $n = [2, 3, \dots, 10]$ .

Este método es adecuado para estimar integrales del tipo:

$$\int_0^\infty e^{-x} f(x) \approx \sum_{i=1}^n w_i f(x_i) \quad (1.101)$$

*Hint:* Los pesos para la cuadratura de Gauss-Laguerre están dados por:

$$w_k = \frac{x_k}{(n+1)^2 [L_{n+1}(x_k)]^2} \quad (1.102)$$



## Capítulo 2

# Método de MonteCarlo

### 2.1. Generación pseudo-aleatoria de números

En general, no es posible generar auténticos números aleatorios por computador. Para obtener una secuencia aparentemente aleatoria, se utilizan métodos deterministas con altos periodos de repetición. Los métodos de congruencia lineal están definidos por:

$$x_{n+1} = (ax_n + c) \bmod m \quad n \geq 0, \quad (2.1)$$

donde  $0 \leq a < m$  se denomina multiplicador,  $0 \leq c < m$  es el incremento y  $m$  es el módulo del método. En la secuencia  $x_0$  se denomina la semilla del generador y debe ser ajustada *inteligentemente* para obtener secuencias diferentes en cada generación. En la expresión congruencial, el modulo representa el resto entre  $(ax_n + c)$  y  $m$ , Por ejemplo si  $x_0 = 1$ ,  $a = 16$ ,  $c = 0$  y  $m = 7$  tenemos la siguiente secuencia  $x_1 = 2$ ,  $x_2 = 4$ , .... Uno de los generadores más conocidos y usados tanto en **C++** como en **Python** está definido por:

$$\begin{aligned} a &= 5DEECE66D \text{ (base 16)} \\ c &= B \text{ (base 16)} \\ m &= 2^{48} \end{aligned} \quad (2.2)$$

Para obtener una secuencia de números normalizada entre cero y uno ( $0 \leq r \leq 1$ ), se debe normalizar al modulo ( $m$ ) la ecuación congruencial.

$$r = \frac{x_{n+1}}{m} = \frac{(ax_n + c) \bmod m}{m} \quad (2.3)$$

Para obtener una distribución de número en otro intervalo cualquiera ( $A \leq x \leq B$ ) se tiene que escalar la distribución  $\mathcal{U}(0, 1)$ .

$$x = A + (B - A)r. \quad (2.4)$$

Por otro lado, es importante garantizar la aleatoriedad de la secuencia, para tal propósito, se tienen algunas pruebas a la función de distribución de dicha secuencia [1], por ejemplo, podemos usar el momento de la distribución ( $P(x)$ ):

$$\frac{1}{N} \sum_{i=1}^N x_i^k \cong \int_0^1 x^k P(x) dx \cong \frac{1}{k+1} + \mathcal{O}\left(\frac{1}{\sqrt{N}}\right) \quad (2.5)$$

También se puede estudiar la correlación entre los  $k$ -vecinos cercanos, donde  $k$  es pequeño:

$$C(k) = \frac{1}{N} \sum_{i=1}^N x_i x_{i+k}, \quad k = 1, 2, \dots \quad (2.6)$$

Si la secuencia es aproximadamente uniforme podemos aproximar la suma como:

$$\frac{1}{N} \sum_{i=1}^N x_i x_{i+k} \cong \int_0^1 dx \int_0^1 dy xy P(x, y) = \frac{1}{4}. \quad (2.7)$$

De modo que, si la distribución de números pseudo-aleatorio sigue una función de distribución uniforme,  $C(k)$  debe tener como máximo  $1/4$ . La Figura [2.1] muestra las correlaciones de los primeros 30 vecinos para un generador de números aleatorios malo y uno bueno. Note que un buen generador de eventos debe mantener las correlaciones alrededor de  $1/4$ .

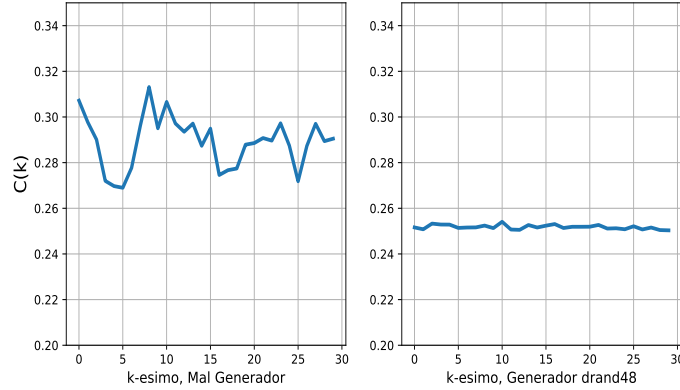


Figura 2.1: Correlaciones en los primeros  $k$ -vecinos para un mal generador (izquierda) y un buen generador (drand48) de números aleatorios.

## 2.2. Integración MonteCarlo

La integración Montecarlo es un método de integración de funciones generales que se sustenta en la ley de grandes números.

### 2.2.1. Ley de grandes números

Sea  $\{x_i\}$  un conjunto de puntos muestrales (observaciones o datos simulados por computador) y el promedio muestral de dicho conjunto.

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i. \quad (2.8)$$

Al aumentar el tamaño de la muestra, el promedio muestra converge al valor esperado con probabilidad 1.

$$\begin{array}{ccc} \text{Muestral} & \rightarrow & \text{Poblacional} \\ \bar{x} & \rightarrow & \mathbb{E}(x) \end{array}$$

De esta manera, el operador integración  $\int_0^1 f(x)dx$  puede verse como el valor esperado de  $\mathbb{E}(f(x))$  donde  $x \sim \mathcal{U}(0,1)$  tiene una distribución uniforme. De este modo, tomemos una muestra  $x_1, x_2, \dots \sim \mathcal{U}(0,1)$  y evaluamos en la función  $f(x_1), f(x_2), \dots$ , entonces:

$$\frac{1}{N} \sum_{i=1}^N f(x_i) \xrightarrow[N \rightarrow \infty]{} \mathbb{E}(f(x)) \quad (2.9)$$

Otro resultado interesante es que si  $x$  es una variable aleatoria con densidad de probabilidad  $f$  y  $g$  es una función  $g : \mathbb{R} \rightarrow \mathbb{R}$ , entonces:

$$\mathbb{E}(g(x)) = \int_{-\infty}^{\infty} g(x)f(x)dx = \sum_{i=1}^N g(x_i)f(x_i). \quad (2.10)$$

### 2.2.2. Método de la transformada inversa

Sea  $x$  una variable aleatoria con densidad de probabilidad  $f$ , y sea  $F(x) = \int_{-\infty}^x f(s)ds$  la función acumulada de  $f$ . Dado que, la función acumulada de probabilidad es monótona creciente resulta ser inyectiva. De manera que, existe un único valor en el rango tal que:

$$F^{-1}(u) = x. \quad (2.11)$$

Por tanto, podemos generar una distribución uniforme y tomar la pre-imagen en la distribución acumulada. El resultado es el siguiente:

Sea  $U \sim \mathcal{U}(0,1)$  y  $F$  una distribución acumulada, entonces  $F$  es inyectiva en algún intervalo pre-imagen de  $[0,1]$ . Si  $X$  se define como  $F^{-1}(U) = X$ , entonces  $X$  tiene distribución  $F$ .

**proof:**

Queremos ver si  $P(X \leq x) = F(x)$ , lo que significa que  $x$  tiene distribución  $F$ . Por definición de  $X$ :

$$P(X \leq x) = P(F^{-1}(U) \leq x) \quad (2.12)$$

Como  $F$  es monótona creciente en el intervalo  $[0,1]$  es invertible:

$$\begin{aligned}
P(X \leq x) &= P(F(F^{-1}(U)) \leq F(x)) \\
&= P(U \leq F(x)) \\
&= F(x)
\end{aligned} \tag{2.13}$$

El último paso se justifica dado que  $U$  tiene distribución uniforme  $U \sim \mathcal{U}(0, 1)$ . El algoritmo se resume como sigue:

1. Generar un número aleatorio que siga una distribución uniforme  $U \sim \mathcal{U}(0, 1)$ .
2. Tomar la pre-imagen de  $U$ ,  $X = F^{-1}(U)$ .

### 2.2.3. Método de aceptación y rechazo

En la gran mayoría de los casos es imposible encontrar la función inversa para usar el método de transformada inversa. En estos casos, se utiliza el método de aceptación y rechazo, el cual mediante la distribución uniforme de números puede calcular el área bajo la curva de distribuciones complicadas. Este método consiste se base en los siguientes pasos:

1. Generamos un  $x_i$  que siga una distribución uniforme entre contenido en el intervalo de integración  $[a, b]$ .
2. Para  $x_i$ , generamos un  $y_i$  que siga una distribución uniforme entre 0 y el máximo de  $f(x)$ .
3. Si  $y_i < f(x_i)$  incluimos el valor  $x_i$  en la lista, de otro modo, no incluimos el valor  $x_i$ .

### 2.2.4. Incertidumbre del método de MonteCarlo

La incertidumbre en el método de MonteCarlo se estima propagando el error cometido en cada uno de los puntos muestrales  $x_i$ . Recordemos que la media muestral está dada por:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \tag{2.14}$$

entonces:

$$\frac{\partial \bar{x}}{\partial x_i} = \frac{1}{N} \tag{2.15}$$

Teniendo en cuenta que cada medición es independiente la varianza asociada a la media muestra es:

$$Var(\bar{x}) = \sum_{i=1}^N \left( \frac{\partial \bar{x}}{\partial x_i} \right)^2 \delta_i^2 = \sum_{i=1}^N \frac{1}{N^2} \delta_i^2 \tag{2.16}$$

Por tanto, la incertidumbre está dada por:

$$\sqrt{Var(\bar{x})} \cong \frac{\delta}{\sqrt{N}} \tag{2.17}$$

$\delta$  es despreciable con el número de eventos simulados, en general se ajusta a 1 para tener una relación exacta con  $1/\sqrt{N}$ .

En esta ecuación es importante resaltar que la precisión del Método disminuye con  $\sqrt{N}$ , lo cuál debe tenerse en cuenta para cada calculo realizado con esta técnica. En el cálculo del número aureo usando la sucesión Fibonnaci aunque los puntos no provienen de un proceso aleatorio, la precisión del método tiene una dependencia similar.

### 2.2.5. Metropolis-Hasting algorithm

Suppose you wish to sample the random variable  $x \sim f(x)$ , but you can not use the direct simulation, the inverse CDF or accept-reject methods. But you can evaluate  $f(x)$  at least up to a proportionality constant; then you can use the Metropolis-Hastings algorithm.

Let  $f(x)$  be the (possible unnormalized) target density,  $x^j$  be a current value and  $q(x/x^j)$  be a proposal distribution, which might depend on the current value  $x^j$ . The algorithm is summarized as follows:

- a) Sample  $x^* \sim q(x/x^j)$ .
- b) Calculate the acceptance probability:

$$\rho(x^j, x^*) = \min \left[ 1, \frac{f(x^*)q(x^j/x^*)}{f(x^j)q(x^*/x^j)} \right], \quad (2.18)$$

if  $\rho(x^j, x^*) > r$ ,  $x^*$  is accepted. Where  $r \sim \mathcal{U}(0, 1)$ .

- c) Set  $x^{j+1} = x^*$  with probability  $\rho(x^j, x^*)$ , otherwise set  $x^{j+1} = x^j$ .

In general, this sequence is not independent, moreover, we concentrate on symmetry chains (random walks) where  $q(x/x^j) = q(x^j/x)$ . The acceptance probability is, therefore:

$$\rho(x^j, x^*) = \min \left[ 1, \frac{f(x^*)}{f(x^j)} \right] \quad (2.19)$$

**How this algorithm works?**

Let  $D_n(x)$  be the points density  $x_n$  around  $x$  and  $D_n(y)$  the points density  $y_n$  around  $y$ .  $x$  will be accepted or rejected. We can quantify the flux of points on the neighborhood  $x$  as:

$$\delta D_n(x) = \delta D_{n+1}(x) - \delta D_n(x) \quad (2.20)$$

$$\delta D_n(x) = \underbrace{\sum_y D_n(y) P(y \rightarrow x)}_{Gain} - \underbrace{\sum_y D_n(x) P(x \rightarrow y)}_{Lost} \quad (2.21)$$

$$\sum_y D_n(y) P(x \rightarrow y) \left[ \frac{P(y \rightarrow x)}{P(x \rightarrow y)} - \frac{D_n(x)}{D_n(y)} \right] \quad (2.22)$$

We have two properties:

1. The equilibrium is achieved asymptotically.
2. There is flux toward  $y$ , which tends  $D_n(x)$  to the equilibrium.

$$\frac{P(y \rightarrow x)}{P(x \rightarrow y)} < \frac{D_n(x)}{D_n(y)} \quad (2.23)$$

As a reminder  $\rho(x, y) > r$  is the acceptance condition. We can calculate  $P(x \rightarrow y)$  as  $p_{xy}A_{xy}$ , where  $p_{xy}$  is the probability of generation of  $y$  given  $x$ , and  $A_{xy}$  is the acceptance probability. Since the Metropolis strategy is symmetric ( $p_{xy} = p_{yx}$ ), we get:

$$\frac{P(y \rightarrow x)}{P(x \rightarrow y)} = \frac{A_{yx}}{A_{xy}} = \rho(x, y) = \frac{f(x)}{f(y)} \quad (2.24)$$

Therefore, the equilibrium distribution is given by:

$$D_{eq} = D(x) = f(x) \quad (2.25)$$

### 2.2.6. Estimación por máxima verosimilitud (Likelihood)

Sea  $\mathcal{A}(x_1, x_2, \dots, x_n)$  un vector aleatorio cuya distribución depende de un parámetro  $\theta$ . Definimos la función de verosimilitud de  $\mathcal{A}$  como:

$$\mathcal{L}(\theta) = f_{x_1, x_2, \dots, x_n}(x_1, x_2, \dots, x_n; \theta) \quad (2.26)$$

que es la función de densidad de probabilidad evaluada en el vector  $\mathcal{A}$  y es función del parámetro; note que no deben ser idénticamente distribuidas. En caso que sean variables idénticamente distribuidas tenemos:

$$\begin{aligned} \mathcal{L}(\theta) &= f(x_1; \theta)f(x_2; \theta)\dots f(x_n; \theta) \\ &= \prod_{i=1}^n f(x_i; \theta). \end{aligned} \quad (2.27)$$

Este método de estimación de parámetros consiste en maximizar la función  $\mathcal{L}(\theta)$ , el valor del parámetro que maximiza la función se denota por  $\hat{\theta}$  y se denota como estimador máximo verosímil.

#### Ejemplo:

Sea  $\mathcal{A}(x_1, x_2, \dots, x_n)$ , donde  $\mathcal{A} \sim e^\alpha$ .

$$\mathcal{L}(\alpha) = \prod_{i=1}^n \frac{1}{\alpha} e^{-x_i/\alpha} = \left(\frac{1}{\alpha}\right)^n e^{-\sum_{i=1}^n x_i/\alpha} \quad (2.28)$$

Dado que la función logaritmo natural es monótona, podemos maximizar  $Ln(\mathcal{L}(\alpha))$ .

$$Ln(\mathcal{L}(\alpha)) = nLn(1/\alpha) - \sum_{i=1}^n \frac{x_i}{\alpha} \quad (2.29)$$

Encontrando el máximo de esta función, tenemos:

$$\hat{\alpha} = \frac{1}{n} \sum_{i=1}^n x_i = \bar{X} \quad (2.30)$$

**Ejemplo:**

Sea  $\mathcal{A}(x_1, x_2, \dots, x_n)$ , donde  $\mathcal{A} \sim \text{Pois}(\lambda)$ .

$$\mathcal{L}(\lambda) = \prod_{i=1}^n \frac{e^{-\lambda} \lambda^{x_i}}{x_i!} = \frac{e^{-n\lambda} \lambda^{\sum_{i=1}^n x_i}}{\prod_{i=1}^n x_i!} \quad (2.31)$$

Tomando logaritmo natural tenemos:

$$\text{Ln}(\mathcal{L}(\lambda)) = -n\lambda + \sum_{i=1}^n x_i \text{Ln}(\lambda) - \text{Ln}\left(\prod_{i=1}^n x_i!\right) \quad (2.32)$$

El siguiente sería el parámetro máximo verosímil.

$$\hat{\lambda} = \frac{1}{n} \sum_{i=1}^n x_i = \bar{X} \quad (2.33)$$

### 2.2.7. Varianza de los estimadores

Vamos a expandir la función de verosimilitud alrededor del estimador, lo que significa que estamos en el punto máximo.

$$\text{Ln}(\mathcal{L}(\theta)) \cong \text{Ln}(\mathcal{L}(\hat{\theta})) + \left. \frac{\partial \text{Ln}(\mathcal{L}(\theta))}{\partial \theta} \right|_{\theta=\hat{\theta}} (\theta - \hat{\theta}) + \frac{1}{2} \left. \frac{\partial^2 \text{Ln}(\mathcal{L}(\theta))}{\partial \theta^2} \right|_{\theta=\hat{\theta}} (\theta - \hat{\theta})^2 \quad (2.34)$$

Dado que estamos en el máximo:

$$\text{Ln}(\mathcal{L}(\theta)) \cong \text{Ln}(\mathcal{L}(\hat{\theta})) - \frac{(\theta - \hat{\theta})^2}{2\hat{\sigma}^2} \quad (2.35)$$

donde hemos definido:

$$\hat{\sigma}^2 = - \left( \frac{\partial^2 \text{Ln}(\mathcal{L}(\theta))}{\partial \theta^2} \right)^{-1} \quad (2.36)$$

Note que si evaluamos la función de verosimilitud en  $\theta = \hat{\theta} \pm \hat{\sigma}$  obtenemos

$$\text{Ln}(\mathcal{L}(\hat{\theta} \pm \hat{\sigma})) = \text{Ln}(\mathcal{L}(\hat{\theta})) - 1/2 \quad (2.37)$$

Esto significa que a  $1\sigma$  de desviación estándar el logaritmo de la función de verosimilitud evaluada desde el máximo debe cambiar en  $1/2$ .

### 2.3. Ejercicios

1. Demostrar la Ecuación (2.4).
2. Programar un mal generador de números aleatorios y usar `drand48` en `Python` para reproducir la Figura [2.1].
3. La distribución Beta está dada por:

$$f(x; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1}, \quad 0 \leq x \leq 1 \quad (2.38)$$

donde  $\Gamma(n) = (n-1)!$ . Para  $f(x; 2, 4)$ , halle el área bajo la curva usando el método de aceptación y rechazo con una incertidumbre del 1 %.

4. Sea  $\mathcal{A}(x_1, x_2, \dots, x_n)$ , donde  $\mathcal{A} \sim \mathcal{N}(\mu, \sigma)$ . Muestre que los estimadores máximo verosímiles son:

$$\begin{aligned} \hat{\mu} &= \frac{1}{n} \sum_{i=1}^n x_i \\ \hat{\sigma}^2 &= \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \end{aligned} \quad (2.39)$$

5. Para el ejercicio anterior, muestre que la matriz Hessiana evaluada en los estimadores es:

$$H(\hat{\mu}, \hat{\sigma}^2) = \begin{pmatrix} -n(\hat{\sigma}^2)^{-1} & 0 \\ 0 & -\frac{n}{2}(\hat{\sigma}^2)^{-2} \end{pmatrix}$$

¿Es esta matriz definida positiva o negativa?

6. La siguiente integral multidimensional:

$$\int_0^1 \cdots \int_0^1 2^{-7} \left( \sum_{i=1}^8 x_i \right)^2 dx_1 dx_2 \dots dx_8, \quad (2.40)$$

tiene el valor exacto  $\frac{25}{192}$ , usando el método de MonteCarlo estime esta integral con tres cifras de precisión.

7. Sampling of the Likelihood function using the Metropolis Hastings algorithm (Frequentist approach). Implemente el algoritmo de Metropolis Hastings para muestrear la distribución de likelihood como función de sigma ( $\mathcal{L}(\sigma|data)$ ). Use esta distribución para estimar la desviación estándar en el intervalo de confianza del 68 % (i.e,  $\sigma_{\pm}$ ).
  - a) Descargue los datos usados en clase: MLikelihood Data.
  - b) Escriba la función de Likelihood asociada a la distribución Gausiana.
  - c) El código debe calcular el Likelihood asociado a los datos para cada  $\sigma$  en la cadena de Markov.



- d) La condición de aceptación es  $\alpha = \frac{\mathcal{L}(\sigma^*|data)}{\mathcal{L}(\sigma_0|data)}$ , donde  $\sigma^*$  corresponde al valor candidato y  $\sigma_0$  al valor actual.
  - e) Se sugiere un número de grande de iteraciones  $\sim 10^5$ .
  - f) Dibuje el Histograma. ¿Cuál es la mediana y los errores asimétricos de esta estimación?
  - g) Compare con el valor del  $\sigma$  obtenido en la clase (which is obtained using a Bayesian approach).
8. En general la varianza de estimadores es no calculable:

$$V(\hat{\theta}) = E(\hat{\theta}^2) - E(\hat{\theta})^2 \quad (2.41)$$

En el caso de la distribución exponencial tenemos un valor analítico dado por:

$$\begin{aligned} V(\hat{\theta}) &= \int_0^\infty \dots \int_0^\infty \left( \frac{1}{n} \sum_{i=1}^n x_i \right)^2 \frac{1}{\theta} e^{-x_1/\theta} \dots \frac{1}{\theta} e^{-x_n/\theta} dx_1 \dots dx_n \\ &\quad - \left[ \int_0^\infty \dots \int_0^\infty \left( \frac{1}{n} \sum_{i=1}^n x_i \right) \frac{1}{\theta} e^{-x_1/\theta} \dots \frac{1}{\theta} e^{-x_n/\theta} dx_1 \dots dx_n \right]^2 \\ &= \frac{\theta^2}{n} \end{aligned} \quad (2.42)$$

- a) Intente encontrar este resultado analíticamente.
- b) Con el método de MonteCarlo compruebe este resultado para un conjunto de  $n=5$  variables aleatorias y  $\theta = 2$ .



# References

- [1] Landau, Rubin H, Páez, José, Mejía, Manuel José Páez, and Bordeianu, Cristian C, *A survey of computational physics: introductory computational science*, Princeton University Press, 2008, <https://psrc.aapt.org/items/detail.cfm?ID=11578>