

# Universidad Nacional de Ingeniería



## Facultad de Ciencias

### Curso:

Calculo de Probabilidades

### Sección:

B

### Alumnos:

Penadillo Lazares, Wenses Johan  
Raymundo Muñoa, Abrahan  
Ventocilla Tamara, Franz Rony  
Yacolca Huamán, Jesús Miguel

### Códigos:

20182174D  
20180562G  
20180502D  
20183004E

### Profesor:

Lara Avila, Cesar

### Fecha:

24 de Setiembre del 2018



# Índice

## 1. Introducción

1.1. Presentación del problema.....	2
1.2. Vinculación con Python.....	2
1.3. Vinculación con el curso .....	2
1.4. Objetivo del estudio.....	2
1.5. Números aleatorios y Montecarlo.....	2
1.6. Integración.....	3
1.5. Ley de los grandes números .....	3

## 2. Estado del Arte

2.1. Artículos relacionados.....	4
2.1.1. Valuación de opciones por Montecarlo .....	4
2.1.2. Resolución de la ecuación de Laplace.....	4
2.1.3. Agujas de buffon.....	5

## 3. Diseño del experimento

3.1. Descripción de objetos, funciones y técnicas.....	5
3.2. Metodología.....	5

## 4. Experimentos y resultados .....

5
---

## 5. Discusión

5.1. Interpretación de resultados obtenidos.....	6
5.2. ¿Como se podría mejorar los resultados? .....	6

## 6. Conclusiones .....

6
---

## 7. Referencias.....

7
---

# Metodo Montecarlo

Penadillo Lazares, Wenses Johan  
20182174D  
*wpenadillol@uni.pe*

Ventocilla Tamara, Franz Rony  
20180502D  
*fventocillat@uni.pe*

Raymundo Muñoa, Abrahan  
20180562G  
*araymundom@uni.pe*

Yacolca Huamán, Jesús Miguel  
20183004E  
*jyacolcah@uni.pe*

## 1. Introducción

Los modelos de simulación son útiles para describir algunos fenómenos de la realidad. Así la simulación trata fundamentalmente de construir modelos abstractos en una computadora de tal forma que describan la parte esencial del comportamiento de un sistema de interés, pudiéndose entonces diseñar y realizar experimentos con el modelo y posteriormente extraer conclusiones de sus resultados. Los modelos de simulación tienen su origen en los trabajos de John Von Neumann, pero solo es apreciable su poder cuando se introduce el computador moderno.

Para la implementación de un modelo de simulación se hace necesario entonces, tener en primera instancia una fuente de números aleatorios que representen las observaciones del fenómeno, unas transformaciones de estas observaciones para alimentar el modelo, un procesamiento de las salidas del modelo y un análisis de las respectivas estadísticas resultantes.

### 1.1. Presentación del problema

En este trabajo el problema a tratar será "Integración por Montecarlo". Para esto se usará el método de Montecarlo, el cual nos ayudara a estimarlas mediante el uso de variables aleatorias. El problema principal de este método es la necesidad del uso de una gran cantidad de estas variables para poder minimizar el error. A causa de que este método funciona con la probabilidad esto es necesario. Antes se creía imposible, pero gracias a las computadoras se puede hacer. Por esto es necesario usar herramientas computacionales que nos ayuden a generar las variables y realizar los cálculos. De aquí que sea necesario el uso de el lenguaje de programación Python que cuenta con librerías necesarias para esto.

### 1.2. Vinculación con Python

Para la estimacion de las integrales es necesario usar el lenguaje de programación Python, el cual nos ayudará generar la gráfica de la función y los puntos aleatorios los cuales se usará para el método de Montecarlo. Al ser Python un lenguaje de alto nivel nos da las herramientas necesarias.

### 1.3. Vinculación con el curso

Todo lo expuesto anteriormente se vincula con el curso pues el Método de Montecarlo está relacionado con la probabilidad. Un mayor entendimiento de esto dará mayor solidez a conceptos más complejos del curso. También ayudará a ver cuales son las aplicaciones del mismo. Ya que es de gran utilidad en diferentes campos desde la economía hasta la física, pasando la computación y el aprendizaje de máquina.

### 1.4. Objetivo del estudio

- Estudiar los conceptos de la simulación Monte Carlo.
- Conocer algunas aplicaciones de la simulación Monte Carlo.
- Introducir este metodo a la estimacion de integrales.
- Mostrar como los números aleatorios permite hacer evaluaciones de integrales definidas.

### 1.5. Números aleatorios y Montecarlo

En el corazón de los Métodos de Montecarlo encontramos un generador de números aleatorios, es decir un procedimiento que produce un flujo infinito de variables aleatorias, que generalmente se encuentran en el intervalo  $(0, 1)$ ; los cuales son independientes y están uniformemente distribuidos de acuerdo a una distribución de probabilidad. La

mayoría de los lenguajes de programación hoy en día contienen un generador de números aleatorios por defecto al cual simplemente debemos ingresarle un valor inicial, generalmente llamado *seed* o semilla, y que luego en cada invocación nos va a devolver un secuencia uniforme de variables aleatorias independientes en el intervalo  $(0, 1)$ .

El concepto de una secuencia infinita de variables aleatorias es una abstracción matemática que puede ser imposible de implementar en una computadora. En la práctica, lo mejor que se puede hacer es producir una secuencia de números pseudoaleatorios con propiedades estadísticas que son indistinguibles de las de una verdadera secuencia de variables aleatorias.

Entonces se requiere sortear variables aleatorias según una distribución de probabilidad. Por ejemplo, para hallar  $\int f(x)g(x)dx$  los valores de  $X$  deben distribuirse según  $f(x)$  y la integral es el valor medio de  $g(x)$  sobre el conjunto de  $X$ . En este procedimiento es fundamental muestrear  $X$  siguiendo  $f(x)$ . Veamos qué quiere decir exactamente muestrear. Consideremos un espacio

$\Omega$  formado por todos los posibles valores de una o varias variables aleatorias  $X = X_1, X_2, \dots$  y sea  $f(x)$  su PDF, que cumple

$$\int_{\Omega} f(x)dx = 1$$

El muestreo es un algoritmo que produce una secuencia de variables aleatorias  $X$  tales que para cualquier  $\Omega' \subset \Omega$ ,

$$P(X \in \Omega') = \int_{\Omega'} f(x)dx \leq 1$$

En particular, si tenemos una sola variable aleatoria  $X$  que se distribuye según una PDF unidimensional definida sobre el intervalo  $[0, 1]$ , lo anterior significa que

$$P(X \in (a, b)) = \int_a^b f(x)dx, 0 < a < b < 1,$$

o, más informalmente, si  $dx = b - a$ ,

$$P(X \in dx) = f(x)dx$$

Podemos producir variables aleatorias  $X_1, X_2, \dots$  según una PDF cualquiera si disponemos de variables aleatorias  $\varepsilon_1, \varepsilon_2, \dots$  que se distribuyan uniformemente en el intervalo  $[0, 1]$ . Las variables aleatorias uniformemente distribuidas se imitan mediante un ordenador, pues siendo generadas por una rutina determinista, un generador de números pseudoaleatorios (prng), no son por tanto realmente aleatorias. Sin embargo, son satisfactorias

siempre que cumplan ciertos criterios de calidad. Estudiaremos los números pseudoaleatorios (prn) en esta sección.

## 1.6. Integración

Sea  $f(x)$  una función, deseamos resolver entonces el problema de calcular la siguiente integral:

$$\theta = \int_0^1 f(x)dx$$

Para calcular el valor de la integral observemos que si  $U$  es una variable aleatoria distribuida uniformemente en el intervalo  $(0, 1)$ , entonces es posible expresar a  $\theta$  como:

$$\theta = E(f(u))$$

Si  $U_1, \dots, U_p$  son Variables aleatorias independientes y uniformes, entonces  $g(U_1), \dots, g(U_p)$  son variables aleatorias independientes e idénticamente distribuidas con valor esperado  $\theta$ . Utilizando la ley fuerte de los grandes números que enuncia que con probabilidad 1.

$$\sum_{i=1}^p \frac{f(U_i)}{p} \rightarrow E(f(u)) = \theta$$

Por consiguiente, es posible aproximar el valor de la integral, generando una gran cantidad de números aleatorios  $U_i$ , evaluar  $f(x)$  en cada  $U_i$ , sumar estas cantidades y dividir entre el número de números aleatorios; obteniendo entonces una aproximación al verdadero valor de la integral cuando la cantidad de números aleatorios tiende al infinito.

## 1.7. Ley de los grandes números

Aunque el tema central de este documento es la integración a través del uso de números aleatorios, es conveniente hacer algunas claridades y justificaciones sobre el teorema que sustenta todo el método.

Primero que todo se debe mostrar un teorema de Concentración de distribuciones llamado "Desigualdad de Chebyshev" que va a ser la base de todo nuestro análisis.

Si  $X$  es una variable aleatoria con media  $\mu$  y varianza  $\sigma^2$ , entonces para cualquier valor de  $k > 0$  se tiene que

$$P[|X - \mu| \geq k\sigma] \leq \frac{1}{k^2} \quad (1)$$

Esta desigualdad muestra que la cantidad de área de la distribución, situada fuera del intervalo  $\mu -$

$k\sigma \leq x \leq \mu + k\sigma$  es como mínimo  $1 - \frac{1}{k^2}$ , dando así una idea de la concentración de una variable aleatoria sin importar como se distribuya.

Ahora si se tiene una sucesión de variables aleatorias  $X_1, X_2, \dots, X_n$  Independientes con media  $\mu$  y desviación estándar  $\sigma^2$  se tiene para cada  $\varepsilon > 0$ , aplicando la desigualdad de Chebyshev.

$$P\left[\left|\frac{\sum x_i}{n} - \mu\right| \geq \frac{k\sigma}{\sqrt{n}}\right] \leq \frac{1}{k^2} \quad (2)$$

Si se reemplaza  $\frac{k\sigma}{\sqrt{n}}$  por  $\varepsilon$  se tiene

$$P\{|\bar{X} - \mu| \geq \varepsilon\} \leq \frac{\sigma^2}{n\varepsilon^2} \quad (3)$$

Si  $n \rightarrow \infty$  en (3), se convierte en

$$P\{|\bar{X} - \mu| \geq \varepsilon\} = 0 \quad (4)$$

De este modo, la probabilidad de que  $\bar{X}$  difiera de la media  $\mu$  en mas de  $\varepsilon$ , con epsilon pequeño, tiende a ser igual a cero a medida que aumente el tamaño de la muestra.

Este teorema llamado "Ley Débil de los Grandes Números" sería suficiente para nosotros si lo que interesara fuera la convergencia en probabilidad de

$$\frac{\sum f(U_i)}{n} \rightarrow E(f(u)) \quad (5)$$

En realidad, el interés radica en la convergencia casi segura del promedio de variables aleatorias.

O en otras palabras el problema radica en garantizar la convergencia

$$\lim_{n \rightarrow \infty} \frac{\sum f(U_i)}{n} = E(f(u)) \quad (6)$$

Afortunadamente el problema fue solucionado ya por los matemáticos a través de un teorema llamado "Ley Fuerte de los Grandes Números", que enuncia que bajo algunas consideraciones muy generales el promedio de variables aleatorias de una muestra converge con probabilidad 1 a la esperanza matemática de la variable en cuestión.

Así es la "Ley Fuerte de los Grandes Números", el teorema que justifica el método de integración Monte Carlo como llaman algunos autores.

## 2. Estado del arte

### 2.1. Artículos relacionados

#### Valuación de opciones por Montecarlo

(El método Montecarlo y su aplicación a finanzas  
Patricia Saavedra Barrera y Víctor Hugo Ibarra  
Mercado)

(ver más en referencias 6).

### Integración por el método de Montecarlo. Resolución de la ecuación de Laplace

Considere la ecuación de Laplace en un dominio suave  $\Omega \subset R^2$ :

$$\Delta u(x, y) = 0, (x, y) \in \Omega \dots (*)$$

$$u(x, y) = g(x, y), (x, y) \in d\Omega \dots (*)$$

donde  $\Delta u = u_{xx} + u_{yy}$  es el Laplaciano de  $u$ .

Esta ecuación modela por ejemplo el potencial eléctrico dentro de  $\Omega$  cuyo contorno se encuentra a potencial  $g$ , o el estado estacionario de la temperatura en cuando el borde se mantiene a temperatura  $g$ .

El algoritmo de Montecarlo para aproximar  $u$  consiste en lo siguiente. Al introducir una discretización por diferencias finitas de (\*), sobre una malla suficientemente fina, también introducimos probabilidades de transición entre los nodos de esta malla. Esto se debe a que la misma manera de discretizar está asignándole un peso a cada nodo vecino en el cálculo del valor de la solución de la ecuación en un nodo dado. La idea del método es que para conocer el valor de la solución  $u$  en un nodo  $(x_i, y_i)$  de la malla, se larga una partícula desde ese nodo y se la hace evolucionar de acuerdo a las probabilidades de transición calculadas, hasta que choca con el borde, almacenándose el valor del dato de contorno en ese punto. Se repite este procedimiento con una grande cantidad  $M$  de partículas, y se estima el valor de  $u(x_i, y_i)$  como el promedio de esos valores.

En primer lugar puede aplicar este método en el dominio rectangular  $\Omega = [0; 1] \times [0; 1]$  con  $u(x, y) = \sin(\pi x)e^{\pi y}$  (*i.e.*  $g(0, \cdot) = g(1, \cdot) = 0, g(x, 0) = \sin(\pi x), g(x, 1) = e^{\pi} \sin(\pi x)$ ):

1. Defina una malla rectangular:  $x_i = \frac{i}{N}, y_i = \frac{i}{N}$ ,  $0 \leq i \leq N$ , con  $N$  un valor adecuado (podría comenzar con  $N = 10$ , por ejemplo).

2. La discretización habitual del Laplaciano conduce a la siguiente ecuación:

$$\Delta u(x_i, y_j) \simeq \frac{u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1} - 4u_{i,j}}{h^2} = 0,$$

donde  $u_{i,j} = u(x_i, y_j)$  lo que da las siguientes probabilidades de transición: una partícula en el nodo  $(x_i, y_j)$  tiene probabilidad  $\frac{1}{4}$  de pasar a cualquiera de los cuatro nodos adyacentes.

3. Utilice estas probabilidades de transición y el

procedimiento descrito anteriormente para calcular la solución en el punto  $(x, y) = (\frac{1}{2}, \frac{1}{2})$  por ejemplo. Compare con la solución exacta  $u(\frac{1}{2}, \frac{1}{2}) \simeq 4,8$ . Repita este cálculo para varios valores de  $N$ , y para varios números diferentes de partículas (valores de  $M = 1000$  o mayores, podrían ser adecuados). Haga gráficos del error en función de  $N$  y de  $M$ .

En segundo lugar, puede aplicar el método para hallar la solución en la corona circular  $\Omega = \{1 \leq r \leq 3, 0 \leq \theta < 2\pi\}$  con dato de contorno  $g(1, \theta) = 4, g(3, \theta) = 6$  (donde  $(r, \theta)$  son las coordenadas polares).

(ver más en referencias 10).

### Las agujas de buffon

(Office for Mathematics, Science, and Technology Education

College of Education@University of Illinois

MSTE: Buffon's Needle)

Un problema planteado en el s.XVIII por George Louis Leclerc, conde de Buffon, fue el siguiente: Se tiene rectas paralelas equidistantes entre sí, y se arroja una aguja de longitud mayor o igual a la distancia entre dos rectas.

¿Cuál es la probabilidad de que una aguja corte a una de las rectas?

Fijando algunas constantes y variables, para la resolución del problema

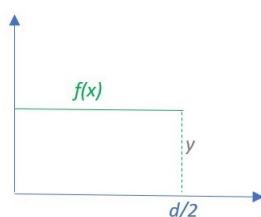
- $d$  la distancia entre las rectas.
- $l$  la longitud de la aguja.
- $\theta$  la medida del ángulo agudo entre la aguja (o su prolongación) y una de las rectas.
- $x$  la distancia entre el punto medio de la aguja y la recta más cercana.

$x$  y  $\theta$  son v.a con distribución uniforme

$$0 \leq x \leq \frac{d}{2}$$

$$0 \leq \theta \leq \frac{\pi}{2}$$

Procediendo a hallar la función densidad  $f(x)$  y  $g(\theta)$  de ambas variables



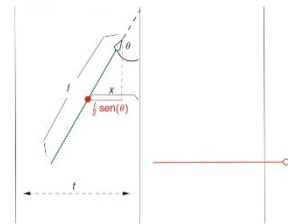
Para  $f(x)$ , ya que  $\int f(x)dx = 1$ , entonces  $\frac{d}{2}f(x) = 1$

Donde se obtiene:  $f(x) = \frac{2}{d}$

Similarmente, se obtiene para  $g(\theta)$ :  $g(\theta) = \frac{2}{\pi}$

Luego, la función densidad conjunta, de ambas v.a.i es  $f(x)g(\theta) = \frac{2}{d} \frac{2}{\pi} = \frac{4}{d\pi}$

Además, se observa de la gráfica, que una aguja cortará la recta si y solo si la distancia de su centro a una de las rectas es menor que  $\frac{l}{2}\sin(\theta)$ , es decir  $x < \frac{l}{2}\sin(\theta)$



Posteriormente se calculara la  $P$ (la aguja corte la recta) como la integral de la función densidad conjunta.

$$P = \int_0^{\frac{\pi}{2}} \int_0^{\frac{l}{2}\sin(\theta)} \frac{4}{d\pi} d\theta dx$$

$$P = \frac{4}{d\pi} \int_0^{\frac{\pi}{2}} \int_0^{\frac{l}{2}\sin(\theta)} d\theta dx = \frac{4}{d\pi} \int_0^{\frac{\pi}{2}} \frac{l\sin(\theta)}{2} d\theta$$

$$P = \frac{4l}{2d\pi} [-\cos(\theta)]_0^{\frac{\pi}{2}} = \frac{2l}{\pi d}$$

(ver más en referencias 7,8 y 9).

## 3. Diseño del experimento

### 3.1. Descripción de objetos, funciones y técnicas

Dentro de los objetos tenemos a los números aleatorios los cuales son la base del método Montecarlo. Del lenguaje Python una de las funciones que usaremos será *Random* la cual nos servirá para la generación de los números aleatorios, unas de las librerías que usaremos para mostrar los gráficos será *matplotlib*, y para obtener datos *numpy* y *pandas*. El método que usaremos será el de Montecarlo, el cual junto con los números aleatorios, el lenguaje Python y sus librerías nos servirá para estimar las integrales.

### 3.2. Metodología

Habiendo abordado ya las cuestiones técnicas referentes a la justificación de los teoremas de convergencia se proceden a ilustrar con un ejemplo el proceso de cálculo de integrales a través de números aleatorios.

Supongamos que se requiere calcular el valor de  $\int_0^1 x^3 dx$ , es evidente que esta integral da como resultado  $\frac{1}{4}$  pero sirve como ejemplo. El proceso entonces es el siguiente:

- Generar una gran cantidad de números aleatorios entre 0 y 1 (para nuestro ejemplo  $n = 1000$ ).
- Evaluar  $f(U_i)$  (en el ejemplo  $U_i^3$ ).
- Sumar y dividir entre  $n$ ,

$$\frac{\sum f(U_i)}{n} = \frac{\sum_{i=1}^{1000} U_i^3}{1000}$$

- Aproximar la integral al resultado

$$\hat{\theta}'' = \frac{\sum f(U_i)}{n} \rightarrow \int_0^1 f(x) dx$$

En el ejemplo de integrar  $x^3$  entre 0 y 1 se utilizó una serie de números aleatorios uniformes generados en Python con una aproximación de 0,254 con un error del 1,6 % con respecto al verdadero valor de la Integral.

La cuestión radica ahora en encontrar alguna transformación apropiada para calcular cualquier integral definida.

Con  $dz = -\frac{1}{(x+1)^2} dx = -z^2 dx$  y obteniendo

$$\theta = \int_0^1 \frac{1}{z^2} f\left(\frac{1}{z} - 1\right) dz$$

Este método de evaluar integrales puede ser mejorado con algunas técnicas de reducción de varianza, como lo son de las variables antitéticas, variables de control, entre otros.

$$\theta = \int_0^1 f(x) dx$$

Si se hace la sustitución:

$$Z = \frac{x-a}{b-a}$$

$$dz = \frac{dx}{b-a}$$

Se tiene que

$$\theta = \int_0^1 f(z(b-a) + a)(b-a) dz$$

O en otros términos

$$\theta = \int_0^1 g(z) dz$$

Con  $g(z) = (b-a)f(a + (b-a)z)$

Y queda solucionando el problema de la integral con límites  $[a, b]$  de integración.

Así sustituciones análogas permiten resolver las integrales impropias tipo

$$\theta = \int_0^\infty f(x) dx$$

Haciendo simplemente la sustitución

$$z = \frac{1}{x+1}$$

## 4. Experimentos y resultados

(ver en referencia 3.)

## 5. Discusión

### 5.1. Interpretación de resultados obtenidos

Luego de haber realizado la simulación un gran número de veces, notamos que el valor de las integrales obtenidas por el método Montecarlo tienen una buena aproximación con un margen de error y tiempo de cálculo aceptables.

### 5.2. ¿Como se podría mejorar los resultados?

Debido a que el método se basa en el uso de una gran cantidad de números aleatorios, el resultado será mejor cuanto más grande sea la cantidad de números usados. En las pruebas realizadas usamos 100000000 números aleatorios con un tiempo de cálculo de 4 min. Estos resultados se pueden mejorar usando una super computadora que pueda soportar una gran cantidad de números aleatorios con un tiempo de cálculo menor.

## 6. Conclusiones

Aunque es cierto que un método numérico tradicional como la regla de Simpson es más eficiente para resolver integrales unidimensionales, resulta ineficaz para una integral multidimensional y es aquí donde es poderoso el uso de números aleatorios, así para evaluar integrales de orden mayor o igual a cinco 5 es superior el método de números

aleatorios que la regla del trapecio y para dimensiones superiores a ocho 8 los números aleatorios se vuelven superiores a cualquier método numérico de integración.

## 7. Referencias

1. <https://relopezbriega.github.io/blog/2017/01/10/introduccion-a-los-metodos-de-monte-carlo-con-python/>
2. [https://www.uam.es/personal\\_pdi/ciencias/carlosp/html/pid/montecarlo.html](https://www.uam.es/personal_pdi/ciencias/carlosp/html/pid/montecarlo.html)
3. <https://github.com/wencez432/Metodo-de-Montecarlo>
4. <http://www.redalyc.org/pdf/849/84911652058.pdf>
5. <https://www.ugr.es/~jillana/Docencia/FM/mc.pdf>
6. <http://mat.izt.uam.mx/mat/documentos/notas%20de%20clase/cfenaoe3.pdf>
7. [http://sistemas.fciencias.unam.mx/~erhc/Apunte\\_21\\_de\\_septiembre.pdf](http://sistemas.fciencias.unam.mx/~erhc/Apunte_21_de_septiembre.pdf)
8. <http://www.ing.unne.edu.ar/assets/pdf/academica/departamentos/computacion/montecarlo.pdf>
9. <https://mste.illinois.edu/activity/buffon/>
10. [http://www.dm.uba.ar/materias/elementos\\_calculo\\_numerico\\_M/2007/2/proyectos/montecarlo.pdf](http://www.dm.uba.ar/materias/elementos_calculo_numerico_M/2007/2/proyectos/montecarlo.pdf)