



**Métodos de Monte Carlo
Introducción y Conceptos Básicos**

Marco Villacañas

Andrés Moguel

Inferencia Estadística I

Dr. Nelson Muriel

Licenciatura en Actuaría

Semestre Otoño 2021

15 de octubre de 2021

Planteamiento

Los problemas y aplicaciones numéricas de la estadística inferencial son aquellos de optimización e integración. A través de métodos y aproximaciones como la máxima verosimilitud, el Bayesiano, método de momentos o incluso los métodos de bootstrap, se puede llegar a calcular analíticamente un estimador. Sin embargo, hay algunos problemas en los que es muy difícil o imposible llegar a calcular estos estimadores asociados a un paradigma dado.

Introducción a Métodos de Monte Carlo

Los Métodos de Monte Carlo (MC) son técnicas computacionales que utilizan, de manera esencial la generación de números aleatorios para el análisis de distintos fenómenos. Estos métodos abarcan un gran número de técnicas de simulación estocástica para la resolución de problemas de índole científica relacionados con optimización y estadística. A este método también se le conoce como simulación de probabilidad múltiple. Los MC son algoritmos computacionales que utilizan un proceso repetitivo de muestreos aleatorios para generar aproximaciones numéricas de parámetros desconocidos. Esta simulación de números aleatorios permite estimar ciertas funciones de una distribución de probabilidad. Utilizando estas técnicas es posible modelar situaciones complejas en las que interactúan e intervienen distintas variables aleatorias.

A diferencia de un modelo de estimación normal, la simulación de Monte Carlo estima un conjunto de resultados basado en un rango estimado de valores en vez de hacerlo con un conjunto de entrada fijo. En un experimento clásico de Monte Carlo esto se puede repetir una infinidad de veces para producir un gran número de posibles resultados. Al completarse el proceso, las salidas obtenidas son un rango de posibles valores con su respectiva probabilidad de ocurrencia. Estas simulaciones invierten el problema clásico de la estadística: en lugar de estimar cantidades aleatorias de manera determinística, emplea cantidades aleatorias para llegar a un valor estimado de cantidades determinísticas.

Procedimiento Básico de Aplicación

Hay una amplia gama de técnicas de simulación, pero todas tienen como punto de partida indispensable, la generación de valores numéricos aleatorios para la resolución de problemas determinísticos. Para aplicar correctamente los distintos métodos es indispensable tener una buena fuente de números aleatorios o utilizar un método adecuado para su generación.

Esencialmente los MC comienzan obteniendo un gran conjunto de potenciales valores de los parámetros deseados y sustituyendo las integraciones por medias muestrales. El proceso se puede resumir en lo siguiente:

1. Proponer el modelo predictivo, identificando la variable dependiente, que se busca estimar, y las variables independientes que impulsarán la simulación. A estas variables independientes se les conoce como variables predictores de riesgo.
2. Especificar las distribuciones y densidades de probabilidad de las distintas variables independientes. Se debe utilizar información histórica o el conocimiento que tenga el

analista acerca de las variables para determinar un rango adecuado de posibles valores que pueden tomar estas variables y asignarles su probabilidad de ocurrencia respectiva.

3. Correr simulaciones repetidamente; generando así valores aleatorios de las variables independientes. Se deben correr tantas simulaciones como sea necesario hasta obtener una muestra representativa suficiente de todas las posibles combinaciones.

Estas técnicas se usan para entender el impacto del riesgo e incertidumbre en la predicción y sus distintos modelos. Al generar un posible rango de valores, estas predicciones son más precisas que las de un modelo de predicción normal que tiene valores fijos de entrada. También pueden ser utilizados para predicciones de largo plazo. Al incrementar la cantidad de variables de entrada, aumenta el número de posibles resultados. Esto permite hacer predicciones hacia un tiempo futuro más lejano.

Marco Histórico

Los métodos Monte Carlo engloban una gran cantidad de técnicas de simulación estocástica que en gran medida son utilizados para resolver problemas de optimización e inferencia: El origen de los métodos Monte Carlo se remontan a los experimentos de Buffon para calcular un valor empírico del *St. Petersburg game*, mide el valor de el ahora conocido “Buffon’s needle”. Algunos estadísticos intentaron experimentar diferentes métodos para la generación de números aleatorios usando: Cartas, ruleta o dados, para verificar empíricamente, a través de algún tipo de simulación estocástica primitiva, sus complicados procedimientos estadísticos.

El desarrollo formal de la teoría de Método Carlo como se conoce hoy en día sucedió hasta los años 1940 – 1950 y esto se le atribuye a Stanislaw Ulam. Un matemático de nacionalidad polaca que al estar jugando un juego de cartas “solitario” tratando de calcular la probabilidad de éxito, se dio cuenta que una forma de hacerlo sería jugar un cierto número de manos y calcular la probabilidad de éxito. Sin embargo, se enteró de la existencia de unas nuevas computadoras, las ENIAC. Buscó a Von Neumann, consultor en Los Álamos y el Laboratorio de Investigaciones Balísticas, quien reconoció de inmediato la relevancia de la idea de Ulam y esbozó un enfoque para resolver problemas de difusión / multiplicación de neutrones mediante muestreo estadístico por computadora en una carta de 1947. El método se probó con éxito y fue nombrado “Montecarlo”, se desarrollaron ordenadores más potentes (como el MANIAC en Los Álamos, y muchos físicos comenzaron a usar métodos de MC basados en computadora para obtener soluciones aproximadas a sus problemas. Los métodos de MC requerían una gran cantidad de números aleatorios, y el desarrollo del número aleatorio esencial. Algunas técnicas de Monte Carlo fueron utilizadas en el Proyecto Manhattan por Von Neumann, Ulam y Metropolis, para modelar las reacciones en cadena del uranio e hidrógeno dentro de las bombas atómicas. Los generadores requeridos por los métodos MC también se iniciaron durante esos años. Por ejemplo, Von Neumann describió el Método de muestreo de rechazo (RS) 1947 a Ulam (aunque no se publicó hasta 1951 y Lehmer introdujo el número aleatorio congruencial lineal generadores en 1951 el muestreo estadístico fue el desarrollo del algoritmo Metropolis-Hastings (MH).

El algoritmo MH fue diseñado inicialmente por Nicholas Metropolis et al. En 1953 como método general para acelerar el cálculo de las propiedades de sustancias compuestas por interacción de

moléculas individuales. La idea del algoritmo es bastante simple: movimientos aleatorios de partículas distribuidos uniformemente en torno a su posición actual se propusieron; si el global la energía del sistema disminuyó, estos movimientos fueron siempre aceptados; de lo contrario, fueron aceptados solo con alguna probabilidad no nula que dependía de la energía aumentada (cuanto mayor sea el aumento, menos probable es que el movimiento sea aceptado). Los movimientos rechazados también se utilizaron para calcular los promedios deseados. Metrópolis y col. demostraron que el método era ergódico y las muestras se extrajeron de la distribución deseada. Este enfoque puede verse como un Cadena de Markov, con un paso de muestreo RS en el núcleo para asegurarse de que la cadena tenga la función de densidad de probabilidad invariante deseada (PDF) y, por lo tanto, la cadena de Markov Nacieron los métodos Monte Carlo (MCMC). Se consideró una densidad de propuesta simétrica. En 1970, Hastings mostró que las densidades de propuesta no simétricas también se habían utilizado lo que permite mucha más flexibilidad en el método, y propuso una probabilidad de aceptación genérica que garantiza la ergonomía de la cadena. Esto permitió que este algoritmo fuera aún más conocido, aunque la comunidad de los ingenieros y estadísticos lo conocieran hasta la década de los 90.

El muestreador de Gibbs Andrew Gelman mostró en 1992 que el muestreo de Gibbs era un caso particular del algoritmo MH, lo que provoca un renovado interés en el Algoritmo MH por estadísticos. Mostraba cómo podría ser utilizado para tratar con distribuciones no estándar en Inferencia Bayesiana. Explicaciones sencillas del muestreador de Gibbs y el algoritmo MH también aparecieron en la década de 1990, estos métodos poco a poco comenzaron a aplicarse a los todo tipos de problemas: Econometría, bioestadística, inferencia.

Aplicaciones y Usos

Ejemplos Aplicados Básicos

Los primeros experimentos de Monte Carlo se centran en estimar el valor de π . Se puede definir el problema de la aguja de Buffon (*Buffon's Needle*). Consideremos dejar caer una aguja de longitud L uniformemente sobre un conjunto de líneas paralelas separadas una de la otra por una distancia $D > L$. En la figura 1 vemos una ilustración de este planteamiento. La pregunta que se plantea es la siguiente: Determinar la probabilidad de que una de las agujas que se dejan caer, interseca una de las líneas paralelas. Para resolver el problema se comienza asumiendo que el conjunto de líneas paralelas es lo suficientemente grande para que los efectos de las esquinas no afecten el resultado. También se asume que las coordenadas (x, y) del centro de las agujas son uniformes sobre el conjunto y que la orientación de cada aguja se distribuye de manera uniforme sobre el intervalo $[0, \pi)$.

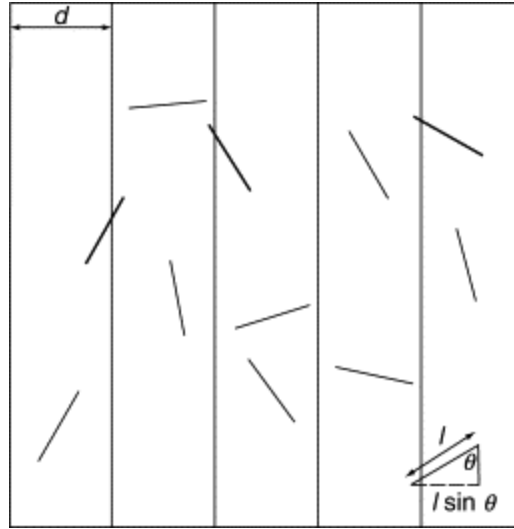


Figura 1. Agujas de Buffon. En negritas son las agujas que intersectan una línea paralela.

La figura 1 muestra que una aguja que intersecta alguna de las líneas paralelas es equivalente a un rectángulo con un ancho de $L \sin(\theta)$ intersectando esa misma línea. Esto sucede, para un θ dado, con probabilidad:

$$P(\text{intersección} \mid \theta) = \frac{L \sin(\theta)}{D}$$

Dado que el ángulo θ es uniforme en $[0, \pi)$ se tiene:

$$\begin{aligned} P(\text{intersección}) &= \int_0^{\pi} P(\text{intersección} \mid \theta) \frac{1}{\pi} d\theta \\ &= \frac{L}{\pi D} \int_0^{\pi} \sin(\theta) d\theta \\ &= \frac{L}{\pi D} [-\cos(\pi) + \cos(0)] = \frac{2L}{\pi D} \end{aligned}$$

Si se considera que se dejan caer n agujas y el experimento se repite muchas veces, entonces el número promedio de agujas que intersectan con una línea es de:

$$\frac{2nL}{\pi D}$$

Considerando un escenario de Monte Carlo esto quiere decir que podemos estimar el valor de π con la siguiente ecuación.

$$\hat{\pi} = \frac{2nL}{MD}$$

Dónde π en negritas es el estimado y M es el número de agujas que cruzan una línea al dejar caer n .

Esta estimación de π se puede también ver como un juego de dardos. Si el tablero está dentro de un cuadrado que tiene lados de longitud $2r$, por lo tanto, el tablero tiene radio r . En este caso la ubicación de los dardos lanzados se distribuye de manera uniforme. La probabilidad de que un dardo lanzado caiga dentro del tablero es el cociente del área del tablero y el cuadrado que lo contiene:

$$P(\text{Clavar en el Tablero}) = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4}.$$

Al expresar π como una función de dicha probabilidad podemos usar los estimadores, en este caso p , para aproximar π . Al desconocer π , no podemos llegar al valor exacto de p y por lo tanto debemos estimar p como \hat{p} . Esto se puede lograr contando la proporción de lanzamientos que caen dentro del tablero. Si se realizan n lanzamientos y x de ellos se clavan en el tablero entonces se puede estimar p como:

$$\hat{p} = \frac{x}{n}$$

En este caso podemos ver que dado que hay x éxitos de n observaciones se está trabajando con una variable aleatoria de distribución $\text{Bin}(n, p)$. De la relación entre π y p tenemos:

$$\hat{\pi} = \frac{(x)^4}{n}$$

Dados todos los elementos anteriores, podemos escribir un programa en Python (véase anexo 1) que permite aproximar π de esta manera. En este caso se asume un círculo de $r = 1$.

Para $n = 10$

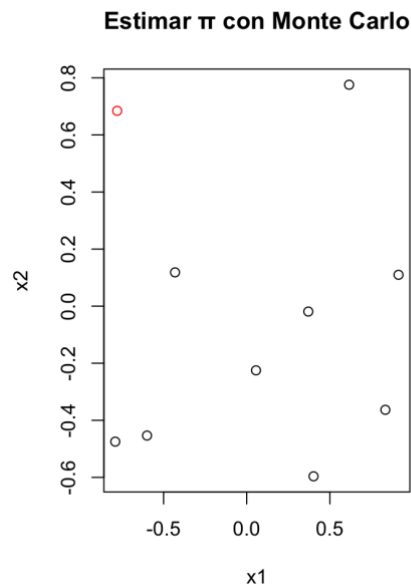


Figura 2: Aproximación de π para $n = 10$, en negro caen dentro del círculo, en rojo caen fuera. Aproximación de $\hat{\pi} = 3.6$. Se puede observar que, con pocas pruebas o lanzamientos, la estimación de π es poco exacta. Esto se debe a que se toma como una distribución uniforme.

Para $n = 100$

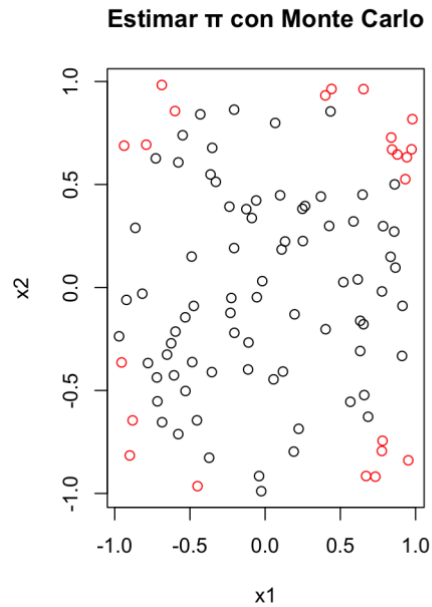


Figura 3: Aproximación de π para $n = 100$, en negro caen dentro del círculo, en rojo caen fuera. Aproximación de $\hat{\pi} = 3.08$. Esta aproximación se acerca más a π al incrementar n .
Para $n = 1,000$

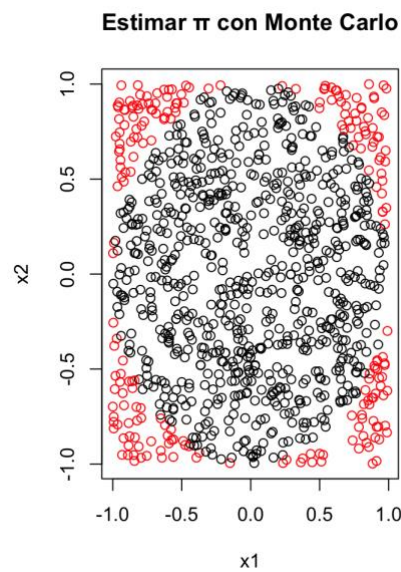


Figura 4: Aproximación de π para $n = 1000$, en negro caen dentro del círculo, en rojo caen fuera. Aproximación de $\hat{\pi} = 3.104$. Esta aproximación se acerca más a π al incrementar n . Se puede apreciar de manera más definida el círculo en negro y el cuadrado que lo encierra en rojo.

Para $n = 10,000$

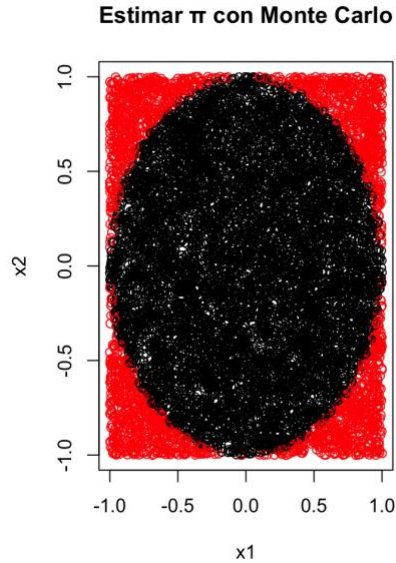


Figura 5: Aproximación de π para $n = 10,000$, en negro caen dentro del círculo, en rojo caen fuera.

Aproximación de $\hat{\pi} = 3.1276$. Esta aproximación ya tiene muchas más decimales y tiene una mayor precisión que todas las anteriores.

De las figuras anteriores podemos concluir que si $n \rightarrow \infty$ entonces el valor de π estimado se aproxima a su valor real.

Estos ejemplos pueden parecer triviales o muy sencillos, pero son esenciales para poder entender la idea base de en qué consisten los Métodos de Monte Carlo. Los MC se pueden utilizar para modelar sistemas muy complicados o de mayores dimensiones. Esto es muy útil ya que en aplicaciones al mundo real los valores que se buscan estimar si son desconocidos, no como el caso de π que si conocemos su valor real. Los Métodos de Monte Carlo son muy versátiles y son utilizados en varias disciplinas.

Aplicaciones en el Mundo Real

Las técnicas de Monte Carlo son interdisciplinarias, se usan principalmente en estos campos de estudio:

Estadística:

Comparar las estadísticas de la competencia para muestras pequeñas de datos en condiciones realistas, pruebas de hipótesis, pruebas de permutación y aleatorización aproximada. Se han utilizado para hacer revisiones de otros métodos de inferencia como la verosimilitud, el método de momentos, métodos de bootstrap, entre otros.

Medicina:

Son utilizados en el diseño de fármacos al simular la conformación y comportamiento de moléculas. El análisis se basa en los procesos de energía del cuerpo. Estas técnicas de simulación se pueden utilizar para simulaciones de proteínas y sus respectivos vínculos.

Ingeniería:

Ingeniería física:

Se utilizan para correr simulaciones del *Standard Model* para simular el mundo cuántico y la colisión de partículas.

En ingeniería aeroespacial:

Se utiliza para asegurar que múltiples partes de un ensamblaje encajen en un componente del motor.

Finanzas:

Evaluación de inversiones en proyectos a niveles corporativos, evaluación de derivados financieros, donde las simulaciones se agregan las estimaciones para los peores casos o viceversa y la duración de cada tarea para determinar los resultados para el conjunto del proyecto. En entregas posteriores se desarrollará más a fondo.

Referencias

- Gentle, J. E. (2010). Computational statistics. *International Encyclopedia of Education*, 93–97.
- Hoffman, J. I. E. (2019). Resampling statistics. *Basic Biostatistics for Medical and Biomedical Practitioners*, 631–637.
- IBM Cloud Education. (2020, August 24). *What is Monte Carlo Simulation?* IBM. Retrieved October 15, 2021, from <https://www.ibm.com/cloud/learn/monte-carlo-simulation#toc-how-does-m-7eGyxMlw>.
- Johansen, A. M. (2010). Markov chain Monte Carlo. *International Encyclopedia of Education*, 245–252.
- Johansen, A. M. (2010). Monte Carlo Methods. *International Encyclopedia of Education*, 296–303.
- Kenton, W. (2021, October 4). *What is a Monte Carlo Simulation?* Investopedia. Retrieved October 15, 2021, from <https://www.investopedia.com/terms/m/montecarlosimulation.asp>.
- Luengo, D., Martino, L., Bugallo, M., Elvira, V., & Särkkä, S. (2020). A survey of Monte Carlo methods for parameter estimation. *EURASIP Journal on Advances in Signal Processing*, 2020(1).
- Pease, C. (2018, September 11). *An overview of Monte Carlo methods*. Retrieved October 15, 2021,
- Robert, C. P., & Casella, G. (2005). *Monte Carlo Statistical Methods*. Springer.
- Robert, C., & Casella, G. (2010). *Introducing Monte Carlo methods with R*. Springer.
- Shonkwiler, R. W., & Mendivil, F. (2009). Introduction to monte carlo methods. *Undergraduate Texts in Mathematics*, 1–49.

Anexo 1

Código en R para la estimación de Pi mediante MC

```
1 #para poder reproducir
2 set.seed(5)
3 # repeticiones del experimento (dardos lanzados)
4 n=10000
5 # hacemos las coordenadas x1 y x2 de una distribución uniforme de[-1, 1]
6 x1<-runif(n, min=-1, max=1)
7 x2<-runif(n, min=-1, max=1)
8 #Distancia de los puntos al centro del círculo (0,0)
9 z<-sqrt(x1^2+x2^2)
10 # el area del círculo (cae dentro) es todo z
11 # en este caso deben ser menores al radio^2 para estar dentro del círculo
12 # in our case radius=1
13 piest<-4*sum((z<=1))/length(z)
14 print(piest)
15 #graficamos dentro o fuera del círculo por color.
16 InOut<-as.factor(ifelse(z<=1, "In", "Out"))
17 plot(x1,x2, col=InOut, main="Estimar  $\pi$  con Monte Carlo")
```

1:23 (Top Level) ↕ R Script ↕