



# **Cadenas de Markov Chain Montecarlo (MCMC) con el método de Metropolis**

**Diana Hernández Bustos  
Diana Rosales Rosales  
Repositorio: Tiro al blanco**

0712/2018  
Programación Básica  
Alma Xóchitl González

## INTRODUCCIÓN

El método de Montecarlo proporciona soluciones aproximadas a una gran variedad de problemas matemáticos posibilitando la realización de experimentos con muestreos de números pseudoaleatorios en una computadora. Este método fue inventado con el objetivo de simular la naturaleza aleatoria de los procesos físicos reales.

## MARCO TEÓRICO

La importancia actual del método Montecarlo se basa en la existencia de problemas que tienen difícil solución por métodos exclusivamente analíticos o numéricos, pero que dependen de factores aleatorios o se pueden asociar a un modelo probabilística artificial (resolución de integrales de muchas variables, minimización de funciones, etc.).

En la práctica este análisis consiste en ejecutar varias veces los diferentes sucesos variando aleatoriamente su valor en función de la función estadística que los define, dando como resultado un conjunto de valores finales. Este conjunto de valores permite calcular el valor medio y la variabilidad para el conjunto.

Dado un valor inicial, se construye una cadena de Markov a través de una densidad condicional la cual contiene una densidad invariante.

Los métodos de Montecarlo se utilizan para calcular, aproximar y simular expresiones o sistemas matemáticos complejos y difíciles de evaluar mediante un sistema no determinista o estadístico numérico.

En las estadísticas y en física estadística, el algoritmo de Metropolis-Hastings es una cadena de Markov Monte Carlo método para la obtención de una secuencia de (MCMC) muestras aleatorias a partir de una distribución de probabilidad para el que el muestreo directo es difícil. Esta secuencia se puede utilizar para aproximar la distribución (es decir, para generar un histograma), o para calcular una integral (tal como un valor esperado). Otros algoritmos MCMC Metropolis-Hastings y se utilizan generalmente para el muestreo de distribuciones multidimensionales, especialmente cuando el número de dimensiones es alta. Para las distribuciones unidimensionales, otros métodos están generalmente disponibles (por ejemplo, rechazo de muestreo adaptativo) que pueden volver directamente muestras independientes de la distribución, y están libres del problema de las muestras de autocorrelación que es inherente a los métodos MCMC.

Se utiliza una minimización de  $X^2$

$$X^2 = \sum \frac{(y_i \text{ obs} - y_i \text{ mod}(x_i, m, b))^2}{\sigma^2}$$

Utilizando el logaritmo natural de la probabilidad:

$$\ln(P(y|m)) = -0.5 \sum ((y_i \text{ obs} - y_i \text{ mod}(x_i, m))^2 / \sigma^2)$$

Los valores del algoritmo se van a repetir en ocasiones o modificarse muy poco durante pequeños intervalos de tiempo, pero de vez en cuando se producen saltos bruscos que permiten que el método recorra toda la función de probabilidad y no se quede estancada.

El nombre de Monte Carlo es en honor al casino de Monte Carlo en Mónaco ya que se juega a juegos de azar.

## CÓDIGO

Dentro del repositorio de Tiro al blanco, se encuentra la carpeta Proyecto, donde se almacena toda la información, archivos, funciones y programa para calcular las cadenas MCMC con el método de Metrópolis, lo cual se explica a continuación:

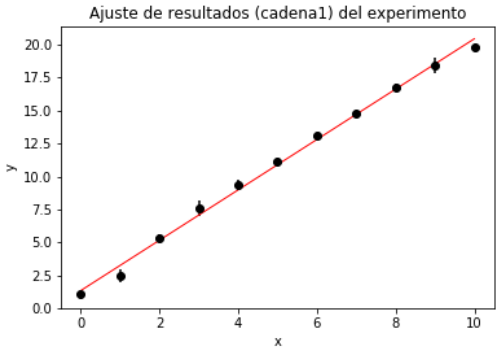
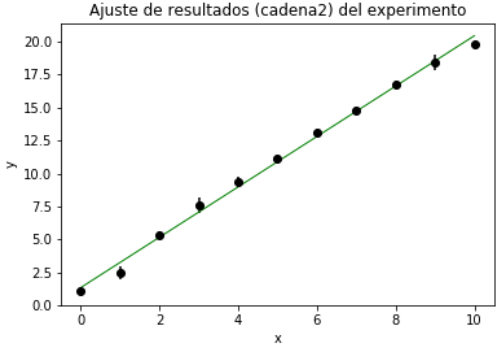
- a) **Función likelihood:** Comienza con la declaración de variables tanto con elementos de entrada y de salida. De entrada: pendiente, ordenada al origen, “N” y barras de error. Se inicializa un For, que correrá “N” veces. Se calculará y nueva[tipo arreglo] con los valores iniciales de pendiente y ordenada al origen junto con el arreglo de x[]. Se calcula el logaritmo natural de “likelihood”, es decir, el ln de la probabilidad de que esa pendiente y ordenada al origen sean adecuados a partir de números al azar. Al final obtenemos el valor del ln de likelihood.
- b) **Funcion num\_gauss:** Empezamos declarando las variables. Los elementos de entrada son la pendiente y la ordenada al origen, con sus respectivos sigma. Se obtienen dos números al azar en el rango de 0 a 1, y a partir de la suma de los cuadrados de esos números se obtiene “s”. Se comienza ciclo while con la condición de que “s” se encuentre dentro del mismo parámetro, al cumplirse, se calcula una nueva pendiente y una nueva ordenada al origen, de no cumplirse, se vuelven a elegir números al azar y vuelve el ciclo.
- c) **Programa final:** Se inicia en función main, declarando variables de tipo arreglo para x, y, y barras de error. Se abre el archivo de texto de donde sacaran los valores de el

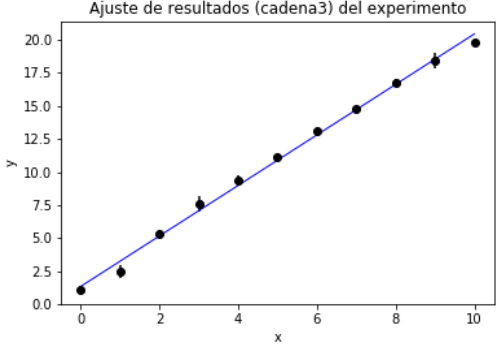
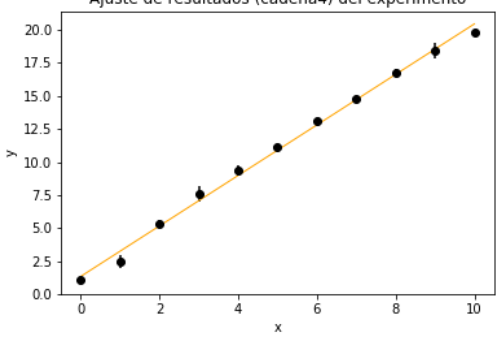
experimento. El cual contiene 11 elementos de x, y y barras de error que serán leídos por un ciclo for. Contando con esos elementos se llama a la función calculo\_monte, la cual tiene esos elementos de entrada.

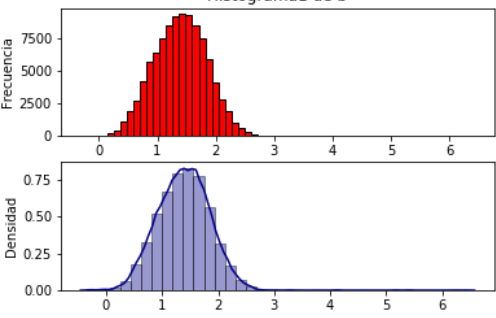
- d) **funcion calculo\_monte:** Se abre archivo de lectura con valores de b y m originales, y el parámetro de máxima y mínima, así como sigmas y las veces que el programa va a correr para encontrar el ajuste necesario. Se le otorgan valores obtenidos de b y m a los apuntadores de m0 y b0. Comienza un ciclo for que correrá las veces indicadas, se utiliza la función likelihood, para obtener el ln., después, se obtienen dos números al azar con la función num\_gauss para obtener m y b new. Se revisa que éstos números estén dentro del parámetro de mínimo y máximo, de cumplirse, se calculan los ln de likelihood para calcular su probabilidad, si su probabilidad es más positiva que la original, se guardan los valores, y si no se cumple, comienza el ciclo de nuevo, al final se imprimen los valores de likelihood con m0 y b0.

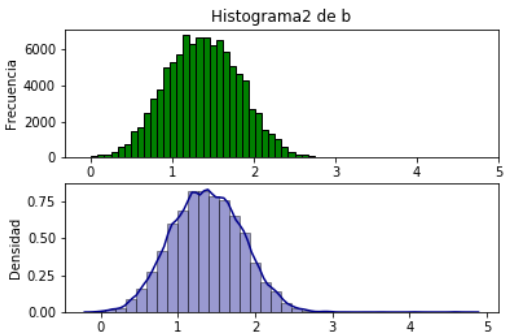
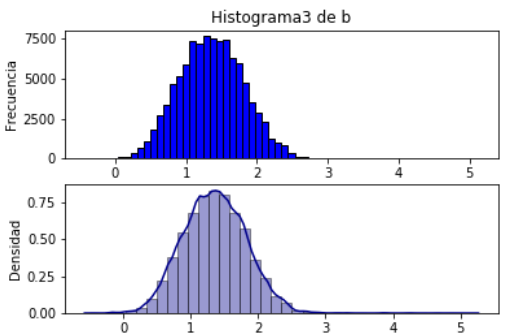
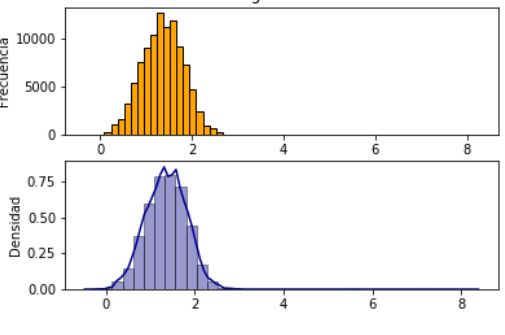
## RESULTADOS

A continuación se muestran los resultados obtenidos :

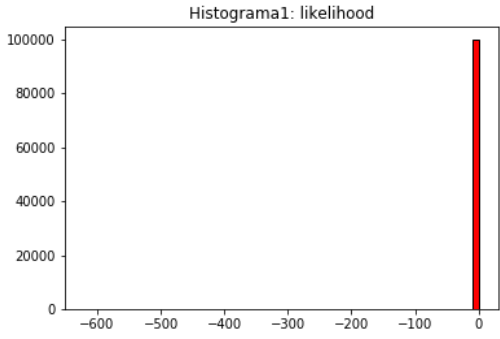
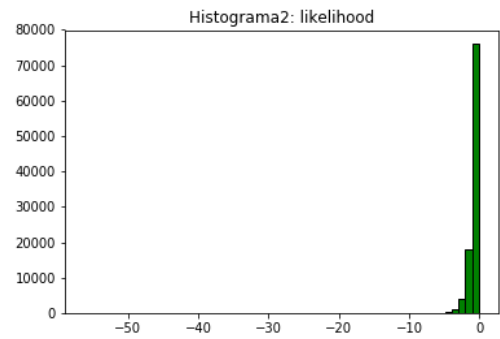
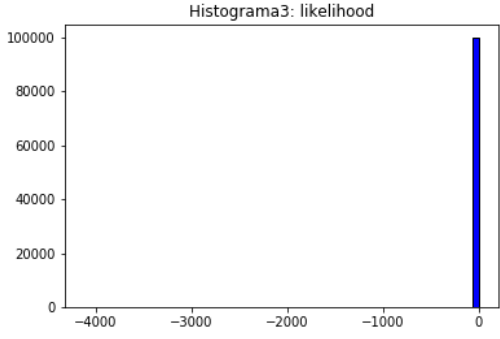
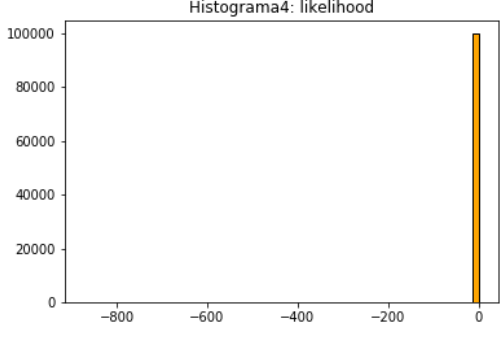
AJUSTE DE RESULTADOS	COMENTARIOS
	<p>Se muestra el ajuste de resultados de la cadena 1 del experimento, de x y y. Se observa que la recta de color rojo es muy próxima a los puntos del experimento, lo cual indica que el ajuste es correcto.</p>
	<p>Se muestra el ajuste de resultados de la cadena 2 del experimento, de x y y. Se observa que la recta de color verde es muy próxima a los puntos del experimento, lo cual indica que el ajuste es correcto.</p>

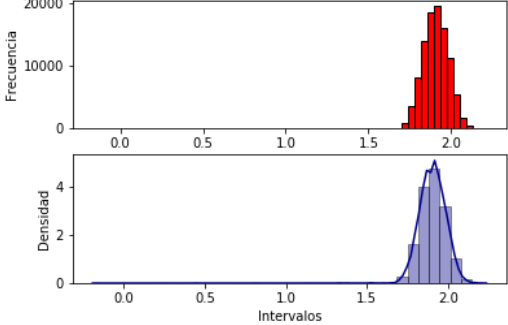
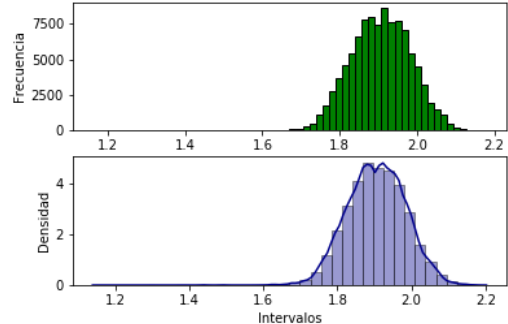
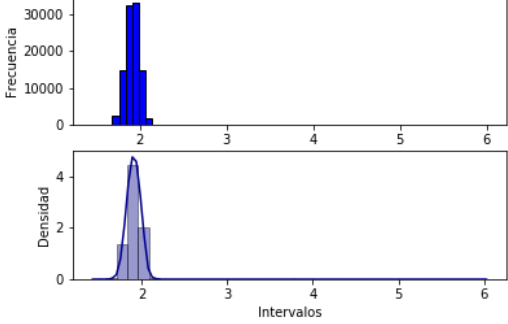
	<p>Se muestra el ajuste de resultados de la cadena 3 del experimento, de x y y. Se observa que la recta de color azul es muy próxima a los puntos del experimento, lo cual indica que el ajuste es correcto.</p>
	<p>Se muestra el ajuste de resultados de la cadena 4 del experimento, de x y y. Se observa que la recta de color amarillo es muy próxima a los puntos del experimento, lo cual indica que el ajuste es correcto.</p>

HISTOGRAMA DE B	COMENTARIOS
	<p>Se muestra el histograma de los valores de b, es decir la ordenada al origen, que tiene un parámetro entre 0 y 3, y que entre el 1 y 2 son los valores más altos.  media: 1.3866  desviación estándar: 0.46788  mediana: 1.3936</p>

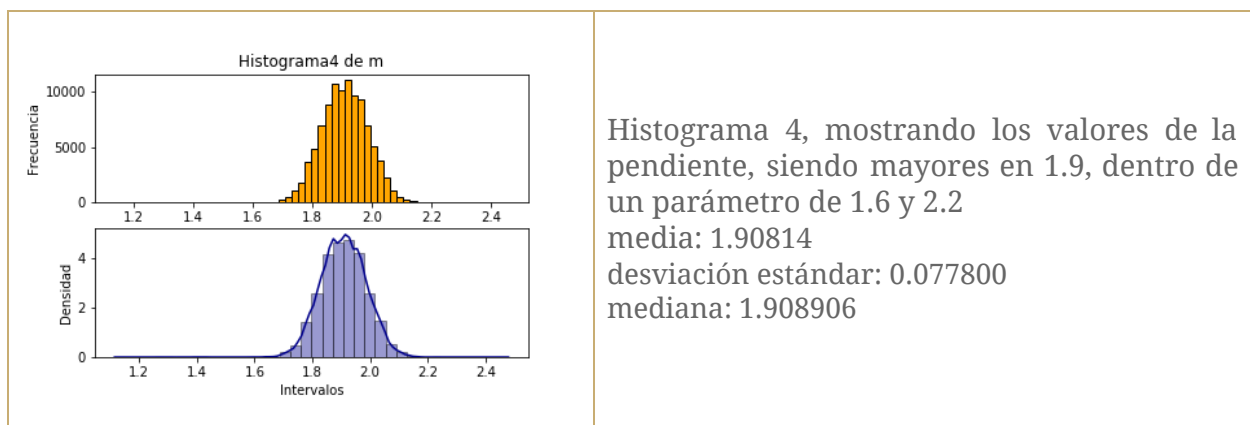
	<p>Se muestra el histograma de los valores de b, la ordenada al origen, que tiene un parámetro entre 0 y 3, y que entre el 1 y 2 son los valores más altos, y tiene un poco de error al tener unas barras un poco más altas de la curva. Tiene bastante amplitud. media: 1.386655 desviación estándar: 0.56788 mediana: 1.393629</p>
	<p>Se muestra el histograma de los valores de b, es decir la ordenada al origen, que tiene un parámetro entre 0 y hay uno pequeño en el 4, y que entre el 1 y 2 son los valores más altos. Lo cual indica que algún número cerca de 4 se arrojó un número incorrecto. media: 1.35555 desviación estándar: 0.462133 mediana: 1.350188</p>
	<p>Se muestra el histograma de los valores de b, es decir la ordenada al origen, que tiene un parámetro entre 0 y 3, y que entre el 1 y 2 son los valores más altos. Tiene menor amplitud. media: 1.358212 desviación estándar: 0.46255 mediana: 1.3580</p>

HISTOGRAMA DE L	COMENTARIOS
-----------------	-------------

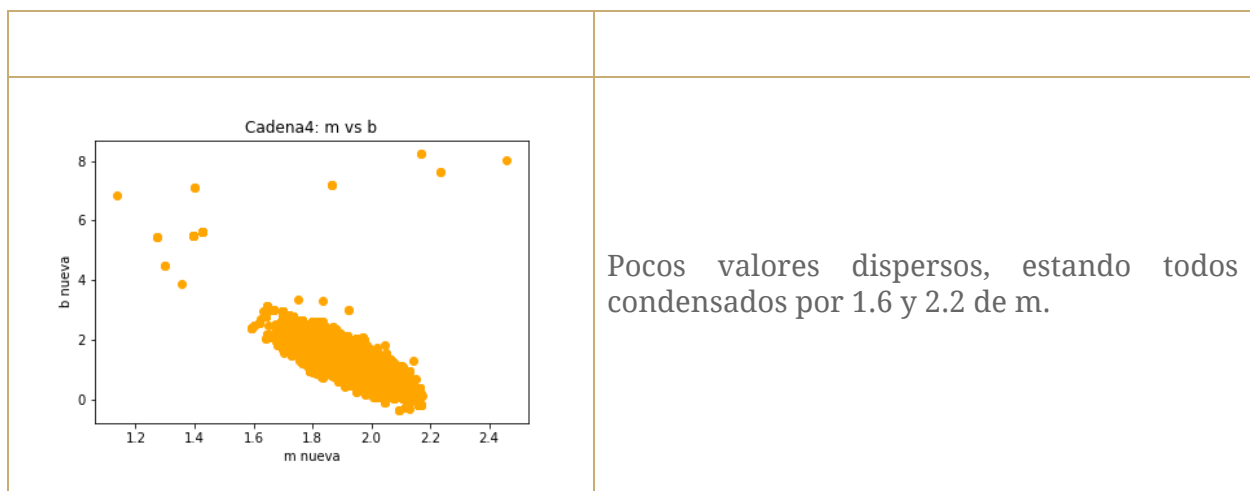
	<p>Se muestra un histograma del logaritmo de likelihood, mostrando un solo valor negativo muy pequeño.</p> <p>media: -0.8689 desviación estándar: 3.0957 mediana: -0.6427</p>
	<p>Se muestra el segundo histograma, mostrando el likelihood como valores negativos pequeños que están entre -5 y 0.</p> <p>media: -0.8689 desviación estándar: 3.0957 mediana: -.06427</p>
	<p>Se muestra el tercer histograma de likelihood, siendo un solo valor muy pequeño negativo.</p> <p>media: -1.3308 desviación estándar: 37.4008 mediana: -0.6420</p>
	<p>El cuarto histograma del logaritmo de likelihood mostrando un solo valor pequeño negativo.</p> <p>media: -0.8653 desviación estándar: 4.11247 mediana: -.063766</p>

HISTOGRAMA DE M	COMENTARIOS
<p data-bbox="207 262 711 289">Histograma1 de m</p> 	<p data-bbox="786 323 1421 537">El histograma de m, la pendiente, que ronda entre 1.5 y 2.5, siendo mayor cerca a 1.9 o 2. Con poca amplitud en la curva. media: 1.903434 desviación estándar: 0.07999 mediana: 1.904606</p>
<p data-bbox="207 674 711 701">Histograma2 de m</p> 	<p data-bbox="786 722 1421 936">El segundo histograma con gran amplitud, que ronda desde 1.6 hasta 2.2, teniendo mayor en 1.9. media: 1.9034 desviación estándar: 0.07999 mediana: 1.904606</p>
<p data-bbox="207 1148 711 1176">Histograma3 de m</p> 	<p data-bbox="786 1104 1421 1281">El tercer histograma, con poca amplitud, siendo mayor alrededor de 1.9. media: 1.9093 desviación estándar: 0.08878 mediana: 1.91012</p>





CADENA M VS B	COMENTARIOS
	<p>Se comparan los valores de la pendiente vs la ordenada al origen, siendo al inicio un poco dispersos pero juntándose alrededor del 1.5 y 2.2.</p>
	<p>En la segunda cadena, se empiezan a arrojar valores lejanos, pero después se condensan por 1.6 y 2.2.</p>
	<p>Tercera cadena, entre valores más cercanos a 0, la condensación de los puntos es mayor, alejándose los puntos mayores de 2.5 de m.</p>



## DISCUSIÓN

Entre mayor sea el número que se repiten las iteraciones, mayor será la probabilidad de encontrar un número con  $\ln$  más positivo, que más se asemeje al original. Las cuatro cadenas quedaron bien y se ajustaron al experimento. Obtuvimos en las gráficas una distribución gaussiana en los histogramas, debido a que de ella obtuvimos los datos al azar. En las cuatro cadenas teniendo diferentes  $m$  y  $b$  iniciales, llegamos a la misma pendiente y a la misma ordenada al origen iniciales, comprobando que el programa realizado funciona correctamente, dando cualquier valores iniciales. Esto se puede comprobar en los GIF contenidos en el archivo. (monte.gif y ajuste.gif)

## CONCLUSIÓN

Este método sirve para determinar el mejor ajuste de los resultados de un experimento para poder encontrar  $m$  y  $b$  que mejor satisfagan una serie de datos de un experimento y así crear una recta que mejor se ajuste y sea la más probable. Una ventaja es que únicamente depende de  $m$  anterior y  $b$  anterior y se basa en la probabilidad. Estos métodos son muy buenos cuando necesitas un resultado aproximado, y sabes que obtener un resultado exacto no va a cambiar mucho el resultado final. En términos generales, las técnicas aquí utilizadas son más eficientes y darán resultados más precisos en la medida en que la aproximación normal asintótica a la distribución final sea más adecuada. Es por esta razón que en la mayoría de los casos resulta conveniente trabajar con estos métodos.

## REFERENCIAS

1. The Monte Carlo method or Metropolis algorithm. 1946
2. The simplex method of linear programming. 1947

3. Mackay, D.J.C.: Introduction to Monte Carlo Methods. Springer (1986)