

ESTUDIO DEL MÉTODO MONTE CARLO EN SIMULACIONES PARA LA ESTIMACIÓN DEL VALOR DE PI

Juan Pablo Crespo Vargas, Carlos Cruz Carpio.

jp.crespo.vargas@gmail.com, ccruz.carloscruz@gmail.com

RESUMEN

En este artículo se presenta los resultados de tres metodologías diferentes en las que se aplicó una simulación Monte Carlo para estimar el valor de Pi: el método de comparación de áreas, el método propuesto por Buffon y la extensión de Laplace. Los tres casos se desarrollaron en un lenguaje de alto nivel, Python y la librería Numpy que le otorgan un performance optimizado. Se estudió con detalle el resultado no determinista de las simulaciones y se demostró que cumplen con los teoremas fundamentales de la probabilidad.

PALABRAS CLAVE

Simulación, Monte Carlo, Pi, Buffon, Buffon-Laplace, Python, Numpy.

1. INTRODUCCIÓN

El método de Monte Carlo tiene un génesis moderno en el trabajo pionero de Stan Ulam y John Von Neumann. Luego de la segunda Guerra Mundial aplicaron distintos métodos de Monte Carlo en simulaciones para el desarrollo de armas termonucleares. Desde entonces y por más de 50 años que se aplicaron estos desarrollos en la investigación y perfeccionamiento de distintos métodos que modelan el transporte de neutrones y radiación gamma con bastante éxito experimental. (Dr. Andreas Kling, 2000)

Hoy resulta una alegre ironía que ningún producto que haya aplicado la metodología Monte Carlo en su desarrollo, se haya empleado en conflicto alguno. Más aún los científicos han explotado el uso de simulaciones Monte Carlo para obtener un beneficio público positivo aplicándola en salud. Por ejemplo, los planeamientos de dosis en radioterapia dependen actualmente en algún grado de cálculos obtenidos mediante simulaciones que emplean Monte Carlo.

El método de Monte Carlo es un método de resolución numérica donde se modelan las relaciones e interacciones de distintos objetos y su entorno, mediante la generación aleatoria de estas interacciones. Mientras mayor sea la repetición de pruebas se obtiene un resultado que va convergiendo a un valor con mayor precisión. Es por el recurso de la aleatoriedad que obtiene el nombre Monte Carlo, pues se inspira en la región del Principado de Mónaco donde se encuentran el casino Monte Carlo.

Un método Monte Carlo se puede definir de la siguiente forma: <<Los métodos Monte Carlo son aquéllos en los que las propiedades de las distribuciones de las variables aleatorias son investigadas mediante la simulación de números aleatorios. Estos métodos, dejando a un lado el origen de los datos, son similares a los métodos estadísticos habituales en los cuales las muestras aleatorias se utilizan para realizar inferencias acerca de las poblaciones origen. Generalmente, en su aplicación estadística se utiliza un modelo para simular un fenómeno que contiene algún componente aleatorio. En los métodos Monte Carlo, por otro lado, el objeto de la investigación es un modelo en sí mismo, y se utilizan sucesos

aleatorios o pseudoaleatorios para estudiarlo.>> (Gentle, 1986)

El método cobra una especial relevancia las últimas décadas debido a que se produjeron sustanciales y significativos avances respecto a la potencia de los procesadores y las distintas arquitecturas informáticas. Es ampliamente usado en problemas donde obtener un resultado analítico no es posible, o en problemas que contienen demasiada complejidad (como es el caso de la ecuación de transporte de Boltzmann para partículas sin carga). (Dr. Andreas Kling, 2000)

2. Marco Teórico.

El estudio matemático formal del azar se remonta hace bastantes siglos. En 1654 motivados por dos problemas propuestos por Antoine Gombaud (le Chevalier de Méré), basados en las observaciones de los juegos de azar de la época, es que se reúnen a resolver el desafío matemático personalidades como Pascal, Cardano y Fermat entre otros dando inicio a la teoría clásica de la probabilidad. Es en este periodo que distintos matemáticos advierten la relación de la teoría combinatoria y la incipiente teoría de la probabilidad.

Un matemático que realiza un interesante aporte en este aspecto es Leibniz, el cual luego de realizar la disertación titulada *Dissertatio de Arte combinatoria*, encuentra particular interés por ‘la certidumbre’.

La probabilidad para Leibniz es un criterio objetivo de verdad. Es una lógica de lo contingente (que puede suceder o no) que se contrapone a la lógica de lo necesario. Permitiendo a la probabilidad alcanzar la verdad. El interés de Leibniz por los juegos iba más allá de la teoría de la probabilidad, sus aplicaciones podrían aplicarse al Arte de Inventar y subsecuentemente a la construcción de las características universales, temas que le parecían de mayor interés que a otros autores. (CHARLES, 1993)

2.1. Experimentación y la teoría de probabilidad.

En 1777, el naturalista francés, el conde de Buffon (1707-1788) propuso el primer experimento que utilizaba un método de Monte Carlo, pues dependía de un hecho completamente

aleatorio: la caída de una aguja luego de lanzarla. Las agujas son lanzadas aleatoriamente en un piso con un patrón de rayas separadas una cierta distancia. Buffon considero que los centros están uniformemente distribuidos en un piso infinito. Las agujas no ruedan a las aberturas como lo harían en la vida real, ni estas interactúan entre sí. Además, el ángulo respecto a la horizontal es considerado distribuido uniforme entre 0 y $\pi/2$.

El resultado al que se llegó relaciona la probabilidad de cruzar una de las rayas con la distancia de separación, la longitud de la aguja y el valor de pi. Esta se expresa en la ecuación 1.

$$P = \frac{2l}{d\pi}$$

Se conoce como la extensión de Laplace al problema de Buffon cuando se considera tanto líneas verticales como horizontales. Se llama Buffon-Laplace pues, aunque Buffon resolvió este problema contenía un error que más tarde, 1812, fue corregido por Laplace.

$$P = \frac{2l(a+b) - l^2}{ab\pi}$$

De esta propuesta, se pretende obtener el valor de pi. La probabilidad se obtiene empíricamente al realizar el experimento un número grande de veces y contar cada caso.

Este problema histórico fue estudiado en ese entonces como una curiosidad poco práctica, debido a que obtener un resultado preciso requería un número grande de repeticiones.

Durante el siglo XX con el advenimiento de las nuevas tecnologías es que el método de Monte Carlo vuelve a cobrar relevancia, pues permite un número grande de repeticiones sin complicación. Sin embargo, el manejo teórico de la matemática involucrada es bastante conocido, se realiza una breve revisión.

2.2. Teorema de los Números Grandes

Sea X_1, \dots, X_n una sucesión de variables aleatorias, diremos que la sucesión de $\{X_n\}$ converge en probabilidad a X (que es otra variable aleatoria) si solo si para todo $\forall \epsilon > 0$:

$$\lim_{n \rightarrow \infty} (P[(X_n - X) > \epsilon]) = 0$$

Siendo conocida esta expresión como criterio de convergencia débil.

Tomamos la variable aleatoria de interés, el promedio: $\bar{x} = \frac{1}{n} \sum X_i$

Aplicando el operador lineal Esperanza:

$$E(\bar{X}_n) = E\left(\frac{1}{n} \sum X_i\right)$$

$$E(\bar{X}_n) = \frac{1}{n} \sum E(X_i)$$

Si consideramos que la sucesión de variables X_i son independientes e idénticamente distribuidas, la Esperanza para cada variable es la media μ .

$$E(\bar{X}_n) = \frac{1}{n} * n\mu$$

$$E(\bar{X}_n) = \mu$$

Esta relación nos muestra que el valor esperado del promedio de una muestra tiende al valor medio.

De igual manera con el operador varianza tenemos:

$$V(\bar{X}_n) = V\left(\frac{1}{n} \sum X_i\right)$$

Con el supuesto de las variables (X_i), son independientes e idénticamente distribuidas, la varianza de la suma, es la suma de las varianzas:

$$V(\bar{X}_n) = \frac{1}{n^2} \sum_{i=1}^n V(X_i)$$

Al ser idéntica para todos tenemos:

$$V(X_i) = \sigma^2$$

$$V(\bar{X}_n) = \frac{1}{n^2} * n\sigma^2 = \frac{\sigma^2}{n}$$

Con los resultados de la esperanza y varianza del promedio, se puede enunciar el siguiente teorema:

Sea X_n una sucesión de una variable aleatoria independiente e idénticamente distribuida tal que $E(X_i^2) < \infty \forall i, y$:

$$E(\bar{X}_n) = \mu$$

$$V(\bar{X}_n) = \frac{\sigma^2}{n}$$

Entonces el promedio de la secuencia X_n converge en probabilidad a μ . Es decir que para una población que tiene una media μ , si se toma una muestra y se calcula el promedio muestral, este tiende al valor de μ mientras n tienda a ∞ .

Esto se demuestra al tomar un valor $\epsilon > 0$, entonces, usando la relación de acotación de Chemichev:

$$0 \leq P[(\bar{X}_n - \mu) > \epsilon] \leq \frac{E[(\bar{X}_n - \mu)^2]}{\epsilon^2}$$

$E[(\bar{X}_n - \mu)^2]$ es la varianza del promedio, entonces tenemos la relación:

$$0 \leq P[(\bar{X}_n - \mu) > \epsilon] \leq \frac{\sigma^2}{n \epsilon^2}$$

Mostrando que cuando n tiende a un número grande, la probabilidad se anula y por consecuencia el promedio converge en probabilidad a μ .

2.3. Teorema del Límite Central

Sea X_1, \dots, X_n una sucesión de variables aleatorias, independientes e idénticamente distribuidas tales que $\forall i$ tengan el segundo momento finito $E(X_i^2) < \infty$ con una media μ y una varianza σ^2 distinta de cero. Entonces, si n es suficientemente grande, la variable aleatoria promedio:

$$\bar{(X)} = \frac{1}{n} \sum X_i$$

Tiene aproximadamente una distribución normal con:

$$\mu_{\bar{(X)}} = \mu$$

$$V_{\bar{(X)}} = \frac{\sigma^2}{n}$$

El teorema central de la probabilidad tiene una importancia categórica al ser enunciada para una sucesión de variables con cualquier distribución.

3. Modelo de simulación.

3.1. Entorno de desarrollo.

Como quedo manifiesto en el anterior punto el método Monte Carlo tiene una precisión proporcional a $\frac{1}{\sqrt{N}}$. En comparación con otros métodos numéricos determinísticos (como por ejemplo el método de trapecios o Simpson para encontrar la integral de una función definida) que tienen un error de aproximación proporcional a $\frac{1}{N^2}$, los métodos que aplican Monte Carlo requieren una cantidad considerablemente mayor de datos a procesar. Sumado este hecho a la complejidad que puede involucrar el modelamiento de las interacciones aleatorias, es que generalmente se prefiere usar lenguajes de programación de bajo nivel que permitan minimizar el tiempo de cómputo total.

Por ejemplo, es común encontrar desarrollos en c, c++ y Fortran entre otros. Existiendo hoy librerías que facilitan la generación de números pseudoaleatorios y el manejo matemático. Aún con esta ventaja el código necesario para una aplicación final suele ser bastante complicado y extenso. Estos dos criterios identificados se consideraron para la selección del entorno de desarrollo:

1. El tiempo de procesamiento.
2. La legibilidad del código.

Siendo Python un lenguaje de alto nivel que cumple con el segundo criterio, al incluir la librería Numpy (librería de procesamiento *numérico* para Python) obtenemos una velocidad de procesamiento comparable a C. Adicionalmente se utilizó una librería para la representación de los datos, Matplotlib.

Se detalla todo el código y las dependencias en el repositorio del proyecto:

<https://github.com/CobraPython/montecarlopi>

3.2. Generación de números pseudoaleatorios.

Existen distintos métodos para la generación de números aleatorios, sin embargo, la generación de estos en ordenador parte necesariamente desde una semilla (seed) que es un valor concedido por el usuario. Con esta semilla

se genera una única serie de números aleatorios, pudiendo ser replicados a partir de esta. Por esta razón es que se denominan números pseudo aleatorios.

La librería Numpy utiliza el algoritmo Mersenne Twister (MT19937) (Tanguy, s.f.) para la generación de números pseudoaleatorios. Este método particular tiene la cualidad de tener una periodicidad bastante grande en la generación de números: $2^{19937} - 1$. (Makoto Matsumoto, 1998)

3.3. Métodos de estimación de Pi

3.3.1. Método Simple para la estimación de Pi.

Se propone estimar el número de Pi con el siguiente modelo:

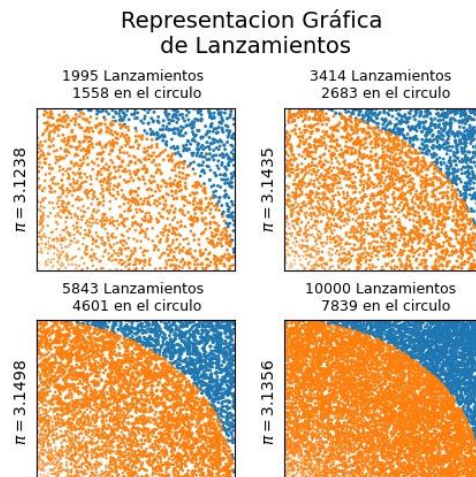
Consideramos un cuadrado de lado L, con una circunferencia en su interior de radio L.

La relación de áreas se da de la siguiente manera:

$$\frac{A_{\text{circunferencia}}}{A_{\text{cuadrado}}} = \frac{\pi L^2}{L^2}$$

Una forma de calcular esta relación de áreas es lanzar al azar puntos dentro del cuadrado. Estos puntos pueden quedar también dentro de la circunferencia, la relación de áreas quedara expresada por aquellos puntos que estén dentro del círculo sobre el total.

En la siguiente gráfica se muestra un ejemplo del experimento propuesto, mostrando un solo cuadrante ya que las áreas son simétricas en cada eje.

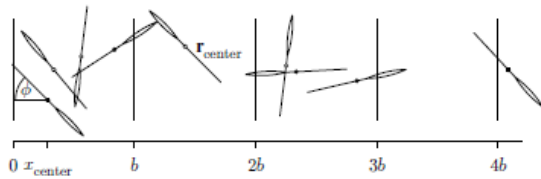


Queda bastante ejemplificado que mientras mayor sea el número de puntos, las áreas quedan mejor definidas.

Para obtener esta aproximación se necesitan generar los puntos aleatoriamente con una distribución uniforme, es decir, con igual probabilidad de caer dentro y fuera del área del cuarto de circunferencia.

3.3.2. Método de Buffon para la estimación de Pi.

En el caso de la estimación de Pi usando la propuesta experimental de Buffon y su extensión de Laplace, tenemos más variables aleatorias a considerar. Puesto que cada aguja, con un centro en (x_i, y_i) , tiene relacionada otra variable aleatoria que corresponde al ángulo de inclinación θ de la aguja.

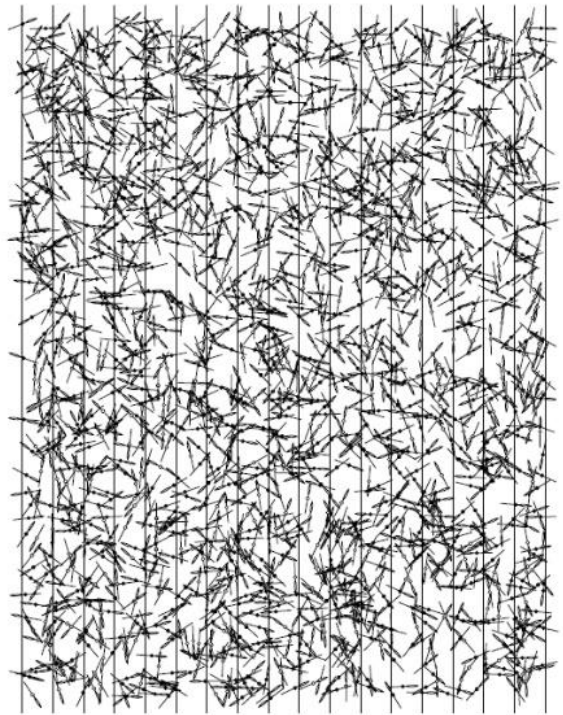


Si bien en este caso se recurre a una función trigonométrica para evaluar la inclinación y considerar a las agujas que cruzan la línea, el ordenador usa recursivamente el valor de Pi para calcular la función coseno, siendo este un error “histórico” ya que recurre al valor de pi para calcular al mismo. Por esta razón se usó una corrección geométrica que evita el uso de funciones trigonométricas.

Pasando de requerir generar aleatoriamente un ángulo θ , a generar aleatoriamente desplazamientos δx y δy . En la siguiente figura se muestra una representación gráfica de cómo se realiza la simulación con un número grande de lanzamientos aleatorios con distribución uniforme. (Krauth, 2006)

Para esta propuesta del lanzamiento de agujas para la estimación de Pi, se comparan dos casos:
-Problema Buffon, cuando las agujas cruzan líneas en un solo eje.

-Problema Buffon-Laplace, cuando las agujas cruzan líneas en sentido vertical y horizontal.

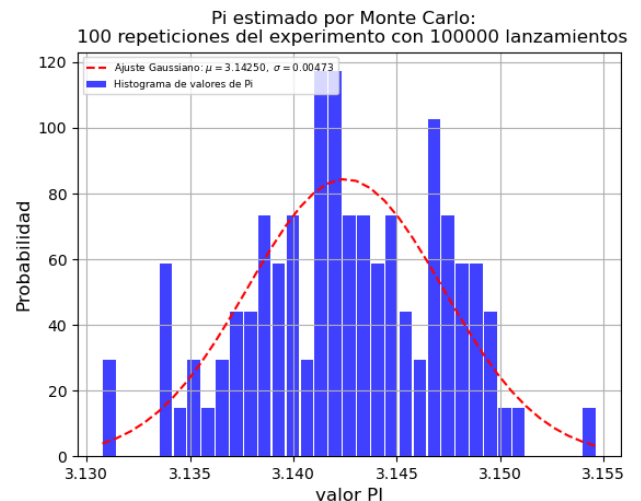


4. Análisis de resultados.

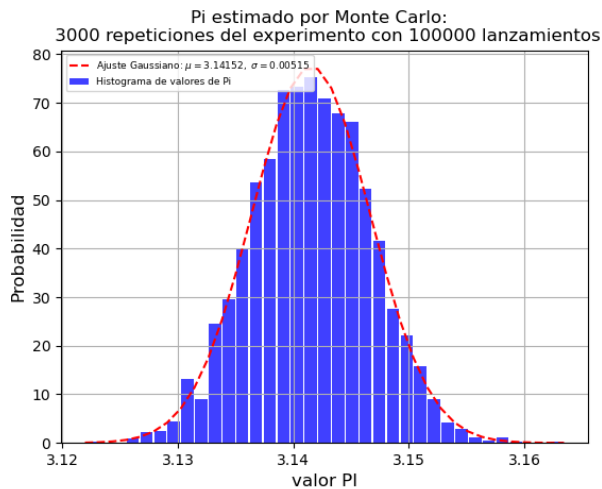
4.1. Método Simple para la estimación de Pi

4.1.1. Comprobación del teorema del límite central

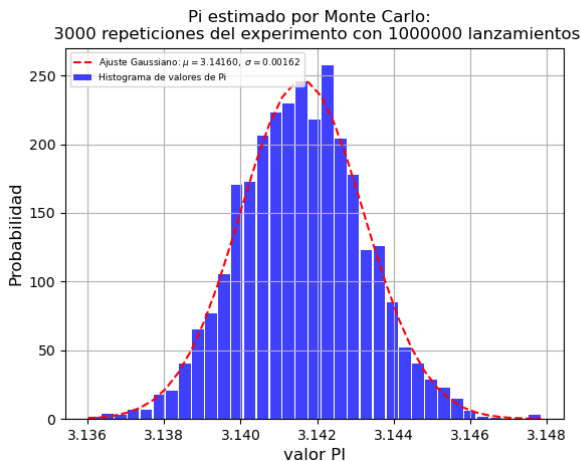
Se consideró el lanzamiento de 10^5 puntos, y se repitió 10^2 veces el experimento, el resultado obtenido tiene el siguiente comportamiento al representar su histograma.



Al realizarse 3×10^3 veces el experimento, se nota más claramente que el histograma del resultado obtenido tiene un comportamiento gaussiano.



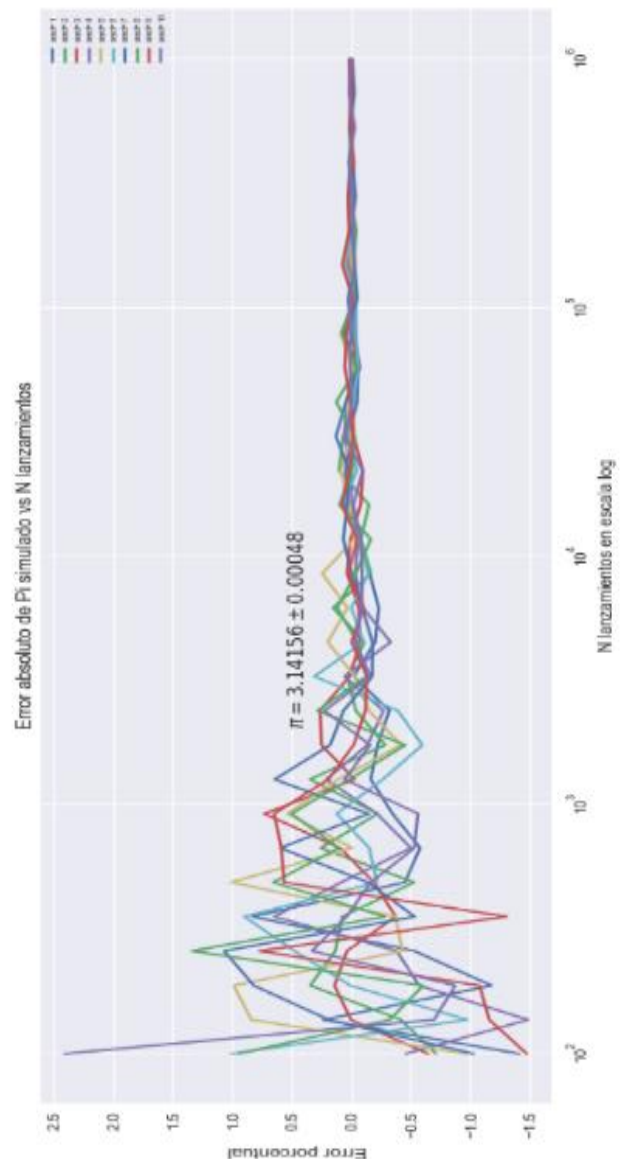
Se compara con realizar 3×10^3 veces el experimento lanzando 10^6 puntos, el comportamiento del histograma es también Gaussiano.



El valor de σ es menor cuando consideramos 10^6 ($\sigma = 0.00162$) en comparación con 10^5 ($\sigma = 0.00473$).

4.1.2. Comprobación del teorema de números grandes.

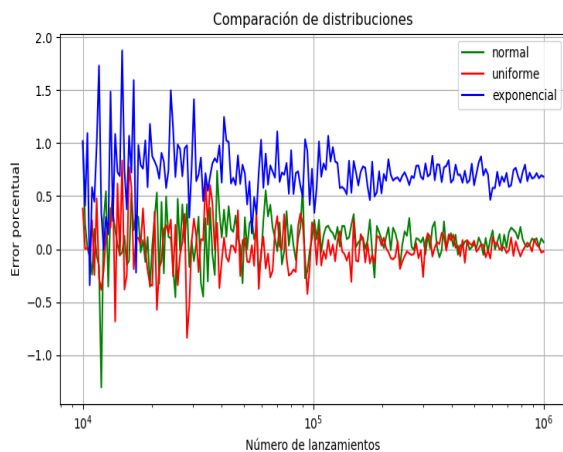
Se ha estimado el valor de Pi para distintas cantidades de puntos generados. En la siguiente gráfica se muestra la congruencia de Pi con el valor más preciso que tiene la librería. Se muestra que mientras mayor es el N tomado para estimar el valor de Pi, su error absoluto converge a cero.



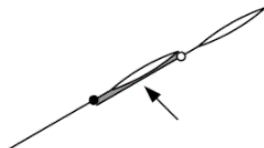
En el gráfico siguiente se muestra que este comportamiento es invariante al variar las semillas en la generación de los números pseudoaleatorios. Finalmente, en la siguiente gráfica se muestra una comparativa de la discrepancia al estimar el valor de π si consideramos distintas distribuciones de probabilidad en la generación aleatoria de puntos.

Se comparan 3 distintas distribuciones:

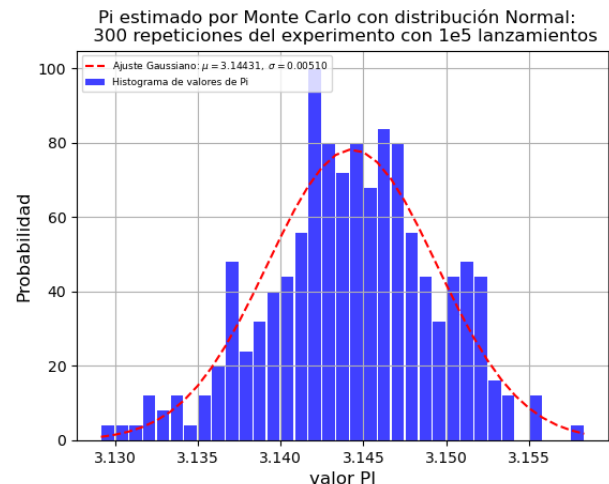
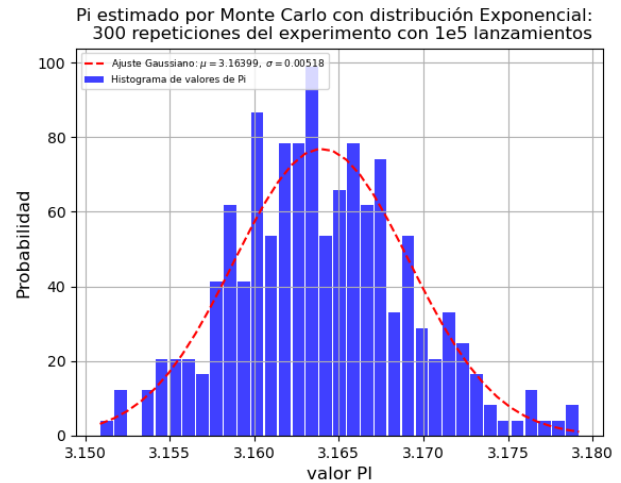
- Normal Gaussiano.
- Uniforme.
- Exponencial.



Es notable que para las tres distribuciones el valor de la discrepancia del valor estimado de π converge a un valor mientras mayor número de lanzamientos se utilice en la prueba. Al aplicar una distribución normal con $\mu = 0.27$ y $\sigma = 0.5$ el valor estimado de π es cercano que cuando se aplica una distribución uniforme. Al aplicar una distribución exponencial con un parámetro $\lambda = 0.42$, el valor estimado de π converge con un error de más del 0.5%. Esto se debe a que los lanzamientos de las agujas tienen mayor probabilidad de caer en cierta posición y ángulo, lo que puede equivaler de manera experimental a por ejemplo que las agujas tengan otra conformación física, como en la siguiente imagen.



Si realizamos los histogramas al estimar el valor de π un número grande de repeticiones, aplicando las distintas distribuciones, se demuestra que tienen un comportamiento Gaussiano en todos los casos. En la siguiente gráfica se muestran los histogramas de las distribuciones normal y exponencial.

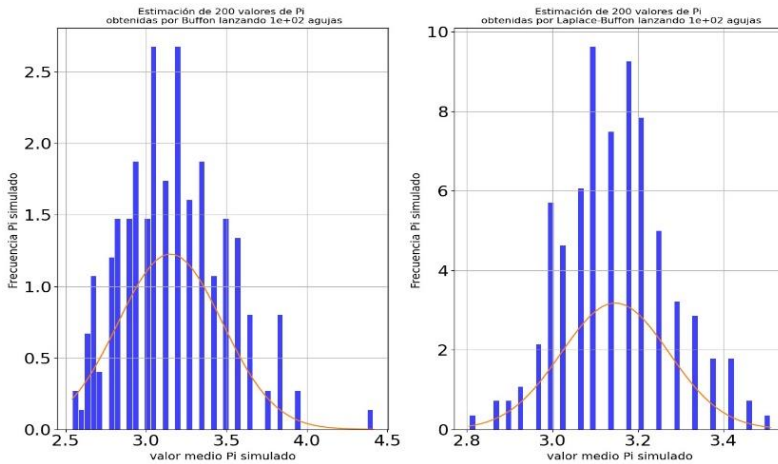


4.2. Método de Buffon para la estimación de π .

4.2.1 Comprobación del teorema del límite central.

Se consideró el lanzamiento aleatorio de 100 agujas, con una distribución uniforme para la posición y ángulo, repitiendo el experimento 200 veces para ambos casos; Buffon y la extensión de Laplace.

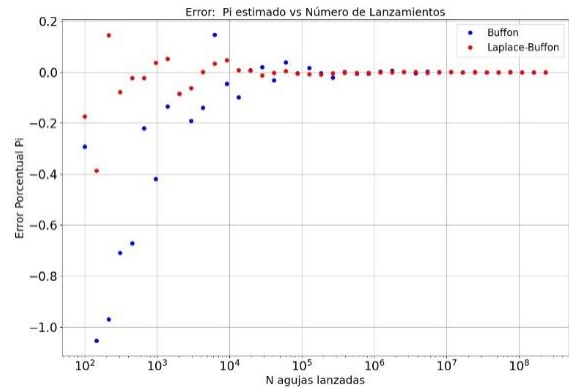
Mostrando que para ambos casos el histograma de las estimaciones de π para ambos métodos presenta un comportamiento Gaussiano, demostrando la validez del teorema del límite central.



4.2.2 Comprobación del teorema de los números grandes.

En la siguiente gráfica se resume el error porcentual al estimar el valor de π vs. El número de lanzamiento de agujas para ambos métodos. El error converge a cero a partir del lanzamiento de 10^6 agujas.

En la anterior gráfica queda evidente que con el método corregido de Laplace el valor estimado de π converge a su valor esperado con un menor número de lanzamiento de agujas, en un factor de 10 al compararlo con el método original de Buffon.



5. Conclusiones

En esta experiencia se mostraron tres distintos métodos que aplican Monte Carlo para estimar el valor de π : el método simple por comparación de áreas, el método propuesto por Buffon mediante el lanzamiento de agujas y el método de Buffon ampliado por Laplace.

Se demostró que los resultados obtenidos mediante una simulación Monte Carlo tienen un carácter no determinista que cumplen con los teoremas centrales de la teoría de la probabilidad.

Finalmente, se mostró que se puede lograr simulaciones con un alto costo computacional aplicando un lenguaje de alto nivel como Python y la librería Numpy, lo que permite tener un código simple y legible.

Bibliografía

CHARLES, M. S. (1993). Leibniz y los dos problemas de mere. *Universidad del País Vasco/EHU*, 24.

Dr. Andreas Kling, P. F. (2000). *Some Random Thoughts on Monte Carlo Electron and Photon Transport. En Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications*. Lisboa: Springer.

Gentle, J. E. (1986). *Random number generation and Monte Carlo methods*. Springer Science & Business Media.

Krauth, W. (2006). *Algorithms and Computations*.
London - Great Britain: Oxford University
Press.

Makoto Matsumoto, T. N. (1998). *Mersenne
Twister: A 623-dimensionally
equidistributed uniform pseudorandom
number generator*. ACM transactions on
Modeling and Computer Simulations.

Tanguy. (s.f.). *How Does Your Computer Generate
Random Numbers?* Obtenido de
[https://www.sicara.ai/blog/2019-01-28-
how-computer-generate-random-
numbers](https://www.sicara.ai/blog/2019-01-28-how-computer-generate-random-numbers)