

P5

Ismael Crespo

16 de marzo de 2022

1. Introducción.

Este trabajo presenta la obtención del valor de una integral definida de una función normalizada por medio del método de Monte-Carlo. Se pretende analizar como la cantidad de elementos que componen el área bajo la curva de la función afectan la precisión del valor obtenido al evaluado directamente por medio de la integración definida. La creación de este análisis se realizó creando rutinas en *R* 4.1.2.

2. Objetivos.

- 1.-Estudiar estadísticamente la convergencia de la precisión del estimado del integral con en método Monte Carlo, comparando con el valor producido por Wolfram Alpha, en términos del (1) error absoluto, (2) error cuadrado y (3) cantidad de decimales correctos, aumentando el tamaño de muestra.
- 2.-Realizar la estimación del valor de π de Kurt [3] y realizar el mismo análisis.

3. Programación en R.

Para realizar la estimación de la integral se definió una función $f(x)$ (ecuación 1) y esta se normalizo para que el área bajo la curva sea igual a 1 $g(x)$ (ecuación 2), posteriormente se construyó un generador de números aleatorios que genera valores dentro de esta función normalizada, para realizar este análisis de realizaron 20 réplicas variando el número de elementos a generar y a manera de estimar la integral definida de $g(x)$ (ecuación 3) entre 7 y 3 se realiza la sumatorio de los elementos en este rango, el código 1 realiza la rutina descrita.

$$f(x) = \frac{1}{\exp(x) + \exp(-x)} \quad (1)$$

$$\int_{-\infty}^{\infty} \frac{2}{\pi} f(x) dx = 1 \quad (2)$$

$$\int_3^7 \frac{2}{\pi} f(x) dx = 0,04883411112604931084064237 \quad (3)$$

Código 1: Normalización de la función y estimación del área de bajo de la curva entre 7 y 3.

```
1 f <- function(x) { return(1 / (exp(x) + exp(-x))) }  
2 g <- function(x) { return((2 / pi) * f(x)) }  
3 generador <- r(AbscontDistribution(d = g)) # creamos un generador  
4 entre=function(n){gen=generador(n);return((pi/2)*sum(gen>=3&gen<=7)/n)}
```

El análisis de la precisión de realizo encontrando las diferencias absolutas, las diferencias cuadradas y la exactitud de cada uno de los decimales del resultado (véase el código 2). [1].

Código 2: Métodos de obtención de la precisión de la estimación comparada con el valor correcto obtenido de *Wolfram Alpha* [2].

```
1  abso=abs(correcto-estimado)
2  absolutos=c(absolutos,abso)
3  abso_cuad=(correcto-estimado)**2
4  absolutos_cuad=c(absolutos_cuad,abso_cuad)
5  decimal=3
6  de=substring(as.character(estimado),1,decimal)
7  dc=substring(as.character(correcto),1,decimal)
8  while(de==dc){
9  de=substring(as.character(estimado),1,decimal)
10 dc=substring(as.character(correcto),1,decimal)
11 decimal=decimal+1
12 }
```

La segunda parte del trabajo realiza la estimación del valor de π despejado de la ecuación 4. El área del círculo se encuentra generando números aleatorios de manera uniforme, que al graficar se observan como en la figura 1a, posteriormente se encuentran los puntos cuya distancia euclidiana sea menor al radio definido previamente, por conveniencia los datos se generaron en el intervalo -1 a 1 tanto en x como en y , por lo tanto el radio del círculo es 1, el área graficada se observa en la figura 1b, la sumatoria de estos datos se considera el área del círculo de radio 1 y se sustituye en la ecuación 5 para estimar el valor de π (véase el código 3). Se realiza el mismo análisis de la precisión de la estimación en relación al valor de π como valor correcto.

$$A = \pi r^2 \quad (4)$$

$$\pi = r^2/A \quad (5)$$

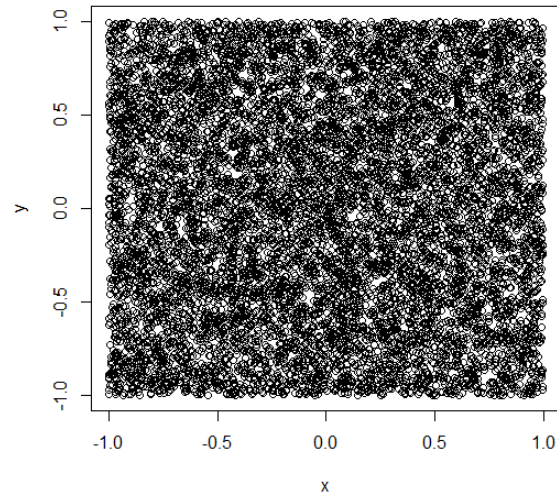
Código 3: Estimación del valor de π .

```
1  for (n in num){
2  for(r in 1:replicas)
3  x=runif(n,-1,1)
4  y=runif(n,-1,1)
5  d=numeric()
6  d=c(d,sqrt(x[i]**2 + y[i]**2))
7  dentro=d<=1
8  estimado=sum(dentro)*4/n
```

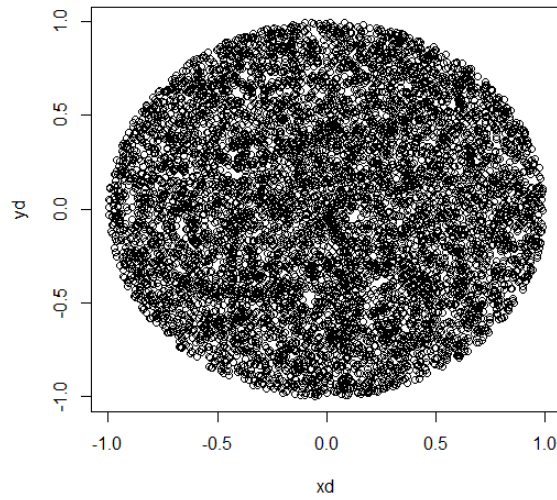
4. Resultados.

La figura 2 presenta las precisiones de cada estimación variando el valor de n , en la figura 2a se analiza las diferencias absolutas, basándose en estos resultados podríamos determinar que un valor de $n = 1000000$ nos dará valores con precisión aceptable considerando que ya no se observa un cambio importante para valores de n mayores, en la figura 2b se observa las diferencias cuadradas, utilizando este análisis como referencia nuestras deducciones cambian y utilizando un valor de $n = 10000$ tendríamos valores aceptables bajo este criterio, el análisis de la exactitud del valor, es decir cuantos decimales son exactamente iguales al valor correcto se observa en la figura 2c, este criterio nos obliga a utilizar valores altos de n para concluir que tenemos valores precisos.

El análisis de la precisión en la estimación de π se presenta en la figura 3. Las conclusiones referente a que valor de n da los valores mas precisos son similares a los obtenidos en la primera parte. La figura 3a presenta la precisión en función de la diferencia absoluta, 3b presenta la precisión en función de la diferencia cuadrada y 3c presenta la precisión en función de la exactitud de los decimales. Al ser una rutina mas compleja se tuvo como limitante los valores máximos permisibles de n para simular.

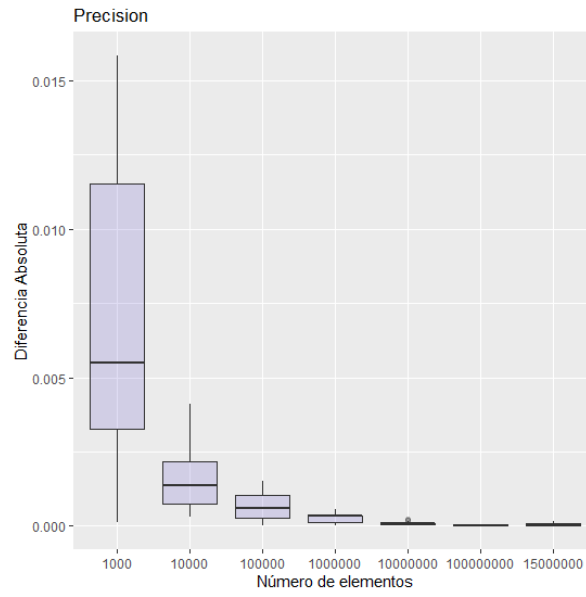


(a) Números uniformemente posicionados en x y y .

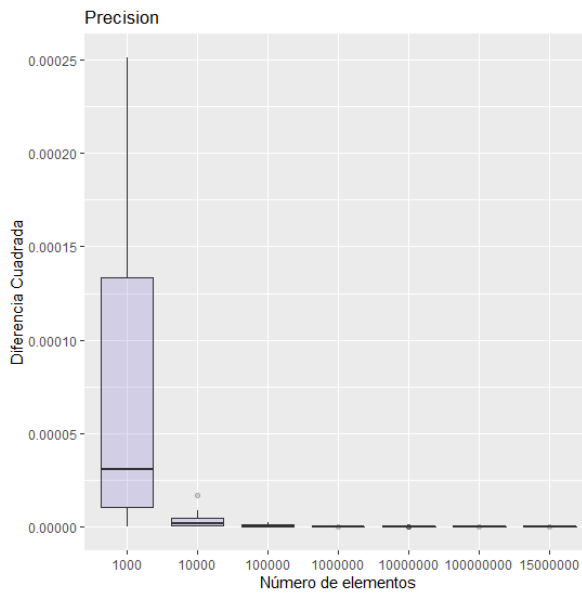


(b) x y y dentro de la circunferencia obtenidos por medio de la distancia euclidiana menor o igual a 1.

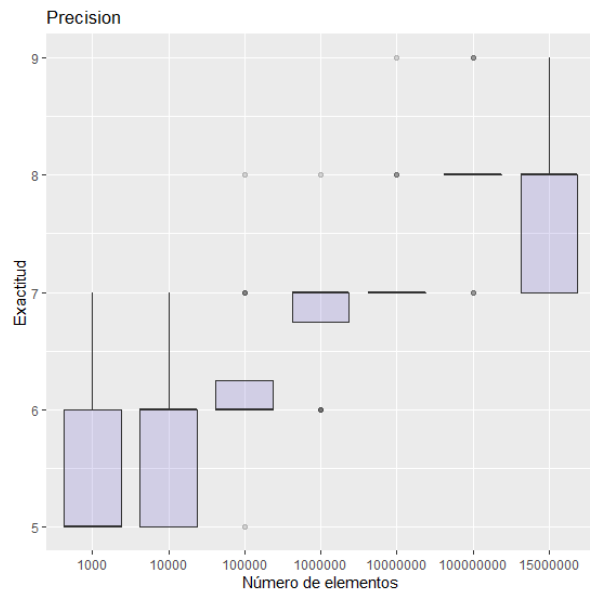
Figura 1: Área de un círculo para despejar.



(a) Diferencia absoluta de la estimación en relación al valor de n .

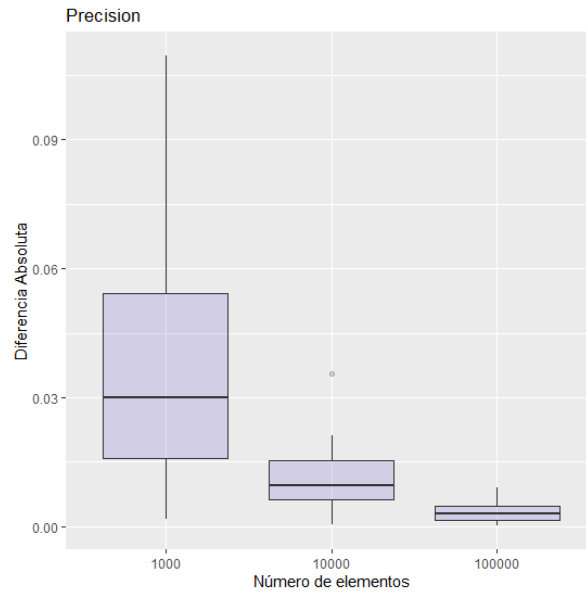


(b) Diferencia cuadrada de la estimación en relación al valor de n .

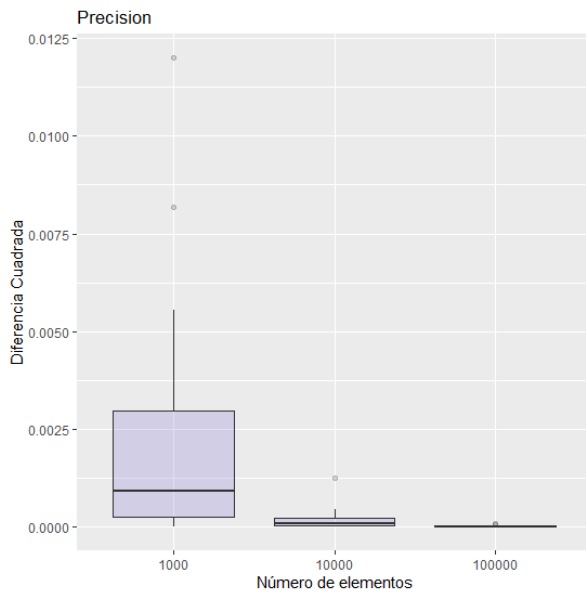


(c) Exactitud de los decimales de la estimación en relación al valor de n .

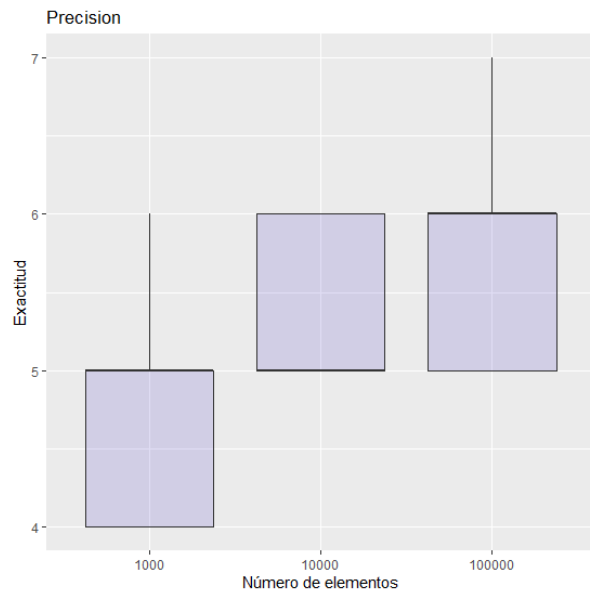
Figura 2: Diferentes análisis de la precisión de la estimación.



(a) Diferencia absoluta de la estimación de π y el valor correcto de π en relación al valor de n



(b) Diferencia cuadrada de la estimación de π y el valor correcto de π en relación al valor de n .



(c) Exactitud de la estimación de π y el valor correcto de π en relación al valor de n .

Figura 3: Diferentes análisis de la precisión de la estimación con respecto al valor correcto de π .

5. Conclusiones.

La estimación del área bajo la curva utilizando el método de Monte-Carlo es útil, sin embargo, para obtener resultados óptimos es necesario aumentar el número de elementos bajo la curva a manera de aumentar la definición de el área. Hay valores que, dependiendo la precisión requerida y el tipo de precisión requerida, nos dan resultados aceptables son rutinas cortas.

Referencias

- [1] I.Crespo. P5:método de monte-carlo. 2022. URL <https://github.com/IsmaelHC/Simulacion-NANO-2022/tree/main/P5>.
- [2] 2022 Wolfram Alpha LLC. 2022. URL <https://www.wolframalpha.com/input?i=integrate>.
- [3] W.Kurt. 6 neat tricks with monte carlo simulations — count bayesie. 2022. URL <https://www.countbayesie.com/blog/2015/3/3/6-amazing-trick-with-monte-carlo-simulations>.