

### Ejercicio 1

Definir la cdf inversa generalizada  $F_X^{-1}$  y demostrar que en el caso de variables aleatorias continuas esta coincide con la inversa usual. Demostrar además que en general para simular de  $X$  podemos simular  $u \sim U(0,1)$  y  $F_X^{-1}(u)$  se distribuye como  $X$ .

#### Solución:

Definimos la inversa generalizada de la variable aleatoria  $X$  como la función

$$F_X^{-1}(t) = \inf\{x \in \mathbb{R} : F_X(x) \geq t\}.$$

Para variables aleatorias continuas sabemos que  $F_X(x)$  es una función monótona creciente en el soporte de  $X$ . Es decir, tomemos  $x, y \in \text{supp}(X)$ , entonces  $x < y$  si y solo si

$$F_X(x) < F_X(y)$$

donde  $F(x)$  es una función biyectiva. Luego, existe su función inversa denotada por  $F_X^{-1}(t)$ . Entonces, usando la invertibilidad de la función de distribución en la definición de inversa generalizada

$$\begin{aligned} F_X^{-1}(t) &= \inf\{x \in \mathbb{R} : F_X(x) \geq t\}, \\ &= \inf\{x \in \mathbb{R} : x \geq F_X^{-1}(t)\} \end{aligned}$$

como  $F_X$  es continua existe  $x_0$  tal que  $F_X(x_0) = t$ , es decir  $x_0 = F_X^{-1}(t)$ . Luego, por la monotonía se sigue que el ínfimo se alcanza y

$$F_X^{-1}(t) = F_X^{-1}(t)$$

que nos muestra que para variables  $X$  continuas su inversa coincide con la inversa generalizada.

Para la segunda demostración, primeramente probaremos lo siguiente. Sea  $X$  una variable aleatoria continua y  $F_X$  su función de distribución, entonces la variable aleatoria  $Y = F_X(X)$  tiene distribución uniforme en el intervalo  $(0,1)$ .

Para probarlo, es inmediato

$$\mathbb{P}(Y \leq y) = \mathbb{P}(F_X(X) \leq y) = \begin{cases} 0 & y < 0 \\ 1 & y \geq 1 \end{cases} \quad \forall x$$

Por otra parte, si  $y \in (0,1)$  o  $y = 0$  y  $F_X(x) = 0$  para alguna  $x$ , definamos:

$$x_0 = \sup\{x \in \mathbb{R} : F_X(x) = y\}$$

Como  $F_X$  es continua, se tiene  $F_X(x_0) = y$ . Además  $F_X(x) > y$  para cualquier  $x > x_0$  y  $F_X(x) \leq y$  para cualquier  $x < x_0$ . Es decir,  $F_X(x) \leq y$  si y sólo si  $x \leq x_0$ . Por tanto

$$\mathbb{P}(F_X(X) \leq y) = \mathbb{P}(X \leq x_0) = F_X(x_0) = y.$$

Así que

$$\mathbb{P}(Y \leq y) = \mathbb{P}(F_X(X) \leq y) = \begin{cases} 0 & y < 0 \\ y & y \in (0,1) \\ 1 & y \geq 1 \end{cases}$$

que es la distribución de la variable aleatoria uniforme en el intervalo  $(0,1)$ . Para mostrar el segundo enunciado del Ejercicio 1, usaremos el resultado previo definiendo  $X = F_X^{-1}(Y)$ . Usando la invertibilidad  $F_X(X) = Y$  en el cálculo con  $Y \sim \text{Unif}(0,1)$ , es directo

$$F_X(q) = \mathbb{P}(X \leq q) = \mathbb{P}(F_X(X) \leq F_X(q)) = \mathbb{P}(Y \leq F_X(q)) = F_X(q)$$

para  $q \in (0,1)$ . Entonces  $X$  tiene como función de distribución  $F_X$ , que es lo que se quería probar.

## Ejercicio 2

Implementar el siguiente algoritmo para simular variables aleatoria uniformes:

$$x_i = 107374182x_{i-1} + 104420x_{i-5} \mod 2^{31} - 1$$

regresa  $x_i$  y recorrer el estado, esto es  $x_{j-1} = x_j; j = 1, 2, 3, 4, 5$ ; ¿Parecen  $U(0, 1)$ ?

### Solución:

Para la implementación se creo una función llamada uniforme que toma dos argumentos,  $n$  es la cantidad de valores que toma de la recursión de congruencia. El segundo argumento es la semilla, si no se ingresa nada, la semilla predeterminada será dada por el tiempo del computador que nos permite acceder la librería time. Con la semilla determinada, se da un vector de valores arbitrarios para ser el vector inicial.

Posterior a los cinco valores iniciales, tomamos un vector  $X_t$  de tamaño  $n$  que nos da la recursión dada pero reescalamos cada valor entre el valor máximo, que es  $2^{31} - 2$ , para tener valores acotados entre cero y uno. Consideremos  $n = 100$  y tomemos un histograma

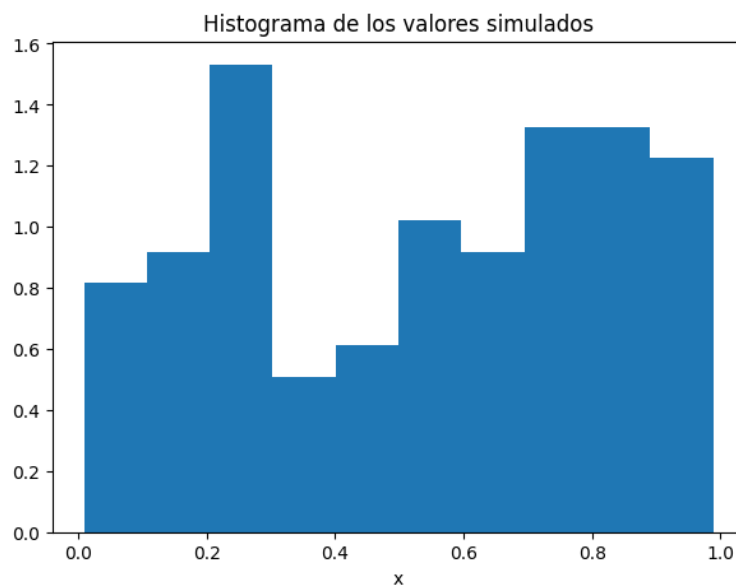


Figura 1: Histograma del generador de congruencia lineal  $n = 100$ .

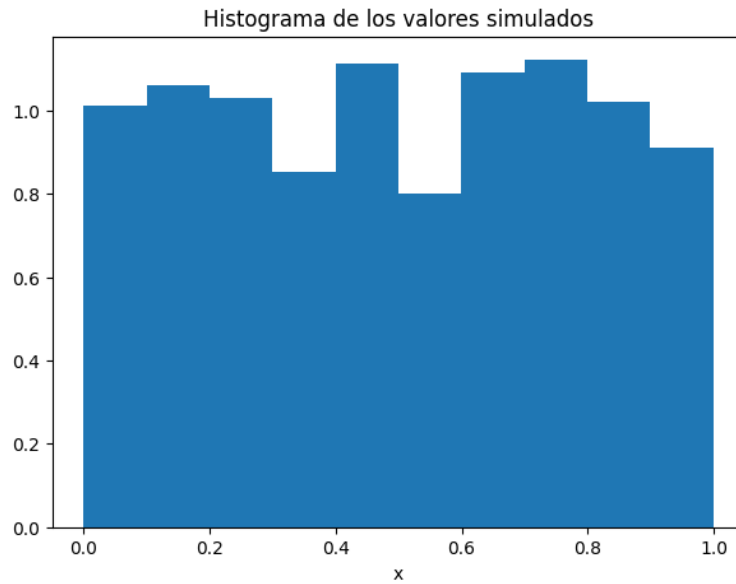


Figura 2: Histograma del generador de congruencia lineal  $n = 1000$ .

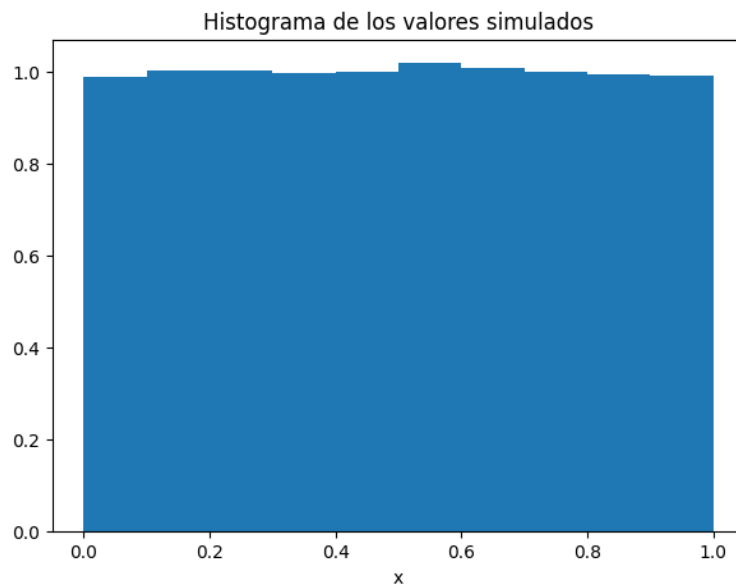


Figura 3: Histograma del generador de congruencia lineal  $n = 100000$ .

Podemos ver que parece converger a una distribución uniforme en  $(0,1)$ . En el código se hace la comparación de este generador con el de scipy. Ambos histogramas tienen comportamientos similares. No podemos afirmar que distribuyen uniforme, sin embargo no tenemos evidencia de que no lo sea. Para poder concluir se requiere de una prueba de ajuste de bondad, sin embargo no es objetivo del ejercicio. Entonces, por lo visto en los histogramas podemos pensar que estamos simulando variables aleatorias uniformes.

**Ejercicio 3**

¿Cuál es el algoritmo que usa `scipy.stats.uniform` para generar números aleatorios? ¿Cómo se pone la semilla? ¿y en R?

**Ejercicio 4**

¿En `scipy` que funciones hay para simular un variable aleatoria genérica discreta? ¿tienen preprocesos?

**Ejercicio 5**

Implementar el algoritmo Adaptive Rejection Sampling y simular de una  $Gama(2, 1)$  10,000 muestras. ¿cuando es conveniente dejar de adaptar la envolvente? (vea alg. A.7, p 54 Robert y Casella, 2da ed.)