GENERACIÓN DE NÚMEROS PSEUDO-ALEATORIOS.

VIDAL ALCALÁ.

En los modelos de simulación es común encontrarse con variables aleatorias con diversas distribuciones. Los lenguajes de programación vienen con generadores de números pseudo-aleatorios que funcionan bien en general y ofrecen una amplia gama de distribuciones. Sin embargo, de vez en cuando aparecen distribuciones que no están implementadas y se vuelve indispensable saber como crear nuestros propios generadores.

1. Notación.

Supongamos que X es una variable aleatoria. La función de distribución de X, $F_X : \mathbb{R} \to [0, 1]$ se define como

$$F_X(x) = P(X \leq x)\,. \tag{1}$$

La función de densidad $f_X : \mathbb{R} \to \mathbb{R}$ se define mediante la ecuación

$$P(X \in (a, b)) = \int_a^b f_X(y)\, dy, \qquad a \leq b\,. \tag{2}$$

A menos que lo especifique de otra manera, voy a asumir que las variables aleatorias tienen función de densidad f_X continua.

2. Densidad uniforme en $(0, 1)$.

La variable aleatoria U tiene densidad uniforme en el intervalo $[0, 1]$ si

$$P(U \in [r, s]) = r - s, \qquad 0 < r \leq s < 1\,.$$

Se sigue que la probabilidad de que U pertenezca a un intervalo no depende de la posición del intervalo y solo depende de la longitud del intervalo. En términos de funciones de distribución y densidad tenemos

$$F(u) = P(U \leq u) = \begin{cases} 0 & u < 0 \\ u & 0 \leq u \leq 1 \\ 1 & 1 < u \end{cases} \tag{3}$$

$$f(u) = F'(u) = \begin{cases} 0 & u < 0 \\ 1 & 0 \leq u \leq 1 \\ 0 & 1 < u \, . \end{cases} \tag{4}$$

Escribiremos $X \sim U(0,1)$ para denotar que la variable aleatoria X tiene distribución uniforme en el intervalo $(0,1)$.

3. Método de congruencias lineales.

El objetivo es construir una secuencia $u_1, u_2, \ldots, u_L$ de números que sean *aproximadamente* aleatorios , independientes e idénticamente distribuidos, con distribución $U(0,1)$. El método de congruencia lineal define una secuencia de *semillas* s_n mediante la iteración

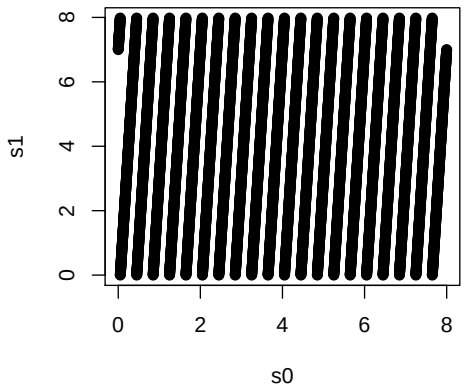
$$s_{n+1} = (as_n + c) \bmod m \, .$$

La siguiente figura muestra la gráfica de la función $s_0 \to (as_0 + c) \bmod m$ con los parámetros $a = 5, c = 7, m = 8$.

```
a <- 20
c <- 7
m <- 8
# definir funcion x (mod m) para x,y > 0
mod <- function(x, m) {
    ((x/m) - floor(x/m)) * m
}

s0 <- seq(0, m, 0.001)
s1 <- mod(a * s0 + c, m)
```

```
plot( s0 , s1 )
```



La secuencia de semillas s_n define la secuencia u_n mediante la ecuación $u_n = s_n/m$. Es claro que la secuencia u_n no es aleatoria, es *pseudo-aleatoria*. Sin embargo, para el propósito de simulaciones tipo Monte Carlo, la propiedad que debe satisfacer es que

$$\frac{1}{L}\sum_{n=1}^{L} f(u_n) \sim E[f(U)]\,, \tag{5}$$

para L grande donde U es una variable aleatoria con distribución normal y f es una función continua en el intervalo $[0, 1]$.

Cualquier generador de este tipo tiene solo m posible valores para las semillas s_n y por lo tanto la secuencia s_n se cicla en m o menos iteraciones y por lo tanto no es razonable esperar que la ecuación (5) se cumpla para $L > m$. El generador *Learmouth-Lewis* usa $a = 7^5$ y $m = 2^{31} - 1$ y generadores mas sofisticados usan $m \sim 2^{128} \sim 10^{38}$. El generador Mersenne-Twister , utilizado en la rutina `runif()` de R, se cicla después de $2^{19937} \sim 10^{6001}$ iteraciones y es el preferido en aplicaciones.

Un método importante en aplicaciones es la habilidad de ajustar la semilla del generador. Esto permite reproducir resultados como se hace en el siguiente ejemplo.

```
set.seed(1001)
## salva la semilla actual
save(".Random.seed", file = "random_state_seed1001.RData")
runif(1)
## [1] 0.9857
```

```
runif(1)
## [1] 0.4126
runif(1)
## [1] 0.4295
## Restaurar la semilla después de set.seed()
load("random_state_seed1001.RData")
runif(1)
## [1] 0.9857
```

También es posible continuar una simulación que ha sido interrumpida si salvamos la semilla después de cada cierto número de iteraciones y reiniciamos la simulación con la última semilla guardada.

4. Método de la transformada inversa.

El objetivo es generar muestras de una variable aleatoria X a partir de una variable aleatoria uniforme U. En realidad es más fácil generar una variable uniforme a partir de X y su función de distribución F_X. Simplemente define $U = F_X(X)$. Asumiendo que $F_X : \mathbb{R} \to (0,1)$ es continua y estrictamente creciente podemos calcular para $0 < u < 1$

$$\begin{aligned} P[F_X(X) \leq u] &= P[X \leq F_X^{-1}(u)] \\ &= F_X(F_X^{-1}(u)) \\ &= u\,. \end{aligned} \tag{6}$$

Este cálculo muestra que $F_X(X)$ tiene función de distribución uniforme (ver ecuación (3)).El siguiente Teorema nos da condiciones suficientes para ejecutar el procedimiento inverso en los casos de interés en aplicaciones.

Theorem 4.1. *Supongamos que F_X es la función de distribución de la variable aleatoria X con valores en un intervalo $I = (a, b)$ (no excluimos la posibilidad $a = -\infty$ o $b = +\infty$) y que $F_X : \mathbb{R} \to (0,1)$ es estrictamente creciente y continua en el intervalo I. Entonces existe la inversa por la derecha $F_X^{-1} : (0,1) \to I$ y esta inversa es una función estrictamente creciente. Esto quiere decir que para cada $u \in (0,1)$ tenemos*

$$F_X(F_X^{-1}(u)) = u, \qquad 0 < u < 1\,, \tag{7}$$

y además $0 < u_1 \leq u_2 < 1$ si y solo si $F_X^{-1}(u_1) \leq F_X^{-1}(u_2)$.

Bajo las hipótesis del Teorema anterior tenemos que $F_X^{-1}(U) \leq x$ y $F_X(F_X^{-1}(U)) \leq F_X(x)$ son el mismo evento y por lo tanto

$$\begin{aligned} P[F_X^{-1}(U) \leq x] &= P[F_X(F_X^{-1})(U) \leq F_X(x)] \\ &= P[U \leq F_X(x)] \\ &= F_X(x)\,. \end{aligned} \tag{8}$$

Se sigue que la variable aleatoria $F_X^{-1}(U)$ tiene función de distribución F_X. Cuando sea claro cuál es la variable aleatoria X en cuestión, escribiremos F en lugar de F_X.

La discusión anterior sugiere el siguiente algoritmo para generar L muestras independientes de una variable aleatoria X con función de distribución $F(x)$, denotadas por $x_1, x_2, \ldots, x_L$.

1. Genera L muestras aleatorias i.i.d $u_n \sim U[0,1]$.
2. Encuentra las soluciones x_n de las ecuaciones $F(x_n) = u_n$ (equivalente a $x_n = F^{-1}(u_n)$).

Por supuesto, las muestras x_n son independientes ya que las muestras uniformes u_n lo son. El segundo paso del algoritmo se resuelve de manera analítica en la mayoría de las aplicaciones. Ilustraremos la metodología con un ejemplo.

La distribución exponencial con parámetro $\lambda > 0$ tiene función de densidad $f(x) = \lambda e^{-\lambda x}$ para $x > 0$ y función de distribución

$$F(x) = 1 - e^{-\lambda x}\,. \tag{9}$$

Es común modelar tiempos de espera con la distribución exponencial. La media de la distribución es $\frac{1}{\lambda}$ y la varianza es $\frac{1}{\lambda^2}$.

Primero escribimos x en función de u usando la ecuación $F(x) = u$.

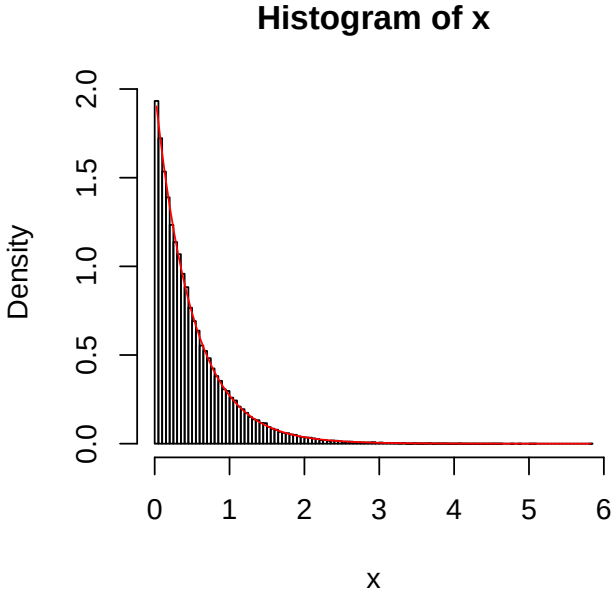
$$\begin{aligned} 1 - e^{-\lambda x} &= u \\ 1 - u &= e^{-\lambda x} \\ \log(1-u) &= -\lambda x \\ -\frac{\log(1-u)}{\lambda} &= x \end{aligned}$$

Entonces, hemos obtenido que $F^{-1}(u) = -\frac{\log(1-u)}{\lambda}$. Ahora generamos L muestras uniformes y las transformamos usando $F^{-1}(u)$.

```
## parametros
L = 1e+05
lambda <- 2
u <- runif(L)
x <- -(1/lambda) * log(1 - u)
```

Finalmente comparamos el histograma de las muestras x_n con la densidad exponencial $f(x) = \lambda e^{-\lambda x}$.

```
xHist <- hist(x, prob = TRUE , breaks = 10*log(L))
curve( lambda*exp(-lambda*x), xHist$mids[1] , xHist$mids[length(xHist$mids)] ,
       add = TRUE, col = 'red')
```



El método de la transformación inversa se puede aplicar también entre dos variables aleatorias X, Y relacionadas mediante $Y = G(X)$. Aquí necesitamos asumir que la función G es diferenciable y $G'(x) \neq 0$ para cada valor x que pueda tomar X. En este caso tenemos

$$f_Y(y) = \pm \frac{f_X(G^{-1}(y))}{G'(G^{-1}(y))} \,. \tag{10}$$

En realidad, es más fácil explicar la metodología en un cambio de variable en particular en lugar de usar la fórmula anterior. El siguiente ejemplo ilustra el procedimiento. Supongamos que X es una v.a. exponencial con parámetro λ y $Y = X^2$.

1. Escribir la función de densidad de X en *notacion* diferencial.

$$f_X(x)dx = \lambda e^{-\lambda x} dx, \qquad x > 0 \,. \tag{11}$$

2. Sustituir x y dx en la ecuación anterior usando la relación $y = x^2$. Despejando x obtenemos

$$x = \sqrt{y} x > 0\,, \tag{12}$$

y calculando la diferencial el resultado es

$$\begin{aligned} dy = d(x^2) &= 2\,x\,dx \\ \frac{dy}{2\,x} &= dx \\ \frac{dy}{2\sqrt{y}} &= dx \end{aligned} \tag{13}$$

Finalmente sustituimos (12) y (13) en la ecuación (11) para obtener

$$f_X(x)dx = \lambda \frac{e^{-\lambda\sqrt{y}}}{2\sqrt{y}} dy, \qquad y > 0\,. \tag{14}$$

3. El factor enfrente del diferencial dy es la función de densidad de la variable aleatoria Y, en otras palabras

$$f_Y(y) = \lambda \frac{e^{-\lambda\sqrt{y}}}{2\sqrt{y}}, \qquad y > 0\,. \tag{15}$$

5. Ejercicios

1. Genera 10,000 muestras independientes de la variable aleatoria X con función de densidad

$$f(x) = \frac{1}{4}(x+1)^3, \qquad -1 < x < 1\,. \tag{16}$$

Compara el histograma de tus muestras con la función de densidad (16) para comprobar que tus muestras tienen la distribución correcta.

2. Considera una variable aleatoria T con función densidad

$$f(t) = \frac{1}{\sqrt{2\pi t^3}} e^{-1/2t}, \qquad t > 0\,. \tag{17}$$

   a) Calcula la función de densidad de $X = \frac{1}{\sqrt{T}}$.

   b) Genera 10,000 muestras independientes de T usando la función `rnorm()` y una transformación inversa. Compara el histograma de tus muestras con la función de densidad (17) para comprobar que tus muestras tienen la distribución correcta.

3. Genera 10,000 muestras independientes de la variable aleatoria X con función de densidad

$$f(x) = xe^{-x^2/2}, \qquad x > 0\,. \tag{18}$$

Compara el histograma de tus muestras con la función de densidad (16) para comprobar que tus muestras tienen la distribución correcta.

6. Método de aceptación-rechazo.

Cuando no es posible calcular la función de distribución F de una variable aleatoria X, debemos utilizar otros métodos que no dependan de la integración de la función de densidad f. El método de aceptación-rechazo utiliza muestras de una v.a. Y para generar muestras de X. La idea es transformar las probabilidades de que la v.a. tome un valor en lugar de transformar los valores mismos. Supongamos que tenemos una función $p : \mathbb{R} \to (0, 1]$ la cual llamaremos la *probabilidad de aceptación*. Ahora supongamos que las muestras de X se generan a partir de muestras de Y con el siguiente algoritmo.

1. Genera una muestra Y_1 de la v.a. Y.
2. Con probabilidad $p(Y_1)$ declara que la muestra de X es $\overline{X} = Y_1$ y con probabilidad $1 - p(Y_1)$ regresa al paso (1).

Podemos calcular la función de densidad de la v.a. que se ha generado con el algoritmo anterior de la siguiente forma. Denotamos por N el paso en el que se acepta la muestra de la v.a. Y.

$$\begin{aligned}
P(\overline{X} = x) &= \sum_{n=1}^{\infty} P(Y_n = x, N = n, N > (n-1)) \\
&= \sum_{n=1}^{\infty} P(N = n | Y_n = x, N > (n-1)) P(Y_n = x, N > (n-1)) \\
&= \sum_{n=1}^{\infty} P(N = n | Y_n = x, N > (n-1)) P(Y_n = x) P(N > (n-1)) \qquad (19) \\
&= p(x) f_Y(x) \sum_{n=1}^{\infty} P(N > (n-1)) \\
&= p(x) f_Y(x) \sum_{n=1}^{\infty} P(N > (n-1))\,.
\end{aligned}$$

Es un ejercicio de probabilidad el mostrar que

$$\sum_{n=1}^{\infty} P(N > (n-1)) = E[N] \,. \tag{20}$$

Por lo tanto hemos obtenido que la función de densidad de $\overline{X}$ satisface

$$f_{\overline{X}}(x) = \frac{1}{Z} p(x) f_Y(x) \,, \tag{21}$$

con Z una constante. Para que $f_{\overline{X}}(x)$ integre a 1 debemos tener

$$Z = \int_{-\infty}^{\infty} p(y) f_Y(y) dy \,, \tag{22}$$

y de acuerdo a el cálculo anterior también podemos asegurar que

$$Z = \frac{1}{E[N]} \,. \tag{23}$$

De cualquier forma, no es necesario calcular la constante Z para implementar el método de aceptación-rechazo.

Nuestro objetivo es obtener muestras de una variable aleatoria con función de distribución f_X y por lo tanto debemos escoger $p(x)$ tal que

$$\begin{aligned} f_X(x) &= \frac{1}{Z} p(x) f_Y(x) \\ p(x) &= \frac{1}{Z} \frac{f_X(x)}{f_Y(x)} \,. \end{aligned} \tag{24}$$

La ecuación anterior solo tiene sentido si

$$0 \leq \frac{f_X(x)}{f_Y(x)} < +\infty \,, \tag{25}$$

o en otras palabras, si $f_X(x) = 0$ cuando $f_Y(x) = 0$. Cuando la condición (28) se cumple, decimos que X *es absolutamente continua con respecto a* Y. Para que el método de aceptación rechazo funcione hay que asumir una condición aún más fuerte. Vamos a suponer que existe una constante M tal que

$$0 \leq \frac{f_X(x)}{f_Y(x)} < M \,. \tag{26}$$

Ilustraremos la metodología con un ejemplo. Supongamos que queremos generar 10,000 muestras de la v.a. normal X con densidad

$$f_X(x) \propto e^{-x^2/2}, \qquad x > 0 \,, \tag{27}$$

usando muestras de la v.a. exponencial Y con densidad

$$f_Y(x) \propto e^{-x}, \qquad x > 0\,, \tag{28}$$

```
## Limpiar variables
rm(list = ls())

## parametros
L <- 10000
```

El primer paso es verificar si la condición (26) se cumple.

$$\begin{aligned}\frac{f_X(x)}{f_Y(x)} &= \frac{e^{-x^2/2}}{e^{-x}} \\ &= e^{x-x^2/2} \\ &\leq e^{1/2}\end{aligned}$$

```
## cota superior
M <- exp(1/2)
```

El segundo paso es escoger $p(x)$ proporcional a $\frac{f_X(x)}{f_Y(x)}$, esto es

$$p(x) \propto e^{x-x^2/2}\,.$$

La mayor constante que podemos elegir en la relación de proporcionalidad anterior , manteniendo el requerimiento $p(x) < 1$ para $x > 0$ es $e^{-1/2}$. Por lo tanto escogemos

$$p(x) = \frac{1}{e^{1/2}} e^{x-x^2/2}\,.$$

Tiene sentido hacer $p(x)$ lo más grande posible ya que esto aumenta la probabilidad de que las muestras de Y sean aceptadas y por lo tanto se reduce el número de pasos N en el algoritmo de aceptación-rechazo.

```
## definir p(x)
p <- function(x) {
    return((1/M) * exp(x - x * x/2))
}
```

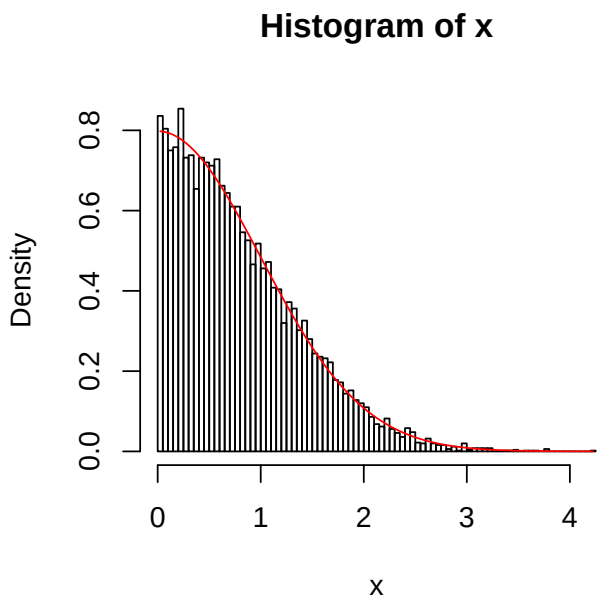
Finalmente, generamos las muestras de X usando el método de aceptación-rechazo.

```
## Generar L muestras de X
x <- c()
y <- c()
for (i in 1:L) {
    y <- rexp(1)
    pAceptar <- p(y)
    pAceptar
    u <- runif(1)
    while (u > pAceptar) {
        y <- rexp(1)
        pAceptar <- p(y)
        u <- runif(1)
    }
    x[i] <- y
    x
}
```

Verificamos que el método es correcto.

```
# El histograma para las muestras de X
xHist <- hist(x, prob = TRUE , breaks = 10*log(L))

# La densidad de X
curve( 2.0*dnorm(x), xHist$mids[1] , xHist$mids[length(xHist$mids)] ,
       add = TRUE, col = 'red')
```

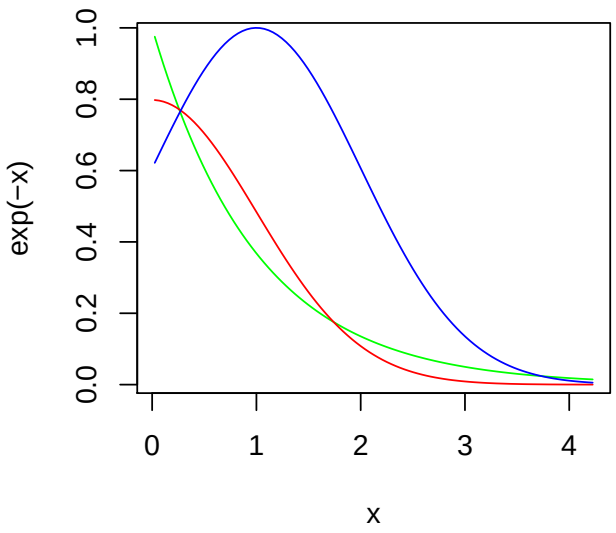



```
#crear una nueva figura
plot.new()

# La densidad de Y
 curve( exp(-x), xHist$mids[1] , xHist$mids[length(xHist$mids)] , col = 'green')

# La densidad de X
curve( 2.0*dnorm(x), xHist$mids[1] , xHist$mids[length(xHist$mids)] , add = TRUE ,

# La probabilidad de aceptación
curve( p(x), xHist$mids[1] , xHist$mids[length(xHist$mids)] ,
       add = TRUE, col = 'blue')
```



7. Ejercicios.

Genera 10,000 muestras de la variable aleatoria X con función de densidad

$$f_X(x) \propto (x-\mu)\exp\left(\frac{-x^2}{2}\right), \qquad x > \mu > 0\,,$$

usando el método de aceptación rechazo con muestras de la variable aleatoria Y con función de densidad

$$f_Y(y) = y\ \exp\left(\frac{-y^2}{2}\right), \qquad y > 0\,.$$