Capítulo 3.3: Generación de Variables Aleatorias: Métodos particulares

La distribución normal

Simulación

2020-11-12

Contents

1 Métodos particulares

1.1 Distribución Normal

Debido a la importancia de la Normal y a que no hay una forma "mejor" de generar valores de la Normal hay una gran variedad de algoritmos para generarla. Por otra parte, dado que si $Z \sim N(0,1)$, entonces $X = \mu + \sigma Z \sim N(\mu, \sigma)$ es suficiente considerar la generación de valores de una normal estándar.

A continuación veremos algunos de estos algoritmos.

1.1.1 Aplicación del teorema Central del Límite

Sea $U_1, U_2, \ldots, U_{12}$ VA i.i.id según una $U(0,1)$, se verifica que

$$Z = \left[\sum_{i=1}^{12} U_i - 6\right] \sim N(0,1)$$

Este generador funciona razonablemente bien pero no es difícil encontrar otros más eficientes.

1.1.2 Método de aceptación y rechazo

En este caso buscamos una VA Y con el mismo soporte de la $N(0,1)$, es decir $\mathbb{R}$.

Sea Z VA con distribución $N(0,1)$ y comprobemos que se puede descomponer de la forma siguiente,

$$\Pr(Z \leq a) = \frac{1}{2}\Pr(|Z| \leq a) + \frac{1}{2}\Pr(-|Z| \leq a)$$

si esto se verifica podremos generar Z utilizando un método de composición.

Para $a > 0$

$$\Pr(Z \leq a) = \frac{1}{2}\Pr(|Z| \leq a) + \frac{1}{2}\Pr(-|Z| \leq a)$$

donde,

- $\Pr(-|Z| \leq a) = \Pr(|Z| \geq a) = 1$
- $\Pr(|Z| \leq a) = \Pr(X \leq 0) + \Pr(0 \leq Z \leq a) = \frac{1}{2} + \Pr(0 \leq Z \leq a) = \Pr(Z \leq a)$

Para $a < 0$

$$\Pr(Z \leq a) = \frac{1}{2}\Pr(|Z| \leq a) + \frac{1}{2}\Pr(-|Z| \leq a)$$

- $\Pr(|Z| \leq a) = 0$

$$\begin{aligned}
\Pr(Z \leq a) &= \tfrac{1}{2}\Pr(|Z| \leq a) + \tfrac{1}{2}\Pr(-|Z| \leq a) \\
&= \tfrac{1}{2}\Pr(-|Z| \leq a) \\
&= \tfrac{1}{2}\Pr(|Z| \ -a) \\
&= \tfrac{1}{2}\left[\Pr(Z \leq a) + \Pr(Z \geq -a)\right] \\
&= \tfrac{1}{2} \cdot 2\Pr(Z \leq a) = \Pr(Z \leq a)
\end{aligned}$$

Luego,

$$F_Z = \frac{1}{2}F_{|Z|} + \frac{1}{2}F_{-|Z|}$$

el algoritmo es

Algoritmo

Paso 1: Generar $U \sim U(0,1)$
Paso 2: Si $U < 0.5$, generar $Y \sim F_{|Z|}$
Paso 3: Si $U \geq 0.5$, generar $Y \sim F_{-|Z|}$
Paso 4: Devolver $X = Y$

Sólo queda encontrar un método para generar $|Z|$.

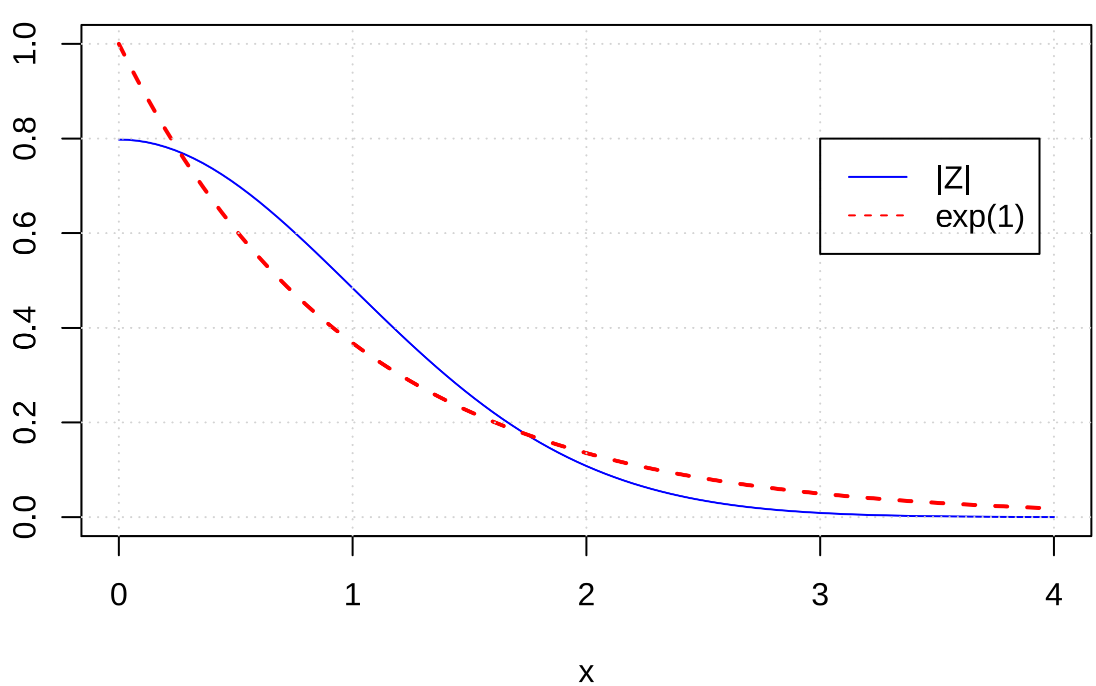
Función de distribución de $|Z|$

$$\Pr(|Z| \leq a) = \Pr(-a \leq Z \leq a) = 2\Pr(0 \leq Z \leq a) = 2\int_0^a \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}\, dx = \int_0^a \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}}\, dx$$

La función de densidad de $|Z|$ es $f_{|Z|}(z) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}}, \quad Z \in [0, \infty)$.

Generamos $f_{|Z|}(z)$ por el método de aceptación y rechazo y tomamos como g una $\exp(\lambda = 1)$

```r
# gráfico
fd <- function(x){
  x <- sqrt(2/pi) * exp(- x^2/2)
}

curve(fd, from = 0, to = 4, col = "blue", lty = 1, ylim = c(0, 1), ylab = "")
curve(dexp, from = 0, to = 4, col = "red", lty = 2,  lwd = 2, add = TRUE)
grid()
legend(3, 0.8, c("|Z|", "exp(1)"), lty = 1:2, col = c("blue", "red"))
```



calculamos $c > 0$.

$$h(x) = \frac{f(x)}{g(x)} = \frac{\sqrt{\frac{2}{\pi}}e^{-\frac{x^2}{2}}}{e^{-x}} = \sqrt{\frac{2}{\pi}}e^{-\frac{x^2}{2}+x} = \sqrt{\frac{2}{\pi}}e^{-\frac{1}{2}(x^2-2x+1-1)} = \sqrt{\frac{2}{\pi}}e^{-\frac{1}{2}(x-1)^2}e^{\frac{1}{2}} = \sqrt{\frac{2e}{\pi}}e^{-\frac{1}{2}(x-1)^2}, \quad [x, \infty)$$

el $\max h(x) = \sqrt{\frac{2e}{\pi}}$ cuando $x^* = 1$. Luego $c = \sqrt{\frac{2e}{\pi}} = 1.3154892$. El numero medio de iteraciones para obtener un valor de $|Z|$ es $c = 1.3154892$

El algoritmo queda

Algoritmo

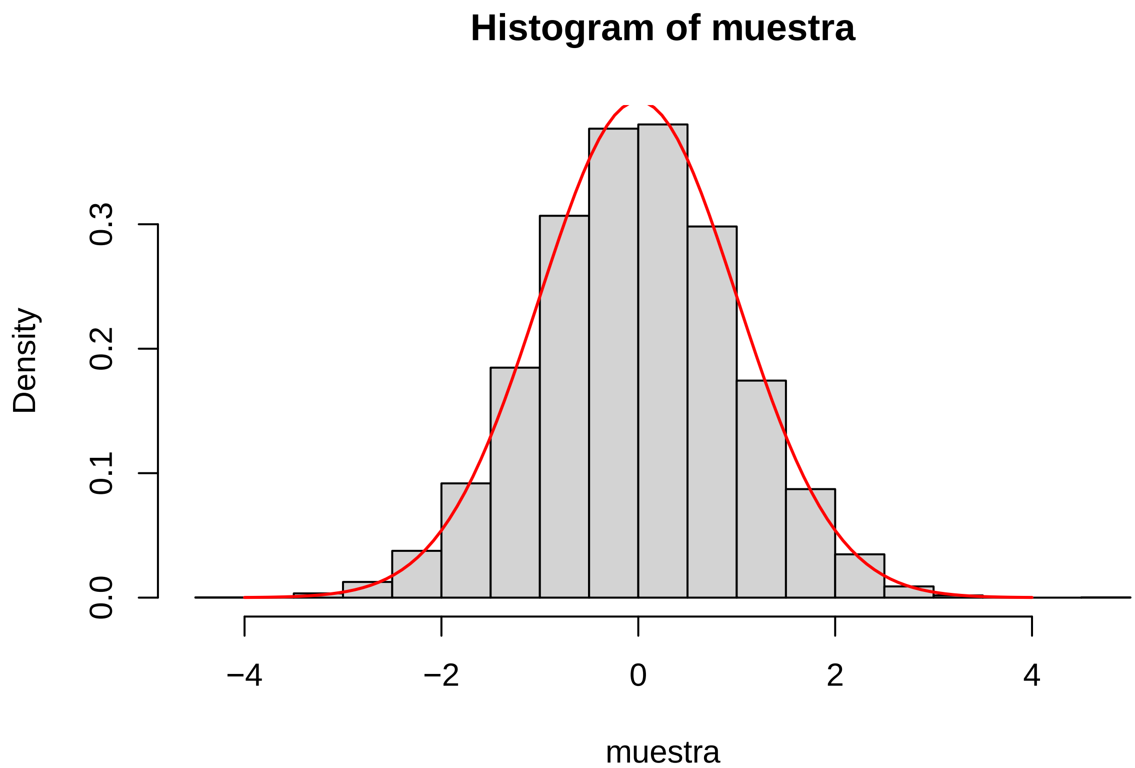
Paso 1: Generar $U_1 \sim U(0,1)$
Paso 2: Mientras $U_1 \geq f(y)/cg(y)$
Paso 3: Generar $U_1 \sim U(0,1)$
Paso 4: Generar $Y \sim \exp(1)$
Paso 5: FinMientras
Paso 6: Devolver $X = Y$

En R,

```
#
generar_nor <- function(){
  c  <- sqrt(2*exp(1)/pi)
  U1 <- runif(1)
  test <- TRUE
  while(test){
    U1 <- runif(1)
    Y  <- rexp(1,1)
    test <- U1 > (fd(Y)/(c*dexp(Y)))
  }
  U2 <- runif(1)
  if (U2 < 0.5){
    X <- Y
  } else {
    X <- -Y
  }
  X
}


(generar_nor()    )
```

```
## [1] 0.4464497
```

```
# Muestra
Nsim <- 10^4

muestra <- replicate(Nsim,generar_nor())

# Histograma
hist(muestra, freq = FALSE)
curve(dnorm, from = -4, to = 4, col = "red", lwd = 1.5, add = TRUE)
```




```r
# Contraste de hipótesis
ks.test(muestra, "pnorm")
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  muestra
## D = 0.012689, p-value = 0.07989
## alternative hypothesis: two-sided
```

1.1.3 Método de Box-Muller

Supongamos que tenemos dos VA normales estándar X e Y independientes, $X, Y \in \mathbb{R}$. Generamos dos valores de X e Y. El par $(x, y) \in \mathbb{R}^2$ determinan un punto P en el plano en coordenadas cartesianas.

Escribimos el punto P en coordenadas polares, $P = (R, \theta)$ donde $R = distancia(O, P)$ y θ es el ángulo que forma el eje de X con el vector P. La relación es,

$$\begin{cases} X & = R \cdot cos(\theta) \\ Y & = R \cdot sin(\theta) \end{cases}$$

ahora como R es una distancia lo podemos escribir,

$$\begin{cases} X & = \sqrt{R^2} \cdot cos(\theta) \\ Y & = \sqrt{R^2} \cdot sin(\theta) \end{cases}$$

La razón para escribirlo de esta forma es que la distribución de R^2 es de las conocidas y fácil de generar.

En resumen, $R^2 \in [0, \infty)$ y $\theta \in [0, 2\pi]$ y calculamos la distribución conjunta de (R^2, θ).

Función de densidad conjunta de (X, Y)

Las VA X e Y son independientes, por tanto

$$f_{X,Y}(x,y) = f_X(x) \cdot f_Y(y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} = \frac{1}{2\pi} e^{-\frac{1}{2}(x^2+y^2)}, \qquad (x,y) \in \mathbb{R}^{\not 2}$$

por otra parte,

$$X^2 + Y^2 = (R^2 \cdot cos^2(\theta)) + (R^2 \cdot sin^2(\theta)) = R^2(cos^2(\theta) + sin^2(\theta)) = R^2$$

realizamos el cambio de variable,

$$f_{R^2,\theta}(d,\alpha) = f_{X,Y}(d,\alpha)|J|$$

el jacobiano es,

$$J(x,y;d,\theta) = \left[\begin{array}{ll} \frac{\partial x}{\partial d} = \frac{1}{2\sqrt{d}} \cdot cos(\alpha) & \frac{\partial x}{\partial \alpha} = -\sqrt{d} \cdot sin(\alpha) \\ \frac{\partial y}{\partial d} = \frac{1}{2\sqrt{d}} \cdot sin(\alpha) & \frac{\partial x}{\partial \alpha} = \sqrt{d} \cdot cos(\alpha) \end{array} \right]$$

entonces,

$$|J| = \left| (\frac{1}{2\sqrt{d}} \cdot cos(\alpha) \cdot \sqrt{d} \cdot cos(\alpha)) - (-\sqrt{d} \cdot sin(\alpha) \cdot \frac{1}{2\sqrt{d}} \cdot sin(\alpha)) \right| = \frac{1}{2} cos^2(\alpha) + \frac{1}{2} sin^2(\alpha) = \frac{1}{2}$$

luego,

$$f_{R^2,\theta}(d,\alpha) = \frac{1}{2\pi} e^{-\frac{d}{2}} \cdot \frac{1}{2} = \frac{1}{4\pi} e^{-\frac{d}{2}}, \quad d \geq 0, \;\; \alpha \in [0, 2\pi]$$

esta función de densidad la podemos escribir como,

$$f_{R^2,\theta}(d,\alpha) = \frac{1}{2\pi} \mathbb{I}_{[0,2\pi]}(\alpha) \cdot \frac{1}{2} e^{-\frac{d}{2}}$$

esto es,

$$\left. \begin{array}{l} f_\theta(\alpha) \sim \mathcal{U}(0,1) \\ f_{R^2}(d) \sim \mathcal{E}xp(0,1) \end{array} \right\}$$

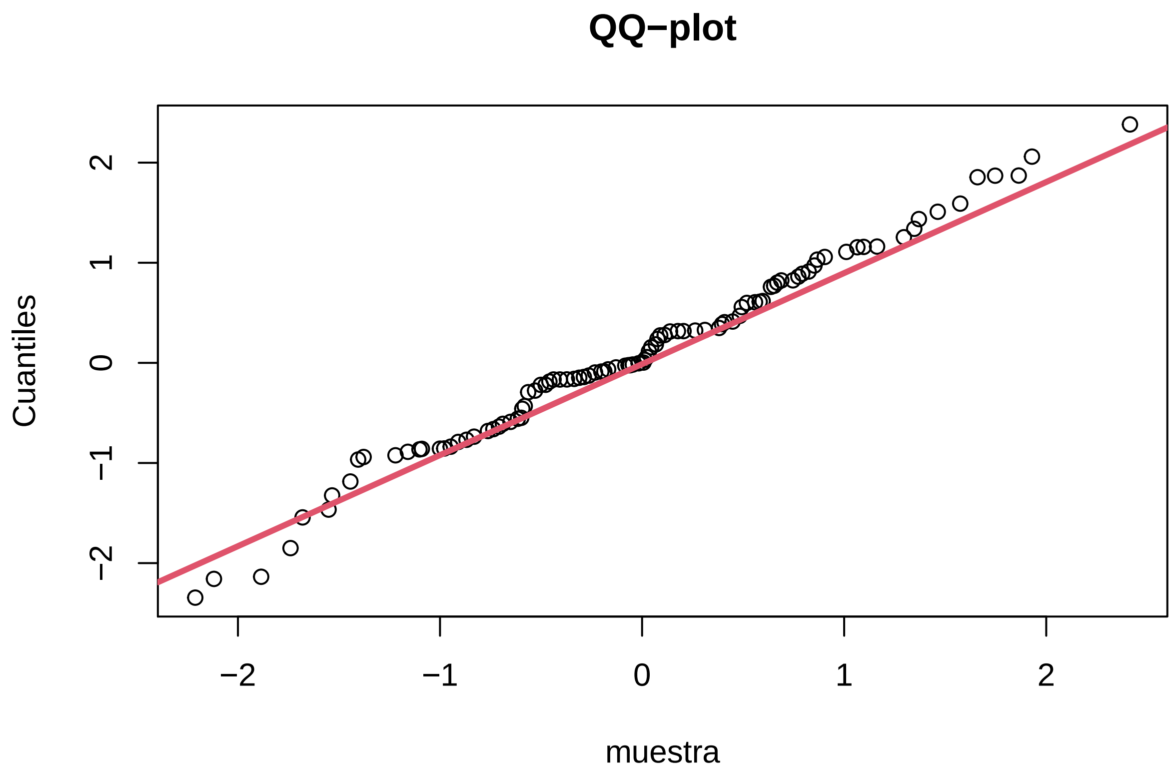
las dos de fácil generación.

El algoritmo de Box-Muller

```r
normalBi <- function(){
  R_2   <- - log(1 - runif(1))*2     # Generar exp(1/2)
  Theta <- runif(1)*2*pi             # Generar U(0,1)
  X <- sqrt(R_2)*cos(Theta)
  Y <- sqrt(R_2)*sin(Theta)
  return(c(X,Y))
}

# La función genera dos valores independientes de una N(0,1)
normalBi()
```

```
## [1] -2.7186342  0.3715484
```

```r
# Simulación
set.seed(711)
Nsim <- 100

muestra <- replicate(Nsim, normalBi())

# Gráfico
qqplot(muestra, rnorm(Nsim), ylab = "Cuantiles", main = "QQ-plot")
qqline(muestra, col = 2, lwd = 3)
```




```r
# Contraste estadístico
ks.test(muestra, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  muestra
## D = 0.057033, p-value = 0.5335
## alternative hypothesis: two-sided
```

1.1.4 Método de cociente de uniformes

Sea X una VA continua con función de densidad f y consideramos el conjunto,

$$C_f = \{(u, v) : 0 < u < \sqrt{f(v/u)}\}$$

Se cumple:

1.- C_f tiene área finita

2.- Si (U, V) sigue una distribución uniforme en la región C_f, entonces la variable aleatoria $X = V/U$ se distribuye según f.

Demostración

1.- C_f tiene área finita

La región $C_f = \{(u, v) : 0 < u < \sqrt{f(v/u)}\}$, hacemos el cambio de variable,

$$\left.\begin{array}{l} X = \frac{V}{U} \\ Y = U \end{array}\right\} \Rightarrow \left.\begin{array}{l} V = XY \\ U = Y \end{array}\right\} \Rightarrow J(u,v;x,y) = \frac{\partial(U,V)}{\partial(X,Y)} \begin{vmatrix} 0 & 1 \\ y & 0 \end{vmatrix} = -y$$

luego el área de $C_f = \{(u,v) : 0 < u < \sqrt{f(v/u)}\} = C_x(f) = \{x \text{ tal que } \exists\ y : 0 \leq y \leq \sqrt{f(x)}\}$ es

$$\int\int_{C_f} dudv = \int_{C_f} \int_0^{f(x)} y\ dy\ dx = \int_{C_f} \frac{f(x)}{2}\ dx \leq \frac{1}{2} \int_{\mathbb{R}} f(x)\ dx < +\infty$$

2.- Si (U,V) sigue una distribución uniforme en la región C_f, entonces la variable aaleatoria $X = V/U$ se distribuye según f.

La función de densidad conjunta es

$$f_{X,Y}(x,y) = \frac{1}{|C_f|} \cdot y; \qquad 0 < y < \sqrt{f(x)}$$

y la función de densidad marginal de X es,

$$f_X(x) = \int_{-\infty}^{\infty} f_{X,y}(x,y)dy = \int_0^{\sqrt{f(x)}} \frac{1}{|C_f|} \cdot ydy = \frac{1}{|C_f|} \cdot \left[\frac{y^2}{2}\right]_0^{\sqrt{f(x)}} = \frac{1}{|C_f|} \cdot \frac{f(x)}{2}$$

tenemos,

- $f_X(x) = \frac{1}{|C_f|} \cdot \frac{f(x)}{2}$
- $f(x)$

dos funciones de densidad, por lo tanto $|C_f| \cdot 2 = 1 \Rightarrow |C_f| = \frac{1}{2} \quad \forall f,$ función de densidad.

La idea del algoritmo es,

1. C_f depende de f. Queremos identificar C_f y encerrarlo en un rectángulo, a ser posible simétrico.
2. Generar puntos (u,v) dentro del rectángulo uniformemente y comprobar si pertenecen a C_f, si $(u,v) \in C_f \Rightarrow X = V/U$.

La dificultad estriba en determinar C_f, una vez hecho esto el resto es un caso particular del algoritmo de aceptación y rechazo.

La forma de C_f cuando $f \sim N(0,1)$

Si $X \sim N(0,1)$,

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \qquad x \in \mathbb{R}$$

y

$$C_f = \{(u,v) : 0 \leq u \leq \sqrt{f(v/u)}\}$$

donde,

$$\sqrt{f(v/u)} = \sqrt{\frac{1}{\sqrt{2\pi}} e^{-\frac{(\frac{v}{u})^2}{4}}} = \frac{1}{\sqrt[4]{2\pi}} e^{-\frac{(\frac{v}{u})^2}{4}}$$

llamo, $k = \frac{1}{\sqrt[4]{2\pi}}$.

De esta forma podemos caracterizar C_f,

$$C_f = \left\{(u,v) : 0 < u < \frac{1}{k} e^{-\frac{(\frac{v}{u})^2}{4}}\right\}$$

por lo tanto, si $(u,v) \in C_f \Rightarrow x = v/u$.

Límites del rectángulo

$$0 < u < \tfrac{1}{k}e^{-\frac{(\frac{v}{u})^2}{4}} \Rightarrow k \cdot u < e^{-\frac{(\frac{v}{u})^2}{4}} \quad \text{tomando logaritmos}$$

$$\log(k \cdot u) < -(\tfrac{v}{u})^2 \cdot \tfrac{1}{4} \quad \text{pero } u > 0$$

$$4u^2 \log(k \cdot u) < -v^2$$

luego la propiedad de pertenencia se puede escribir,

$$(u, v) \in C_f \text{ si } 4u^2 \log(k \cdot u) < -v^2; \quad u > 0$$

Si $u > \frac{1}{k} \Rightarrow k \cdot u > 1 \Rightarrow \log(k \cdot u) > 0 \Rightarrow (u, v) \notin C_f$

por tanto, si $(u, v) \in C_f$ debe cumplirse que $0 < u < \frac{1}{k}$

Por otra parte,

$$V^2 < \underbrace{-4u^2 \log(k \cdot u)}_{acotado}; \qquad V^2 > 0$$

y sabemos que $0 < u < \frac{1}{k}$.

Si $u = \frac{1}{k} \Rightarrow \log(k \cdot u) = \log(k \cdot \frac{1}{k}) = \log(1) = 0$

Si $u = 0 \to -4u^2 \log(k \cdot u) \to 0$ cuando $u \to 0$, ya que $u^2 \to 0$ más rápido que $\log(k \cdot u) \to -\infty$, cuando $u \to 0$.

Llamando $g(u) = -4u^2 \log(k \cdot u)$ y comprobamos si g tiene algún máximo en $(0, \frac{1}{k})$.

$g'(u) = -4[2u \log(k \cdot u) + u^2 \cdot \frac{1}{k \cdot u} \cdot k] = -4u[2 \log(k \cdot u) + 1]$

Tenemos que

$$g(u) = 0 \Leftrightarrow 2 \log(k \cdot u) + 1 = 0 \Leftrightarrow u = \frac{1}{k}e^{-1/2}$$

luego,

$$\max_{0<u<1/k} g(u) = g\left(\frac{1}{k}e^{-1/2}\right) = -4\frac{1}{k^2}e^{-1} \log(e^{-1/2}) = \frac{2}{e \cdot k}$$

concluimos que, $V^2 < \frac{2}{e \cdot k}$, es decir,

$$-\sqrt{\frac{2}{e \cdot k}} < V < \sqrt{\frac{2}{e \cdot k}}$$

llamando $b = \sqrt{\frac{2}{e \cdot k}}$, tenemos $-b < V < b$.

El algoritmo queda,

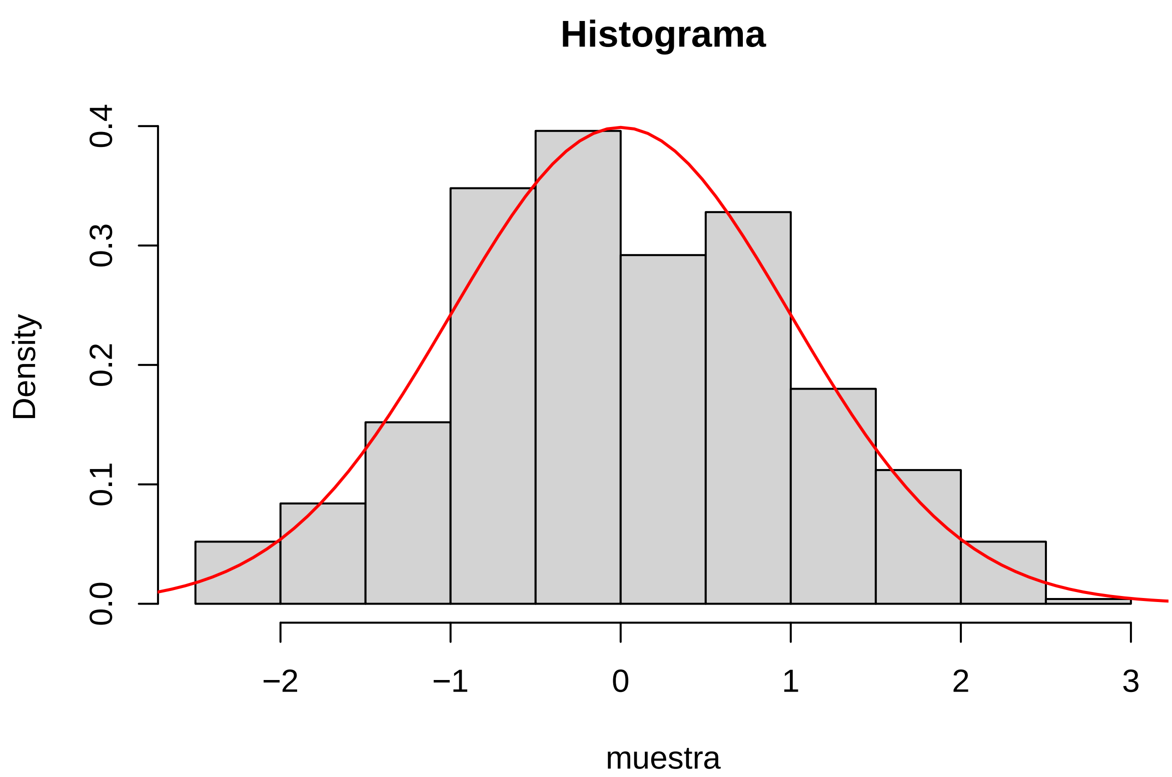
Algoritmo

1.- Generar $U_1, U_2 \sim U(0, 1)$
2.- Hacer $Z = (2e^{-1/2}) \cdot \frac{U_2 - 0.5}{U_1}$
3.- Si $Z^2 < -4 \log(U_1)$, devolver Z
4.- OtroSi ir al 1

En R,

```
constantek <- 4 *exp(-0.5)/sqrt(2)

generar_normal_uni <- function(mu, sigma){
  repeat{
    U1 <- runif(1)
```

```r
    U2 <- 1- runif(1)

    Z  <- constantek * (U1 - 0.5)/U2

    ZZ <- Z^2 / 4
    if(ZZ <= - log(U2)){
      break
    }
  }
  return( mu + Z * sigma)
}

# Muestra
Nsim <- 500

muestra <- replicate(Nsim, generar_normal_uni(0,1))

# Gráficos
hist(muestra, freq = FALSE, main = "Histograma")
curve(dnorm, from = -4, to = 4, col = "red", lwd = 1.5, add = TRUE)
```




```r
# Contraste de hipótesis
ks.test(muestra, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
```

```
##
## data:  muestra
## D = 0.03216, p-value = 0.6792
## alternative hypothesis: two-sided
```

2 Normales multidimensionales