Simulación

Inferencia Estadística

17 de octubre de 2021

Simulación

- Útil para aproximar la distribución exacta de un estadígrafo.
- Se obtienen muchos valores aleatorios del estadígrafo.
- Teorema GC garantiza que su ojiva empírica se aproxima uniformemente a la teórica.

Generación de seudoaleatorios $\mathcal{U}(0,1)$

- Paso básico en procedimientos de simulación.
- Implementada en casi todo lenguaje de programación y calculadora.
- R: `runif(n)`; Octave: `rand`; Maxima: `random(1.0)`; Python: `import random; random.random()`; Common Lisp: `(random 1.0)`; Fortran: `call random_number(r)`
- Algoritmo básico: generación por congruencia
  - m módulo, a multiplicador, c incremento, X_0 semilla (por ejemplo $m = 2^{32}$, $a = 1\,664\,525$, $c = 1\,013\,904\,223$, https://es.wikipedia.org/wiki/Generador_lineal_congruencial)
  - secuencia seudoaleatoria

$$X_{i+1} = [(a \cdot X_i + c) \bmod m]$$

  - *seudo* porque puede reproducirse a partir de la semilla (en R, mediante `set.seed`)
- `RNGkind` en R informa o establece el algoritmo.

Método de la ruleta (Montecarlo original) para variables discretas finitas

- Sea X que toma valores $x_1, \ldots, x_k$ con

$$\Pr[X = x_i] = p_i \qquad P_i = \sum_{j=1}^{i} p_i \qquad P_0 = 0$$

- Sea $C_i = [P_{i-1}, P_i)$ luego $C_1, \ldots, C_k$ partición de $[0, 1)$ con

$$\text{amplitud de } C_i = p_i$$

- Algoritmo:
  1. Generar u según una $\mathcal{U}(0, 1)$.
  2. Si $u \in C_i$ tomar x_i como valor aleatorio de X.
- En R se usan habitualmente `rbinom`, `rpois`, `rgeom`, ...

Método de la ruleta (ejemplo)

- Sea X discreta que toma valores $\{1, 1'5, 3\}$ con probabilidades respectivas $\left\{\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right\}$.
- Objetivo: generar muestra de tamaño $n = 5$ de X.

1. Partir el intervalo $[0, 1]$ en $C_1 = \left[0, \frac{1}{2}\right]$, $C_2 = \left(\frac{1}{2}, \frac{5}{6}\right]$ y $C_3 = \left(\frac{5}{6}, 1\right]$.
2. Generar 5 valores de una $\mathcal{U}(0, 1)$, por ejemplo $(0'08417384, 0'62186518, 0'75045832, 0'76283675, 0'29542565)$.
3. Clasificar los valores generados: $(C_1, C_2, C_2, C_2, C_1)$.
4. Trasformar las clases en valores de X: $(1, 1'5, 1'5, 1'5, 1)$.

- En R:

```
U <- runif (5)
C <- cut (U, c(0, 1/2, 5/6, 1))
X <- c(1,1.5,3)[C]
```

o simplemente (`TRUE` significa *con reposición*):

```
X <- sample (c(1,1.5,3), 5, TRUE, c(1/2,1/3,1/6))
```

Método de la transformación inversa

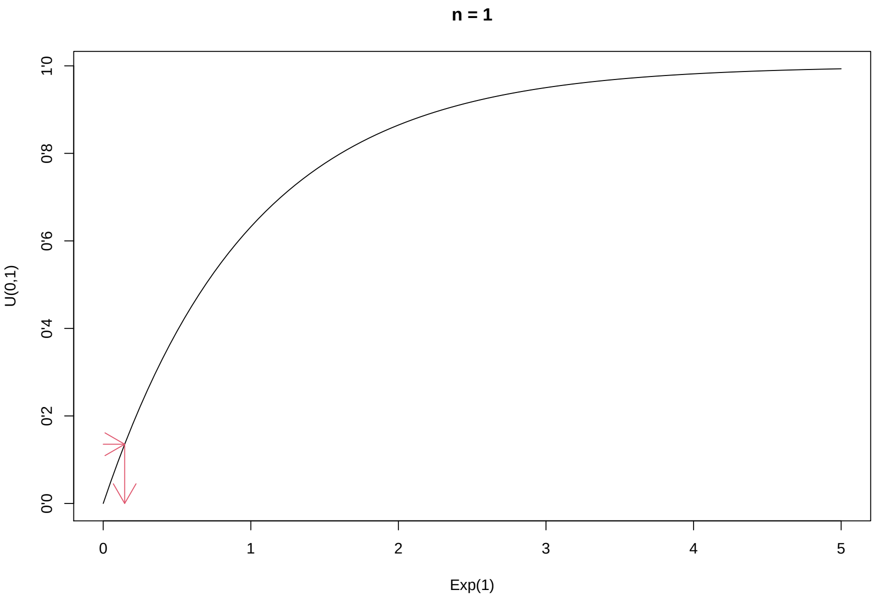
- Sea X continua con ojiva F.
- El método se basa en que $F(x) \sim \mathcal{U}(0,1)$.
- Algoritmo:
  1. Generar u según una $\mathcal{U}(0,1)$.
  2. Tomar $x = F^{-1}(u)$ como valor aleatorio de X.

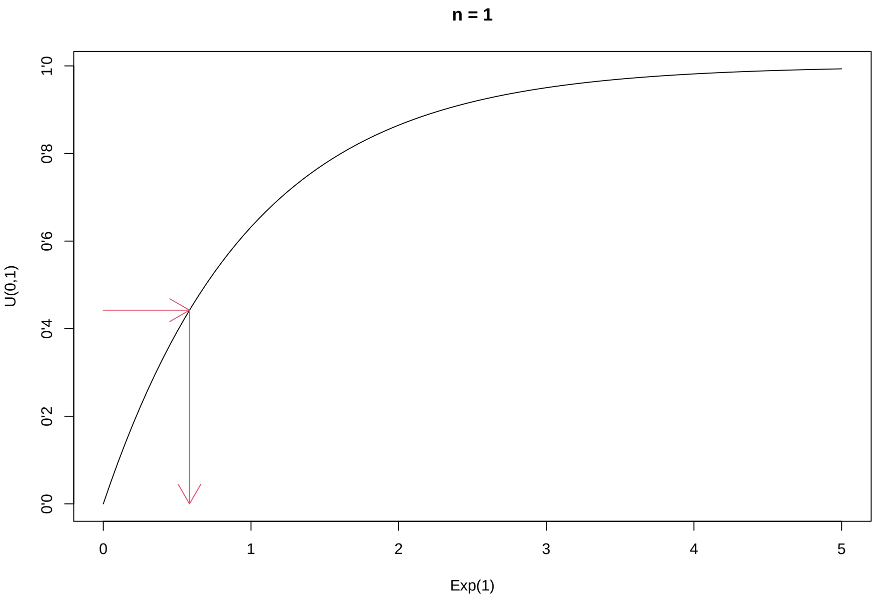
Ejemplo

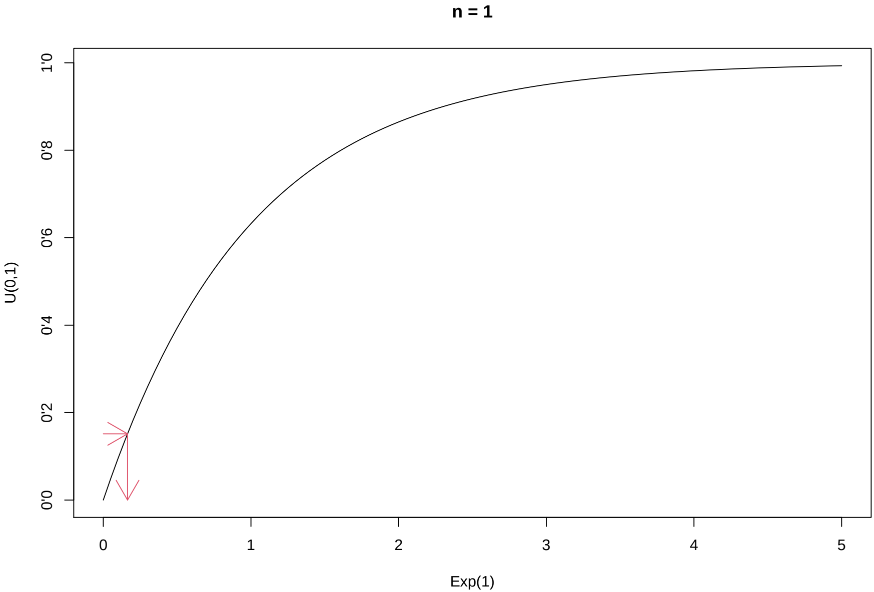
$$X \sim \mathrm{Exp}(\lambda) \quad \Longrightarrow \quad F(x) = 1 - e^{-\lambda x} \quad \text{con } x > 0$$
$$\Longrightarrow \quad F^{-1}(u) = \frac{-\ln(1-u)}{\lambda} \quad \text{con } 0 < u < 1$$

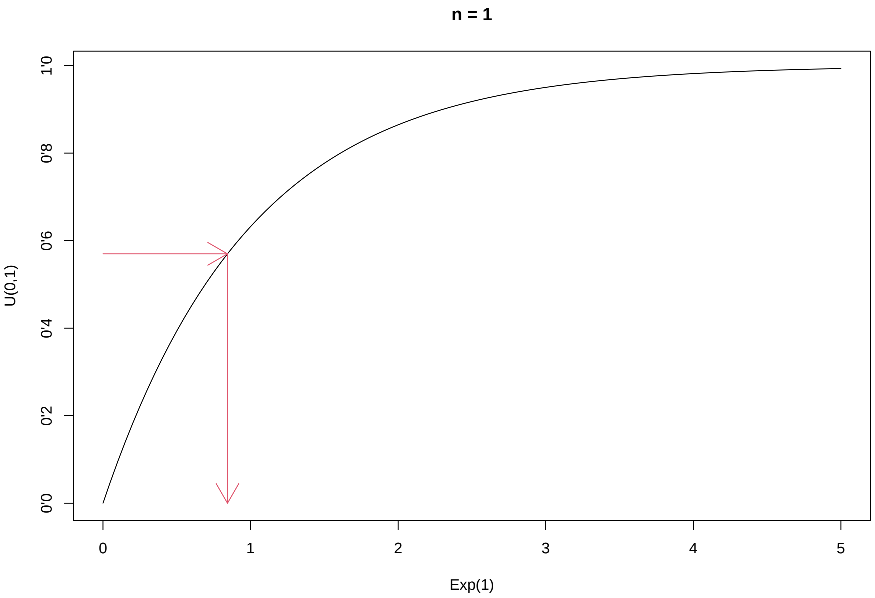
- En R, se usan habitualmente `rexp`, `rnorm`, `rgamma`, `rbeta`, `rchisq`, `rt`, `rf`.

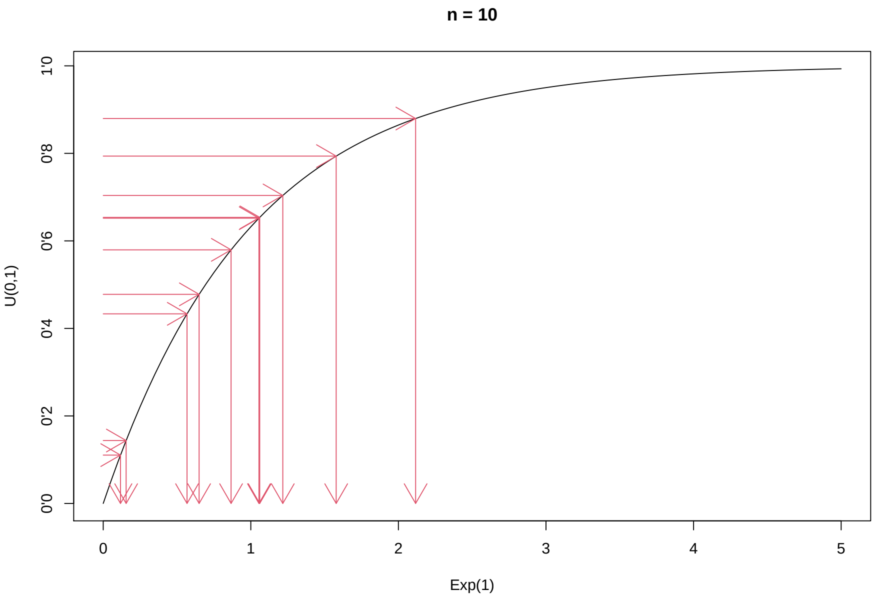
```
options (OutDec="’")
graf <- function (n)
{
    plot (pexp, 0, 5, xlab="Exp(1)", ylab="U(0,1)", main=paste("n =",n))
    X <- c ()
    for (i in 1:n)
    {
        u <- runif (1)
        x <- qexp (u)
        arrows (c(0,x), c(u,u), c(x,x), c(u,0), col=2)
        X <- c (X, x)
    }
    if (n>100)
    {
        d <- hist (X, breaks = sqrt(n), plot=FALSE)
        lines (d$mids, d$density, col=3, lwd=3)
    }
}
pdf ("graf/montecarlo%1d.pdf", onefile=FALSE, width=10)
graf(1); graf(1); graf(1); graf(1)
graf(10); graf(10); graf(10)
graf(100)
graf(1000)
dev.off ()
```

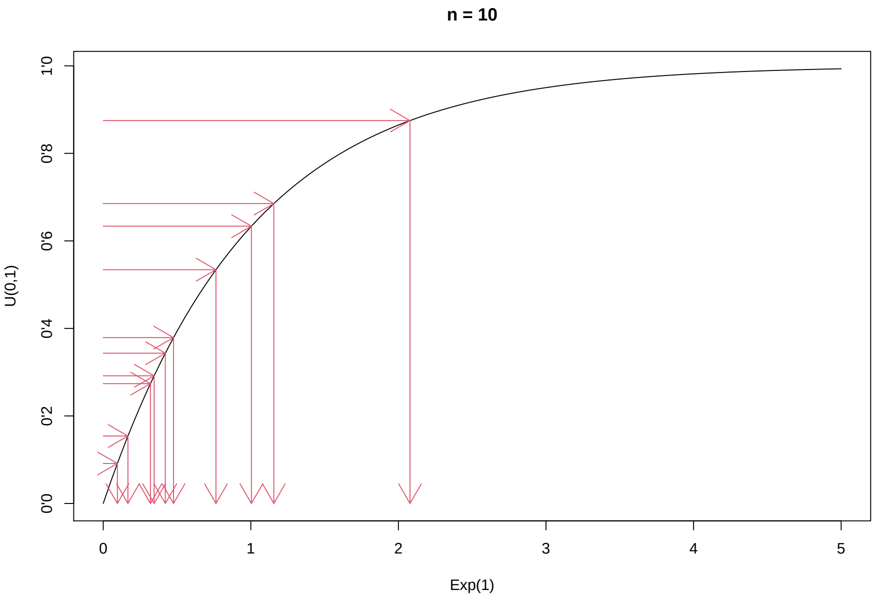

n = 1
U(0,1)
0'0
0'2
0'4
0'6
0'8
1'0
0
1
2
3
4
5
Exp(1)

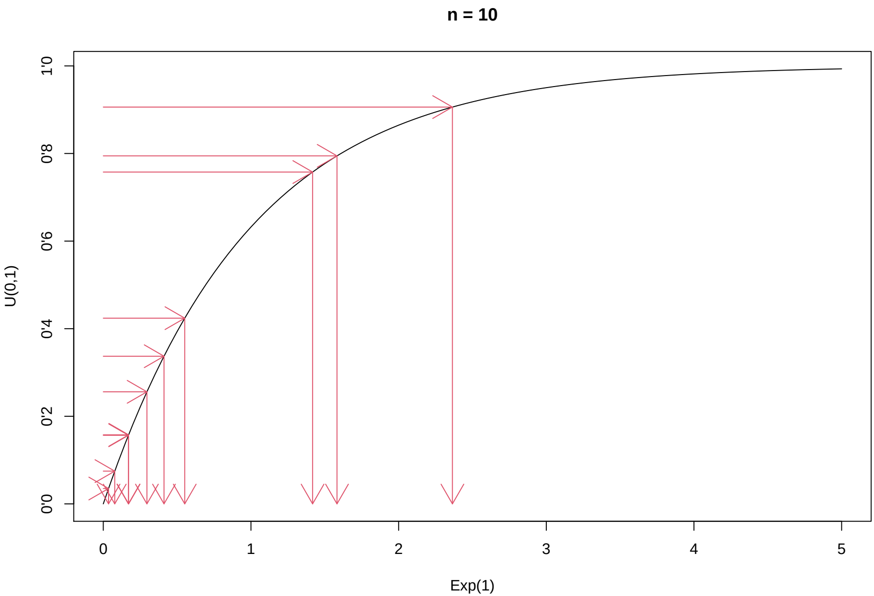

n = 1
1'0
0'8
0'6
0'4
0'2
0'0
U(0,1)
0
1
2
3
4
5
Exp(1)

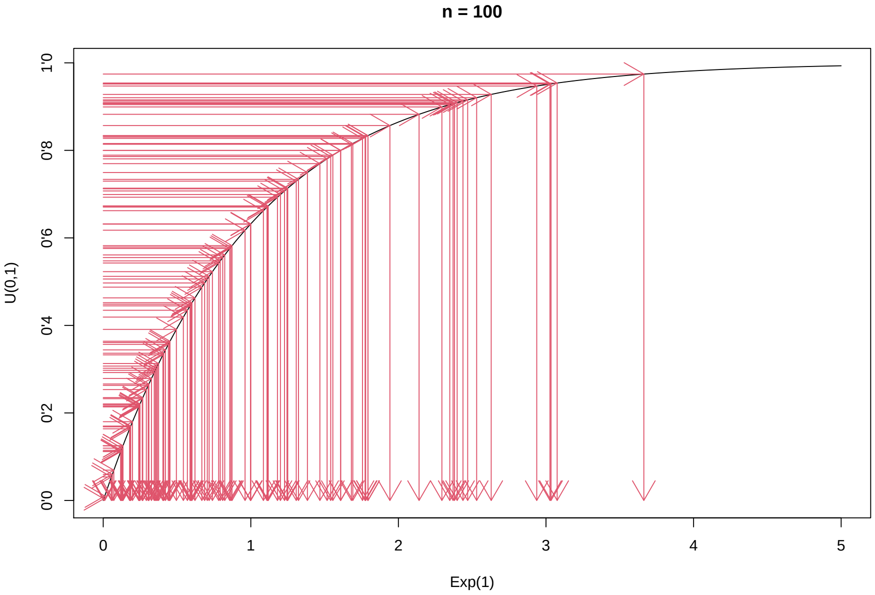

n = 1
1'0
0'8
0'6
0'4
0'2
0'0
U(0,1)
0
1
2
3
4
5
Exp(1)

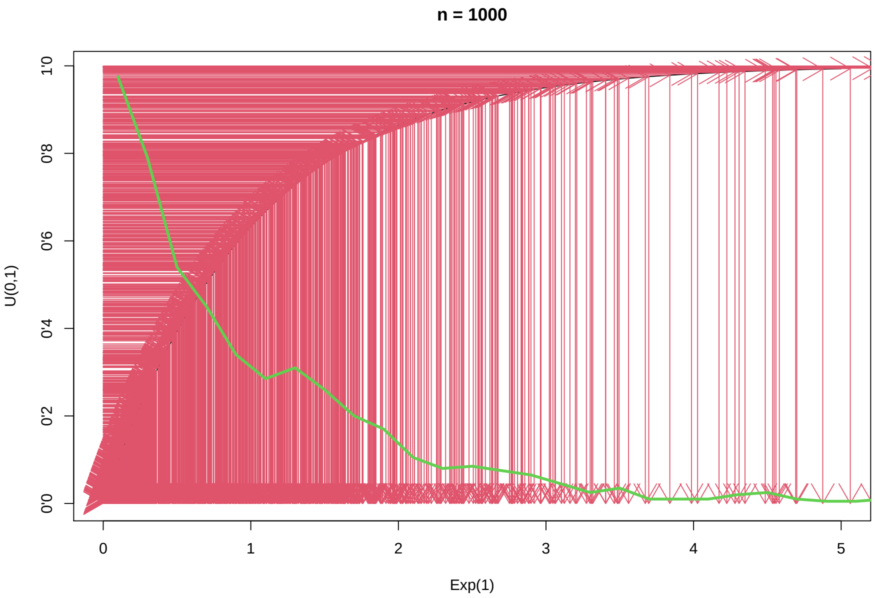

n = 1
1'0
0'8
0'6
0'4
0'2
0'0
U(0,1)
0
1
2
3
4
5
Exp(1)

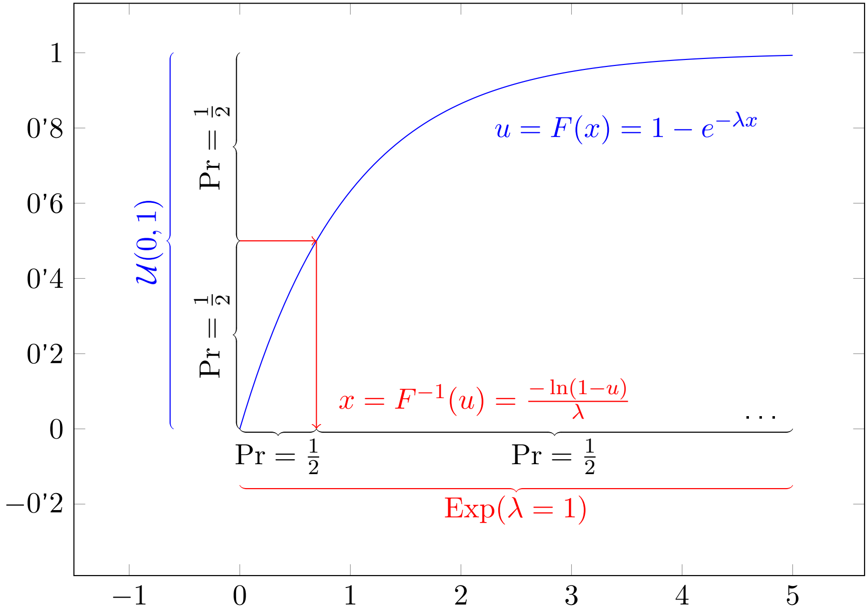

n = 10
U(0,1)
0'0
0'2
0'4
0'6
0'8
1'0
0
1
2
3
4
5
Exp(1)


n = 10
U(0,1)
0'0
0'2
0'4
0'6
0'8
1'0
0
1
2
3
4
5
Exp(1)


n = 10
1'0
0'8
0'6
0'4
0'2
0'0
U(0,1)
0
1
2
3
4
5
Exp(1)


n = 100
U(0,1)
0'0
0'2
0'4
0'6
0'8
1'0
0
1
2
3
4
5
Exp(1)


n = 1000
1'0
0'8
0'6
0'4
0'2
0'0
U(0,1)
0
1
2
3
4
5
Exp(1)


$u = F(x) = 1 - e^{-\lambda x}$
$x = F^{-1}(u) = \frac{-\ln(1-u)}{\lambda}$
$\mathcal{U}(0,1)$
$\Pr = \frac{1}{2}$
$\Pr = \frac{1}{2}$
$\Pr = \frac{1}{2}$
$\Pr = \frac{1}{2}$
$\mathrm{Exp}(\lambda = 1)$
...
1
0'8
0'6
0'4
0'2
0
−0'2
−1
0
1
2
3
4
5

Método de Box y Muller

- Sean U_1 y U_2 dos variables $\mathcal{U}(0,1)$ independientes.
- Sean

$$\begin{aligned} X &= \sqrt{-2\ln U_1}\cos(2\pi U_2) \\ Y &= \sqrt{-2\ln U_1}\,\mathrm{sen}(2\pi U_2) \end{aligned}$$

- Entonces X y Y son $\mathcal{N}(0,1)$ independientes.
- Disponible en R mediante `RNGkind(normal.kind="Box-Muller")` pero por omisión se usa la transformación inversa $\Phi^{-1} =$ `qnorm`.

Método de Box y Muller (ejemplo)

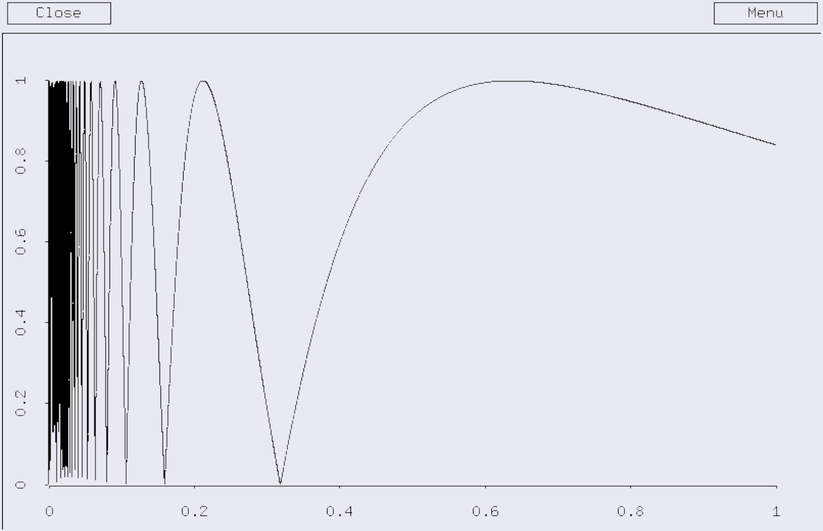
- Objetivo: Generar una muestra de tamaño n de valores aleatorios de una distribución gausiana multivariante $\vec{X} \sim \mathcal{N}_4(\vec{\mu}, \Sigma)$.
- Iterar n veces:
  1. Sean u_1, u_2, u_3 y u_4 valores $\mathcal{U}(0,1)$ independientes.
  2. Sean

$$y_1 = \sqrt{-2\ln u_1}\cos(2\pi u_2) \qquad y_3 = \sqrt{-2\ln u_3}\cos(2\pi u_4)$$
$$y_2 = \sqrt{-2\ln u_1}\operatorname{sen}(2\pi u_2) \qquad y_4 = \sqrt{-2\ln u_3}\operatorname{sen}(2\pi u_4)$$

  3. Box-Muller $\implies (y_1, y_2) \sim (y_3, y_4) \sim \mathcal{N}_2\left(\vec{0}, I\right)$
  4. u_1, u_2, u_3, u_4 indep. $\implies \vec{y} = (y_1, y_2, y_3, y_4)'$ indep. y $\vec{y} \sim \mathcal{N}_4\left(\vec{0}, I\right)$
  5. Σ matriz cov. $\implies \Sigma$ def. positiva $\implies \Sigma = U\Lambda U'$, Λ diag. U ortog.
  6. Sea $\vec{x} = \vec{\mu} + U\Lambda^{\frac{1}{2}}\vec{y} \sim \mathcal{N}_4(\vec{\mu}, \Sigma)$ pues $\operatorname{cov}(\vec{x}) = U\Lambda^{\frac{1}{2}}\operatorname{cov}(\vec{y})\Lambda^{\frac{1}{2}}U' = \Sigma$
  7. Entonces $\vec{x}$ es un valor generado según la distribución de $\vec{X}$.
- En R puede usarse: `mvtnorm::rmvnorm(n,mu,Sigma)`

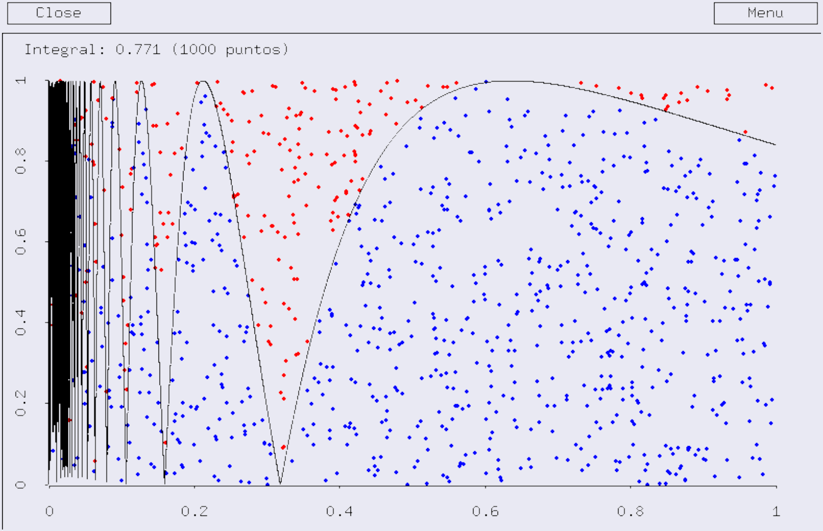
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \mathrm{d}x$



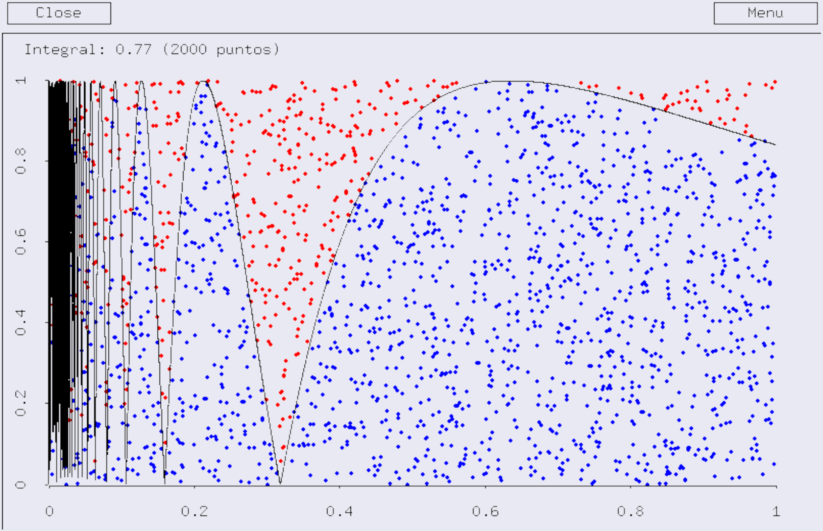
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \mathrm{d}x$



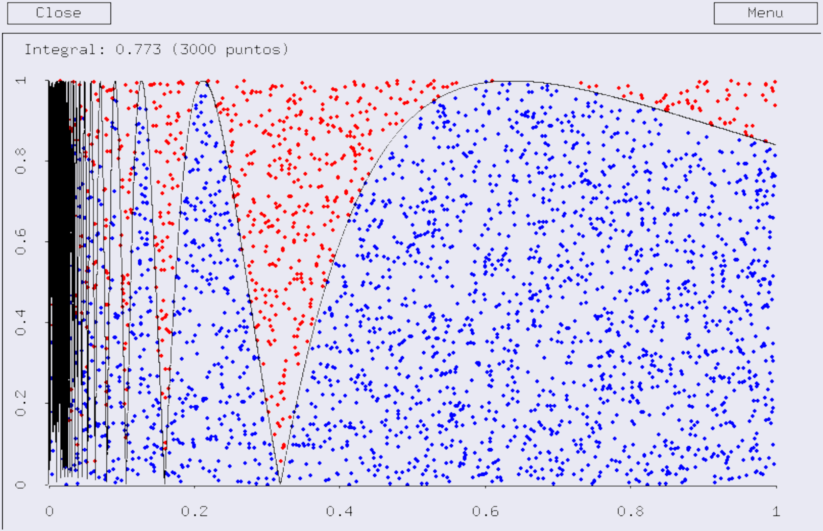
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \mathrm{d}x$



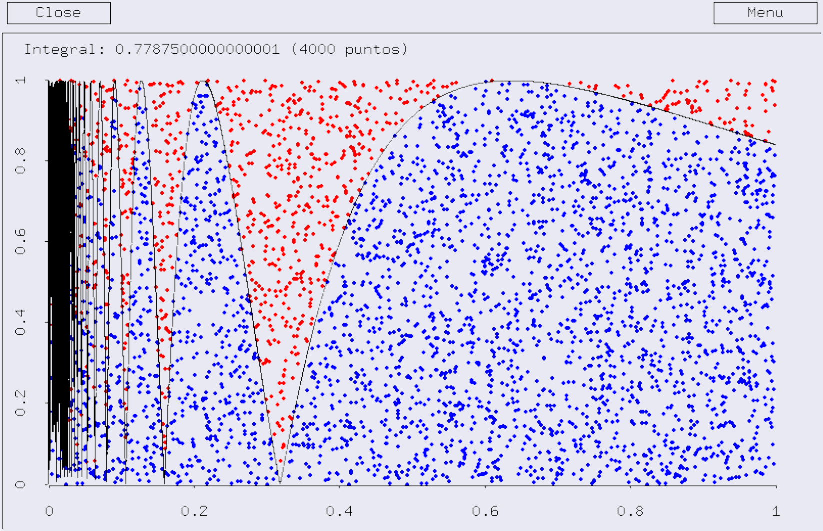
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \, \mathrm{d}x$



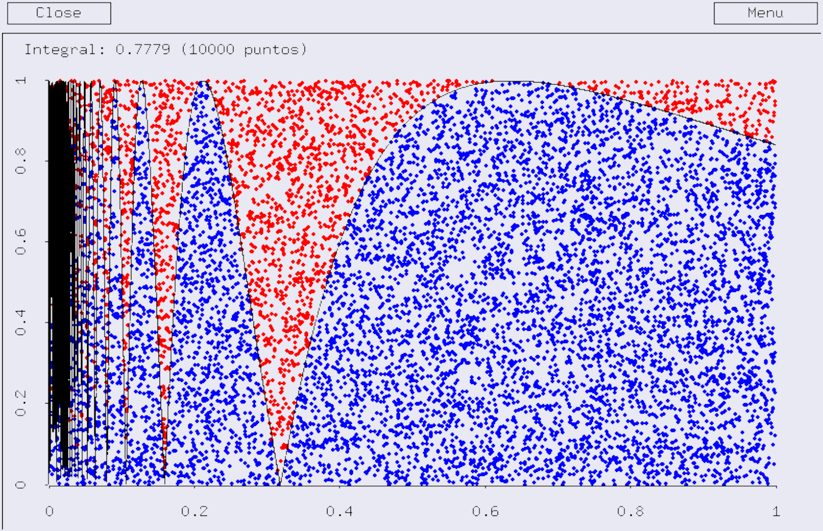
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \,\mathrm{d}x$



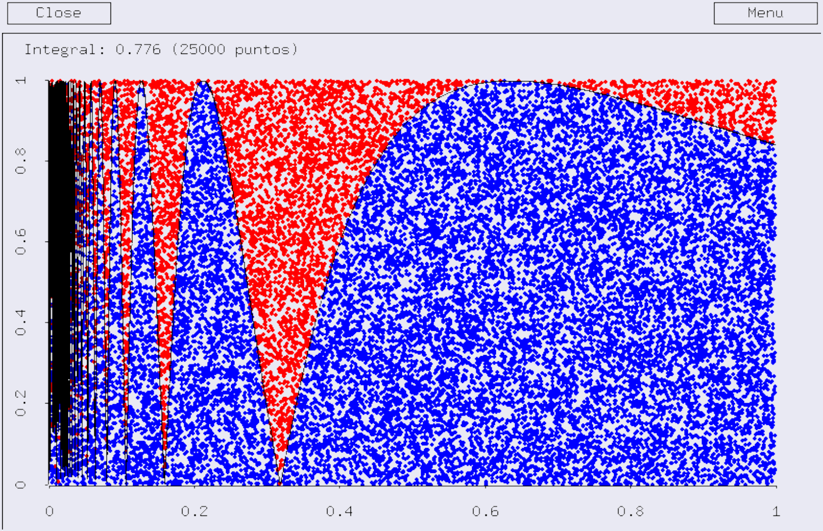
Método de aceptación y rechazo (intro: integración de Montecarlo)

Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \mathrm{d}x$



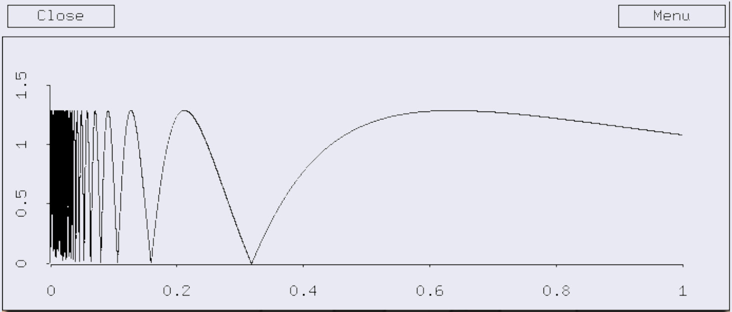
Método de aceptación y rechazo (intro: integración de Montecarlo)

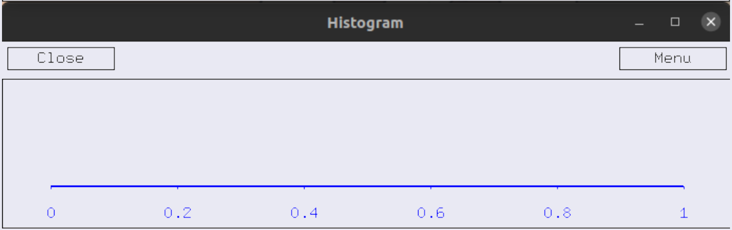
Objetivo: $\int_0^1 \operatorname{sen}\left|\frac{1}{x}\right| \mathrm{d}x$



Método de aceptación y rechazo (intro: simple)

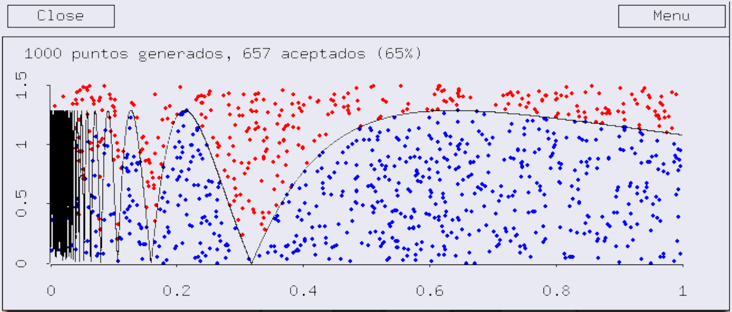
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$

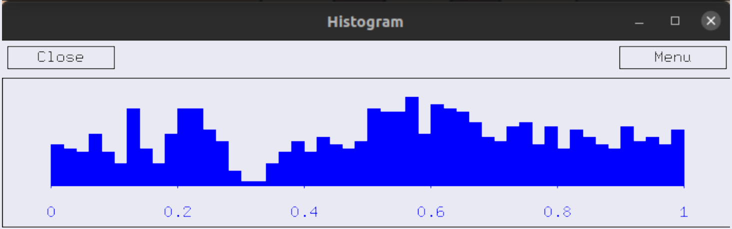




Método de aceptación y rechazo (intro: simple)

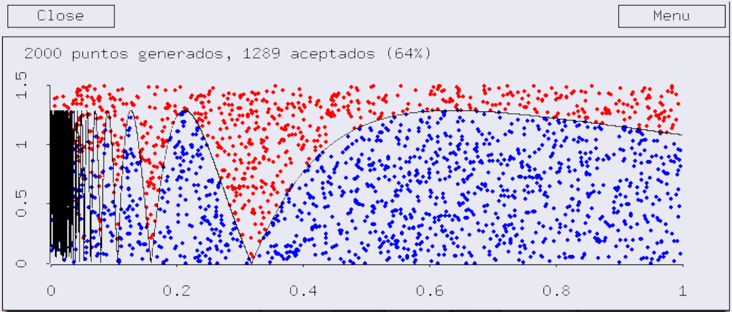
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$

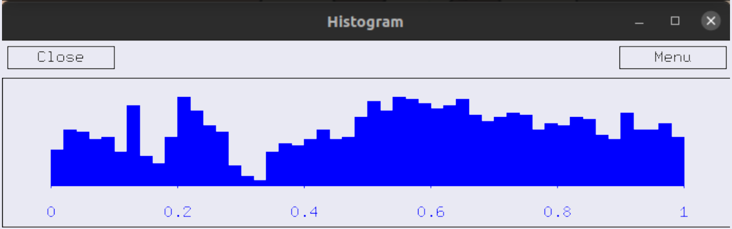




Método de aceptación y rechazo (intro: simple)

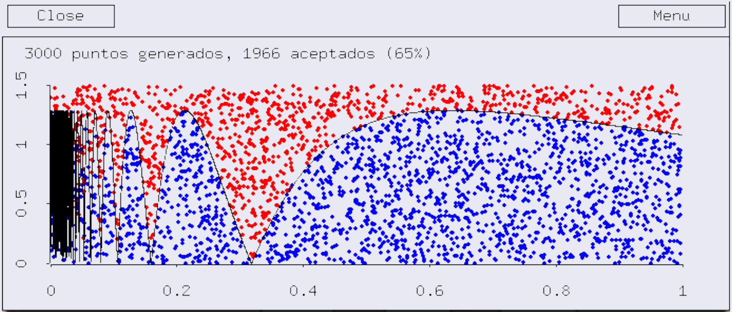
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$

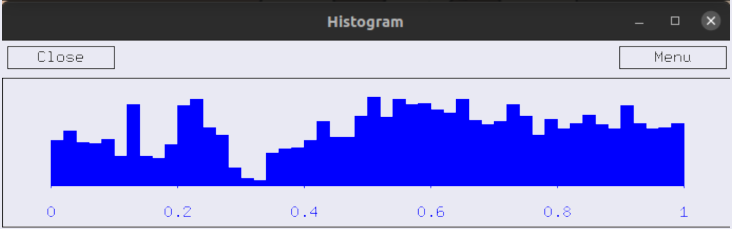




Método de aceptación y rechazo (intro: simple)

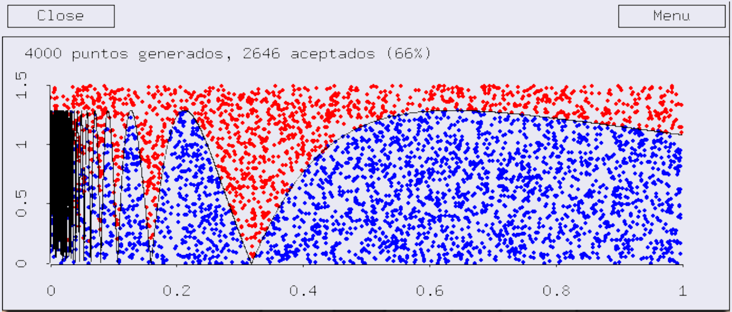
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right|\mathbb{1}_{(0,1)}(x)$

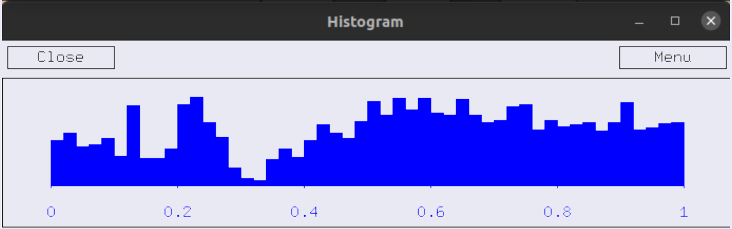




Método de aceptación y rechazo (intro: simple)

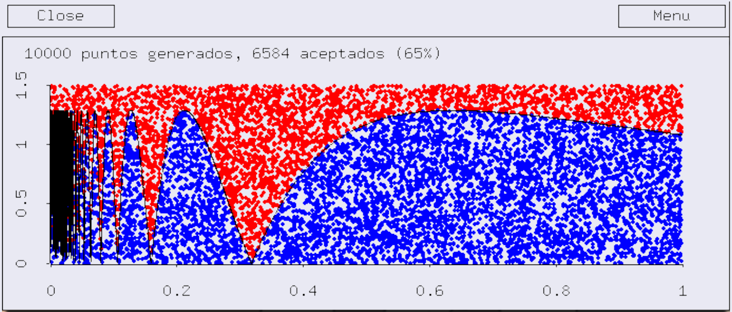
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$

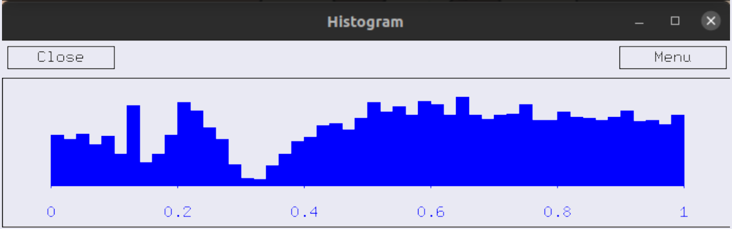




Método de aceptación y rechazo (intro: simple)

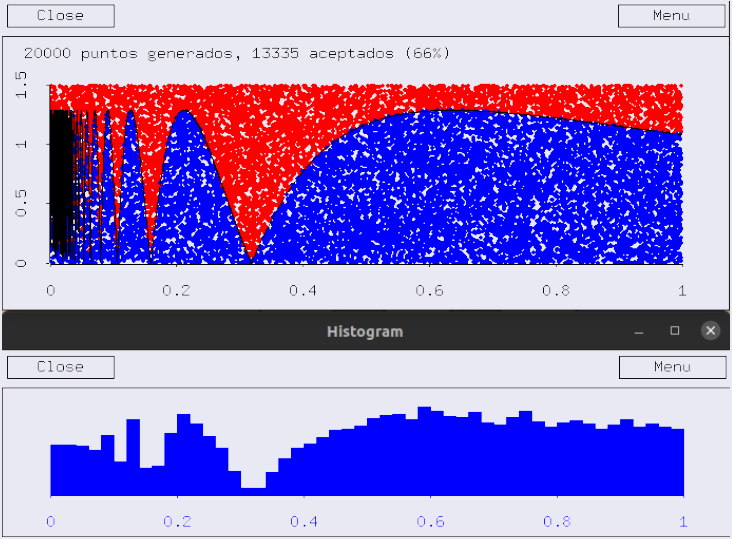
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$





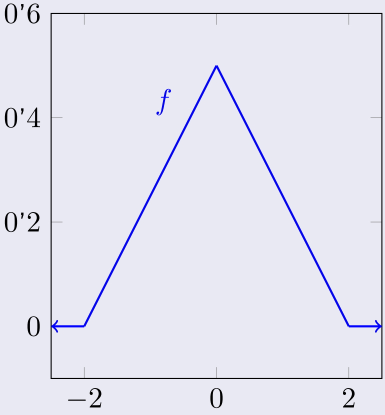
Método de aceptación y rechazo (intro: simple)

Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$



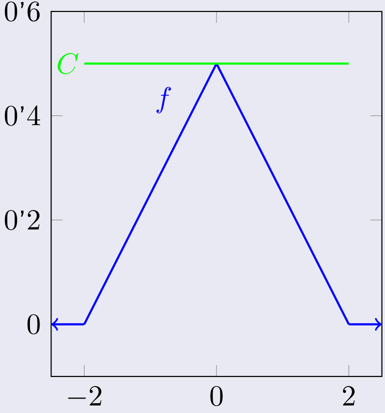
Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.



Método de aceptación y rechazo (simple)

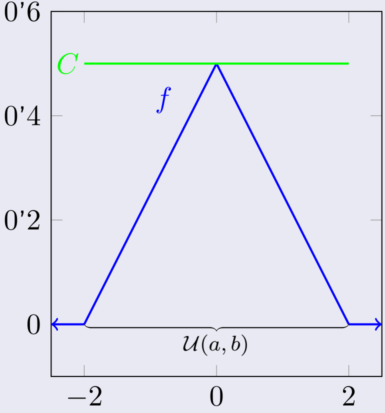
- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.



Método de aceptación y rechazo (simple)

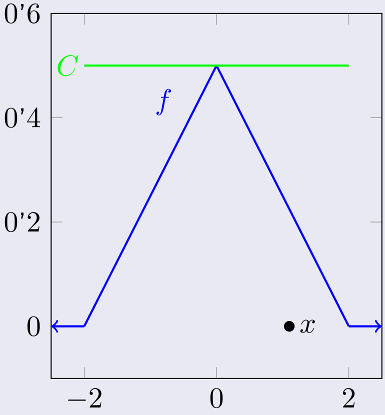
- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:

a partir de $\mathcal{U}(a, b)$.



Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.

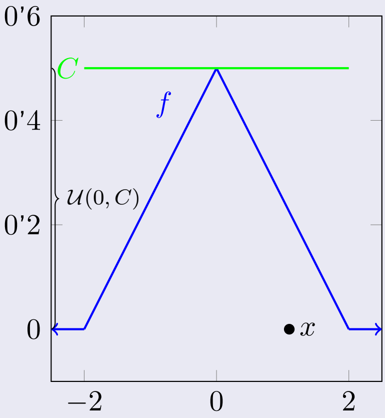


Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a,b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a,b)$.

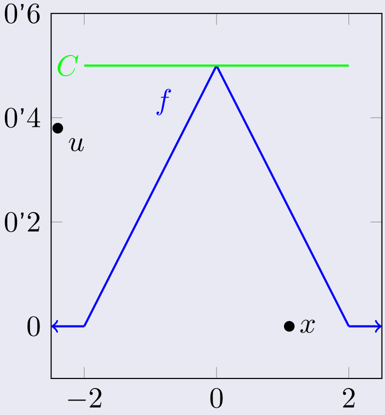
según

$\mathcal{U}(0,C)$.



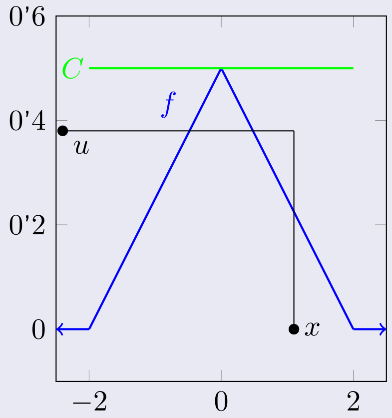
Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.
  2. Sea u generado según $\mathcal{U}(0, C)$.



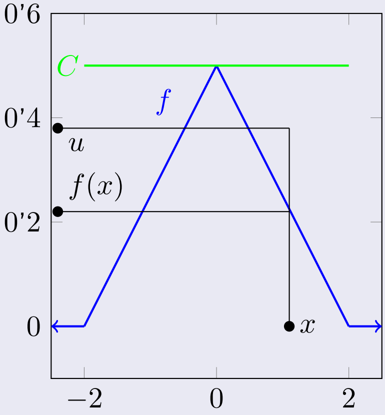
Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.
  2. Sea u generado según $\mathcal{U}(0, C)$.
  3. Si $u \leqslant f(x)$ tomar x como valor de X. Si no, volver a empezar.



Método de aceptación y rechazo (simple)

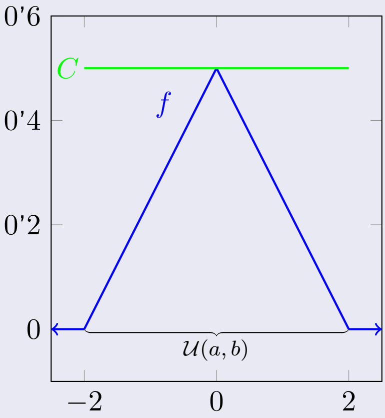
- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.
  2. Sea u generado según $\mathcal{U}(0, C)$.
  3. Si $u \leqslant f(x)$ tomar x como valor de X. Si no, volver a empezar.



Método de aceptación y rechazo (simple)

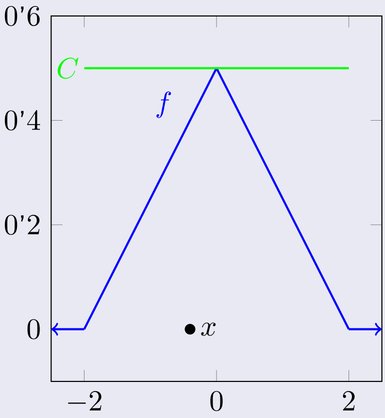
- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:

a partir de $\mathcal{U}(a, b)$.



Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.

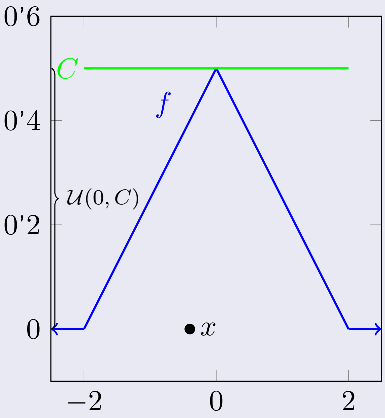


Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.

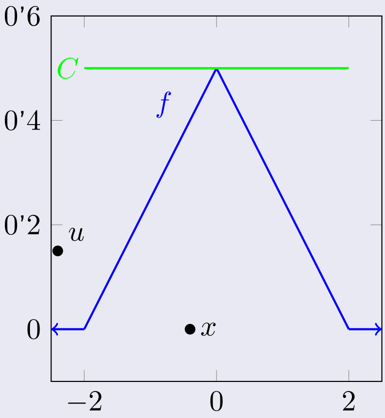
     según

  $\mathcal{U}(0, C)$.



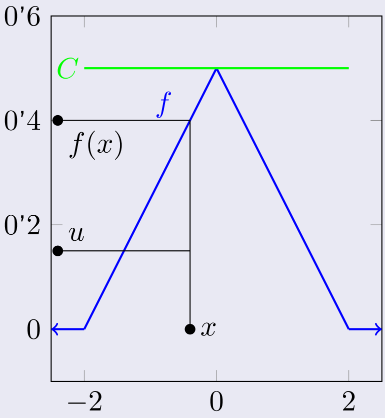
Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.
  2. Sea u generado según $\mathcal{U}(0, C)$.



Método de aceptación y rechazo (simple)

- Sea X continua con densidad f con soporte acotado $[a, b]$.
- Sea C tal que $f \leqslant C$.
- Algoritmo:
  1. Generar x a partir de $\mathcal{U}(a, b)$.
  2. Sea u generado según $\mathcal{U}(0, C)$.
  3. Si $u \leqslant f(x)$ tomar x como valor de X. Si no, volver a empezar.



Método de aceptación y rechazo (simple)

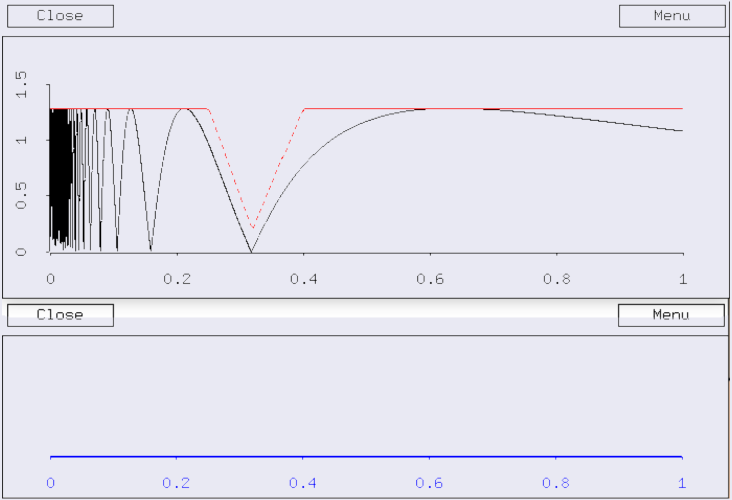
```
f.tri <- function (x, a)      # densidad triangular en [-a,+a]
    1/a^2 * ((a - x) * (x >= 0) * (x < a) +
             (a + x) * (x <  0) * (x > -a))

AR <- function (n, f, a, b, cota) { # método acepta/rechaza
    muestra <- numeric(n)
    i <- 1
    while (i <= n) {
        x <- runif (1, a, b); y <- runif (1, 0, cota)
        if (y <= f(x)) {
            muestra[i] <- x
            i <- i + 1
        }
    }
    muestra
}

muestra <- AR (1e5, function (x) f.tri (x, 2), -2, +2, 1/2)
```

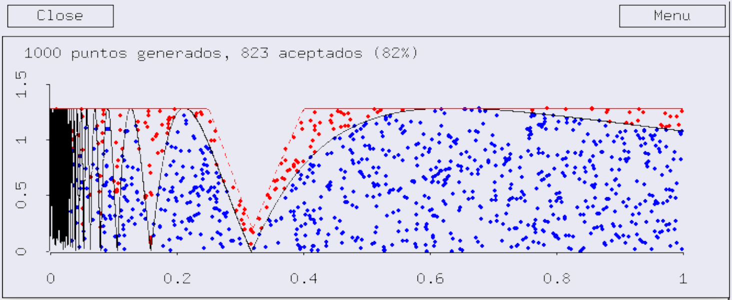
Método de aceptación y rechazo (intro: simple)

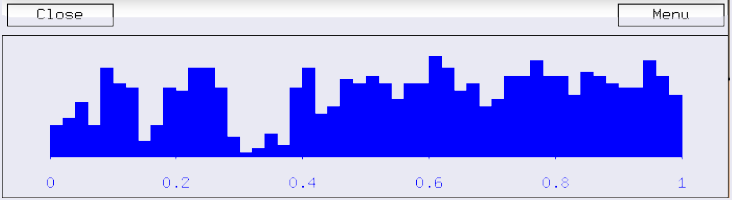
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$



Método de aceptación y rechazo (intro: simple)

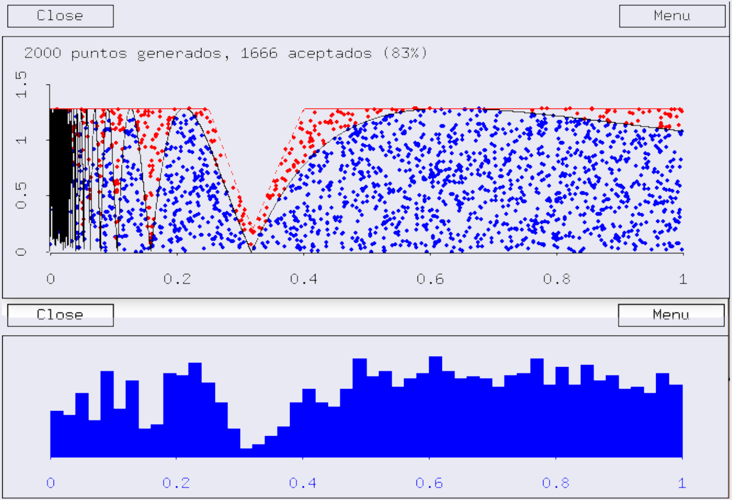
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$





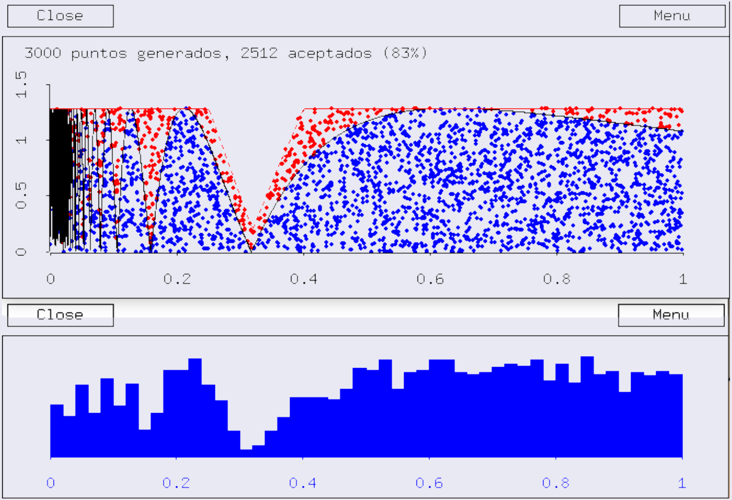
Método de aceptación y rechazo (intro: simple)

Objetivo: generar aleatorios según $f(x) \propto \mathrm{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$



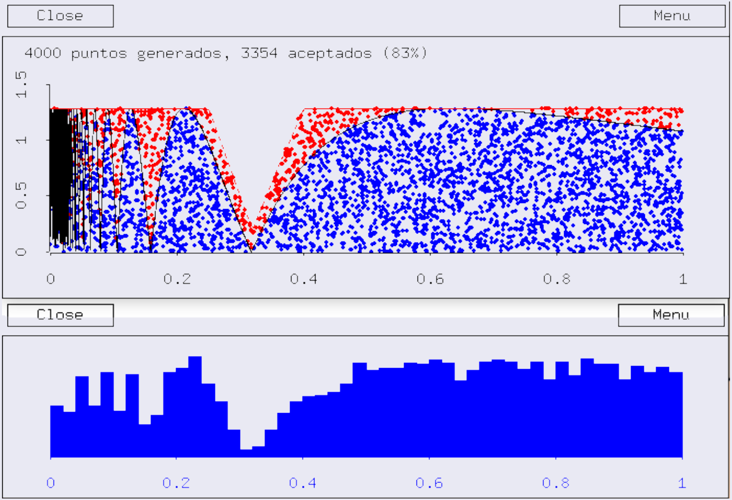
Método de aceptación y rechazo (intro: simple)

Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$



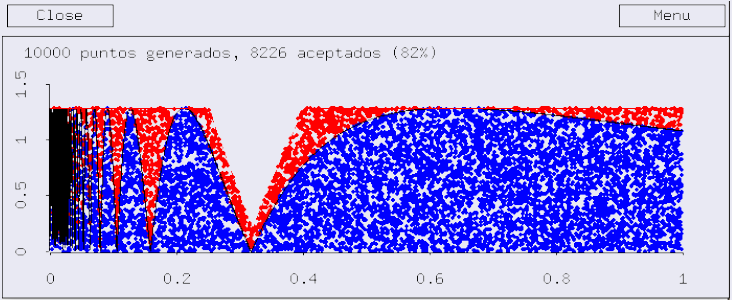
Método de aceptación y rechazo (intro: simple)

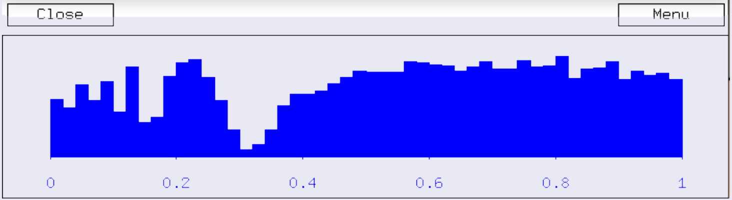
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$



Método de aceptación y rechazo (intro: simple)

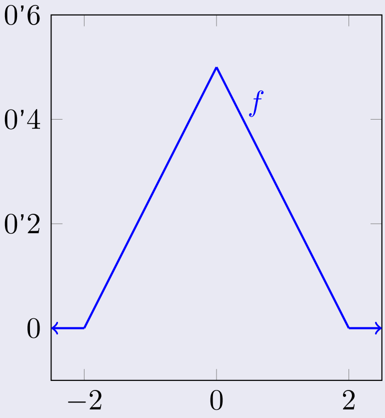
Objetivo: generar aleatorios según $f(x) \propto \operatorname{sen}\left|\frac{1}{x}\right| \mathbb{1}_{(0,1)}(x)$





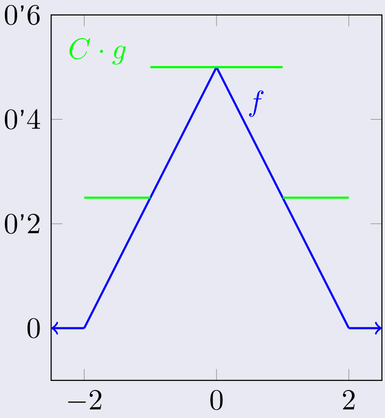
Método de aceptación y rechazo (general)

- Sea X continua con densidad f.
- Sea Y con densidad g, tal que es fácil generar seudoaleatorios de Y y existe C tal que $f \leqslant C \cdot g$.
- Algoritmo:
  1. Generar y a partir de g.
  2. Generar u a partir de $\mathcal{U}(0, C \cdot g(y))$.
  3. Si $u \leqslant f(y)$ tomar y como valor de X. Si no, volver a empezar.



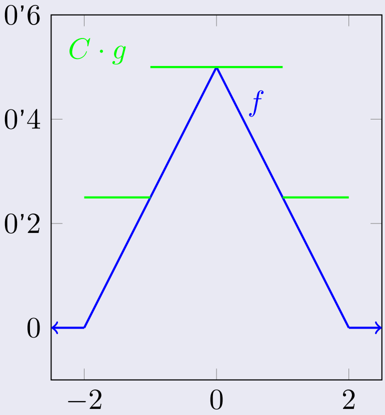
Método de aceptación y rechazo (general)

- Sea X continua con densidad f.
- Sea Y con densidad g, tal que es fácil generar seudoaleatorios de Y y existe C tal que $f \leqslant C \cdot g$.
- Algoritmo:
  1. Generar y a partir de g.
  2. Generar u a partir de $\mathcal{U}(0, C \cdot g(y))$.
  3. Si $u \leqslant f(y)$ tomar y como valor de X. Si no, volver a empezar.



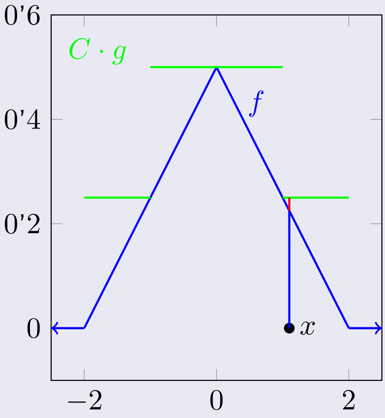
Método de aceptación y rechazo (general)

- Sea X continua con densidad f.
- Sea Y con densidad g, tal que es fácil generar seudoaleatorios de Y y existe C tal que $f \leqslant C \cdot g$.
- Algoritmo:
  1. Generar y a partir de g.
  2. Generar u a partir de $\mathcal{U}(0, C \cdot g(y))$.
  3. Si $u \leqslant f(y)$ tomar y como valor de X. Si no, volver a empezar.



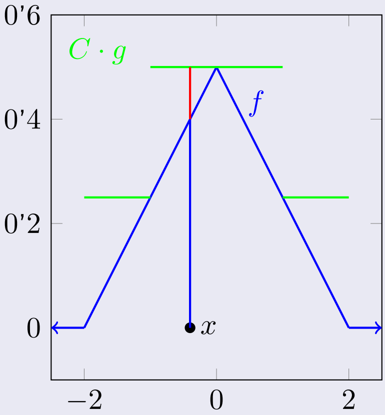
Método de aceptación y rechazo (general)

- Sea X continua con densidad f.
- Sea Y con densidad g, tal que es fácil generar seudoaleatorios de Y y existe C tal que $f \leqslant C \cdot g$.
- Algoritmo:
  1. Generar y a partir de g.
  2. Generar u a partir de $\mathcal{U}(0, C \cdot g(y))$.
  3. Si $u \leqslant f(y)$ tomar y como valor de X. Si no, volver a empezar.



Método de aceptación y rechazo (general)

- Sea X continua con densidad f.
- Sea Y con densidad g, tal que es fácil generar seudoaleatorios de Y y existe C tal que $f \leqslant C \cdot g$.
- Algoritmo:
  1. Generar y a partir de g.
  2. Generar u a partir de $\mathcal{U}(0, C \cdot g(y))$.
  3. Si $u \leqslant f(y)$ tomar y como valor de X. Si no, volver a empezar.

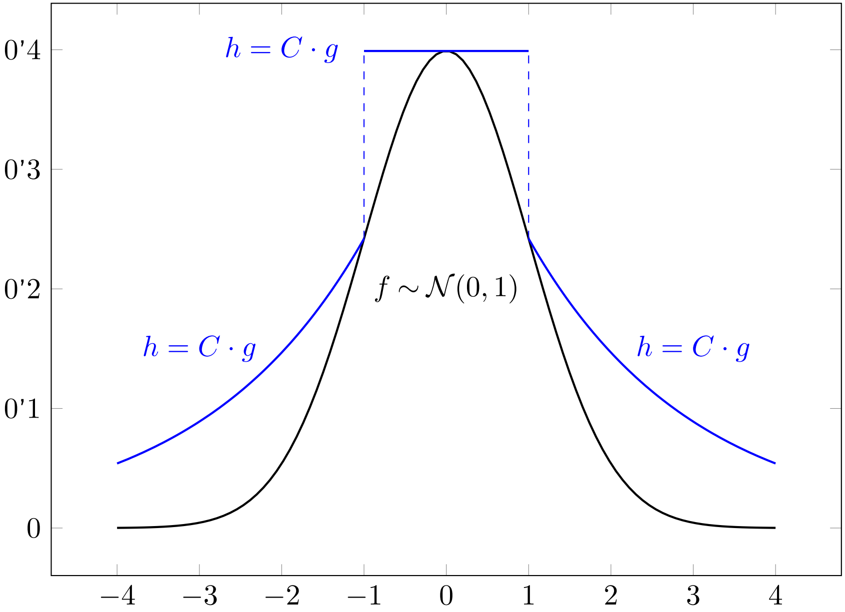


Método de A/R: otro ejemplo

- $X \sim \mathcal{N}(0,1) \implies f_X(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$ $\quad F_X$ no invertible
- f_X campaniforme con máximo $\frac{1}{2\pi}$ en 0
- $x \notin [-1,1] \implies x^2 > |x| \implies f_X(x) < \frac{1}{\sqrt{2\pi}} e^{\frac{-|x|}{2}}$
- $\forall\, x \in \mathbb{R},\ f_X(x) \leqslant h(x)$ con

$$h(x) = \begin{cases} \frac{1}{\sqrt{2\pi}} & x \in [-1,1] \\ \frac{1}{\sqrt{2\pi}} e^{\frac{-|x|}{2}} & x \notin [-1,1] \end{cases}$$

- $\int_{-\infty}^{+\infty} h(x)\, \mathrm{d}x = \sqrt{\frac{2}{\pi}}(2e^{-\frac{1}{2}} + 1) \approx 1'765\,767 = C < \infty$
- Sea Y la variable auxiliar con densidad $g = \frac{h}{C}$


$h = C \cdot g$
$h = C \cdot g$
$h = C \cdot g$
$f \sim \mathcal{N}(0,1)$
0'4
0'3
0'2
0'1
0
−4 −3 −2 −1 0 1 2 3 4

Método de A/R: otro ejemplo

- existe inversa H^{-1} de la acumulada de h, $H(x) = \int_{-\infty}^{x} h$, para aplicar el método de la transformación inversa con $0 < y < C$

$$H^{-1}(y) = \begin{cases} \ln\left(\frac{\pi}{2}y^2\right) & \text{si } 0 < y < \sqrt{\frac{2}{\pi e}} \\ y\sqrt{2\pi} - \frac{2}{\sqrt{e}} - 1 & \text{si } \sqrt{\frac{2}{\pi e}} \leqslant y \leqslant \sqrt{\frac{2}{\pi e}} + \sqrt{\frac{2}{\pi}} \\ -2\ln\left(1 + \frac{2}{\sqrt{e}} - y\sqrt{\frac{\pi}{2}}\right) & \text{si } \sqrt{\frac{2}{\pi e}} + \sqrt{\frac{2}{\pi}} < y < \left(\frac{2}{\sqrt{e}} + 1\right)\sqrt{\frac{2}{\pi}} \end{cases}$$

- alternativamente, existe inversa G^{-1} de la acumulada de g, $G(x) = \int_{-\infty}^{x} g$, para aplicar el mismo método con $0 < y < 1$

$$G^{-1}(y) = \begin{cases} 2\ln[y(2+\sqrt{e})] - 1 & \text{si } 0 < y < \frac{1}{2+\sqrt{e}} \\ 2y - 1 + \frac{4y-2}{\sqrt{e}} & \text{si } \frac{1}{2+\sqrt{e}} \leqslant y \leqslant \frac{\sqrt{e}+1}{\sqrt{e}+2} \\ -2\ln\left(\frac{2-2y}{\sqrt{e}} - y + 1\right) & \text{si } \frac{\sqrt{e}+1}{\sqrt{e}+2} < y < 1 \end{cases}$$

Obtención de G^{-1} con Maxima (Ginversa.mac)

```
f(x)  := 1/sqrt(2*%pi) * %e^(-x^2/2) $ /* f < h  */

h1(x) := 1/sqrt(2*%pi) * %e^(x/2)     $ /* x < -1 */
h2(x) := 1/sqrt(2*%pi)                $ /* -1<x<1 */
h3(x) := 1/sqrt(2*%pi) * %e^(-x/2)    $ /* x < -1 */

h(x)  := if x<-1 then g1(x)
         elseif x<=1 then g2(x) else g3(x) $

/* plot2d (h(x), [x,-3,+3]) $ */
```

Obtención de G^{-1} con Maxima (Ginversa.mac)

```
H1(x) := integrate (h1(t), t, -inf, x) $
H2(x) := H1(-1) + integrate (h2(t), t, -1, x) $
H3(x) := H2(+1) + integrate (h3(t), t, +1, x) $
C     :  H3(inf) $ /* aprox. 1,765767458879439 */
G1(x) := H1(x)/C $ G2(x) := H2(x)/C $ G3(x) := H3(x)/C
G(x)  := if x<-1 then G1(x)
         elseif x<=1 then G2(x) else G3(x) $
plot2d (G(x), [x,-5,+5]) $

G_1(y)  := rhs (solve (G1(x)=y, x) [1]) $ /* G1 inversa */
G_2(y)  := rhs (solve (G2(x)=y, x) [1]) $
G_3(y)  := rhs (solve (G3(x)=y, x) [1]) $
G_(y)   := if y<G(-1) then G_1(y)
           elseif y<=G(+1) then G_2(y) else G_3(y) $
Ginv(y) := G_(y) $ /* inversa definida en [0;1] */
plot2d (Ginv(y), [y,0.001,0.999]) $
```

Obtención de G^{-1} con Maxima (Ginversa.mac)

Sesión normal con display2d:true

```
(%i32) G(-1), ratsimp;
                                    sqrt(%e)
(%o32)                           ---------------
                                 %e + 2 sqrt(%e)
(%i33) G(+1), ratsimp;
                                  %e + sqrt(%e)
(%o33)                           ---------------
                                 %e + 2 sqrt(%e)
(%i34) G_1(y), ratsimp;
(%o34)                     2 log(sqrt(%e) y + 2 y) - 1
(%i35) G_2(y), ratsimp;
                          sqrt(%e) (2 y - 1) + 4 y - 2
(%o35)                    ----------------------------
                                    sqrt(%e)
(%i36) G_3(y), ratsimp;
                                        sqrt(%e)
(%o36)                 2 log(- -------------------------)
                               2 y + sqrt(%e) (y - 1) - 2
```

Obtención de G^{-1} con Maxima (Ginversa.mac)

Sesión con displa2d:false para usar las expresiones con R

```
(%i55) display2d : false $
(%i56) G(-1), ratsimp ;
(%o56) sqrt(%e)/(%e+2*sqrt(%e))
(%i57) G(1), ratsimp ;
(%o57) (%e+sqrt(%e))/(%e+2*sqrt(%e))
(%i58) G_1(y), ratsimp ;
(%o58) 2*log(sqrt(%e)*y+2*y)-1
(%i59) G_2(y), ratsimp ;
(%o59) (sqrt(%e)*(2*y-1)+4*y-2)/sqrt(%e)
(%i60) G_3(y), ratsimp ;
(%o60) 2*log(-sqrt(%e)/(2*y+sqrt(%e)*(y-1)-2))
```

Para evitar sustituir las %e se puede usar fortran :

```
 (%i61) fortran (ratsimp (G_3(y))) $
2*log(-exp(1.0E+0/2.0E+0)/(2*y+exp(1.0E+0/2.0E+0)*(y-1)-2))
```

Obtención de G^{-1} con Maxima (`Ginversa.mac`)

Sesión con `tex` para la presentación en PDFLaTeX

```
(%i63) tex(ratsimp(G(-1))) $
$${{\sqrt{e}}\over{e+2\,\sqrt{e}}}$$
(%i64) tex(ratsimp(G(1))) $
$${{e+\sqrt{e}}\over{e+2\,\sqrt{e}}}$$
(%i65) tex(ratsimp(G_1(y))) $
$$2\,\log \left(\sqrt{e}\,y+2\,y\right)-1$$
(%i66) tex(ratsimp(G_2(y))) $
$${{\sqrt{e}\,\left(2\,y-1\right)+4\,y-2}\over{\sqrt{e}}}$$
(%i67) tex(ratsimp(G_3(y))) $
$$2\,\log \left(-{{\sqrt{e}}\over{2\,y+\sqrt{e}\,\left(y-1\right)-2}}
 \right)$$
```

Obtención de G^{-1} con Maxima (`Ginversa.mac`)

Sesión con `tex` para la presentación en PDFLaTeX

$$G(-1) = \frac{\sqrt{e}}{e + 2\sqrt{e}}$$

$$G(1) = \frac{e + \sqrt{e}}{e + 2\sqrt{e}}$$

$$G_1^{-1}(y) = 2\,\log\left(\sqrt{e}\,y + 2\,y\right) - 1$$

$$G_2^{-1}(y) = \frac{\sqrt{e}\,(2\,y - 1) + 4\,y - 2}{\sqrt{e}}$$

$$G_3^{-1} = 2\,\log\left(-\frac{\sqrt{e}}{2\,y + \sqrt{e}\,(y-1) - 2}\right)$$

Obtención de H^{-1} con Maxima

```
logcontract (solve (H1(x)=y, x)) ;
expand (solve (H2(x)=y, x)) ;
solve (rootscontract (ratsimp (H3(x))) = y, x);
```

Método de aceptación y rechazo (demostración 1)

- Sea $X' = (Y \mid \text{aceptación})$ y su densidad

$$f_{X'}(x) = f(Y = x \mid \text{aceptación}) = \frac{\Pr[\text{aceptación} \mid Y = x] \cdot g(x)}{\Pr[\text{aceptación}]}$$

- Como

$$\begin{aligned} \Pr[\text{aceptación}] &= \int \Pr[\text{aceptación} \mid Y = y]\, g(y)\, \mathrm{d}y \\ &= \int \frac{f(y)}{C \cdot g(y)}\, g(y)\, \mathrm{d}y = \frac{1}{C} \end{aligned}$$

- se tiene

$$f_{X'}(x) = f(Y = x \mid \text{aceptación}) = \frac{\frac{f(x)}{C \cdot g(x)} \cdot g(x)}{\frac{1}{C}} = f(x)$$

- luego X' tiene la misma distribución que X

Método de aceptación y rechazo (demostración 2)

- Sea $X' = (Y \mid \text{aceptación})$ y su función de distribución

$$F_{X'}(x) = \Pr[Y \leqslant x \mid \text{aceptación}] = \frac{\Pr[\text{aceptación} \cap Y \leqslant x]}{\Pr[\text{aceptación}]}$$

- Sean $(U \mid Y = y) \sim \mathcal{U}(0, C \cdot g(y))$ y "aceptación" $=$ "$U \leqslant f(y)$"

$$\Pr[\text{aceptación} \cap Y \leqslant x] = \int_{-\infty}^{x} \int_{0}^{f(y)} f_{U,Y}(u, y)\, \mathrm{d}u\ \mathrm{d}y\ =$$

$$= \int_{-\infty}^{x} \int_{0}^{f(y)} f_{U|Y=y}(u)\, g(y)\, \mathrm{d}u\ \mathrm{d}y\ = \int_{-\infty}^{x} \int_{0}^{f(y)} \frac{1}{C \cdot g(y)}\, \mathrm{d}u\ g(y)\, \mathrm{d}y\ =$$

$$= \int_{-\infty}^{x} \frac{f(y)}{C \cdot g(y)}\, g(y)\, \mathrm{d}y\ = \frac{F_X(x)}{C}$$