Προσομοίωση

Βασίλης Κατσιάνος

Σεπτέμβριος 2023

Περιεχόμενα

1 Παραγωγή Τυχαίων Μεταβλητών **1**

2 Μέθοδος Monte Carlo **50**

3 Προσομοίωση Συστημάτων Διακριτών Γεγονότων **67**

4 Τεχνικές Μείωσης Διασποράς **85**

5 Αλγόριθμοι Markov Chain Monte Carlo **122**

6 Μέθοδος Bootstrap **159**

1 Παραγωγή Τυχαίων Μεταβλητών

Μέθοδος Αντίστροφου Μετασχηματισμού

Ορισμός 1.1. Έστω συνάρτηση κατανομής F με στήριγμα S. Ορίζουμε τη γενικευμένη αντίστροφη της F ως $F^{-} : [0,1] \to S$ με $F^{-}(u) = \inf\{x \in S : F(x) \geqslant u\}$.

Σημείωση 1.1. Από τον ορισμό της γενικευμένης αντίστροφης F^{-} και το γεγονός ότι η συνάρτηση κατανομής F είναι αύξουσα, συμπεραίνουμε ότι $F(x) \geqslant u \Leftrightarrow F^{-}(u) \leqslant x$.

Θεώρημα 1.1. Έστω $U \sim \text{Unif}(0,1)$. Τότε, η τυχαία μεταβλητή $X = F^{-}(U)$ έχει συνάρτηση κατανομής F.

Απόδειξη. Θυμόμαστε ότι $F_U(u) = \mathbb{P}(U \leqslant u) = u$ για $u \in [0,1]$. Για $x \in S$, υπολογίζουμε ότι:

$$F_X(x) = \mathbb{P}(X \leqslant x) = \mathbb{P}\left[F^{-}(U) \leqslant x\right] = \mathbb{P}\left[F(x) \geqslant U\right] = F_U\left(F(x)\right) = F(x).$$

□

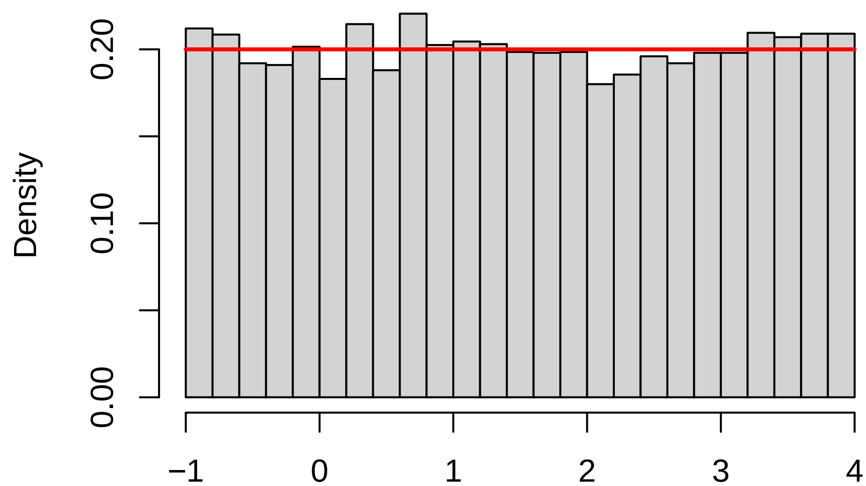
Παραγωγή Απόλυτα Συνεχών Τυχαίων Μεταβλητών

Σημείωση 1.2. Αν η συνάρτηση κατανομής F είναι απόλυτα συνεχής, τότε $F^{-} \equiv F^{-1}$.

Παράδειγμα 1.1. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Unif}[a,b]$. Για $x \in [a,b]$, υπολογίζουμε ότι:

$$F(x) = \frac{x-a}{b-a}, \quad F^{-1}(u) = (b-a)u + a.$$

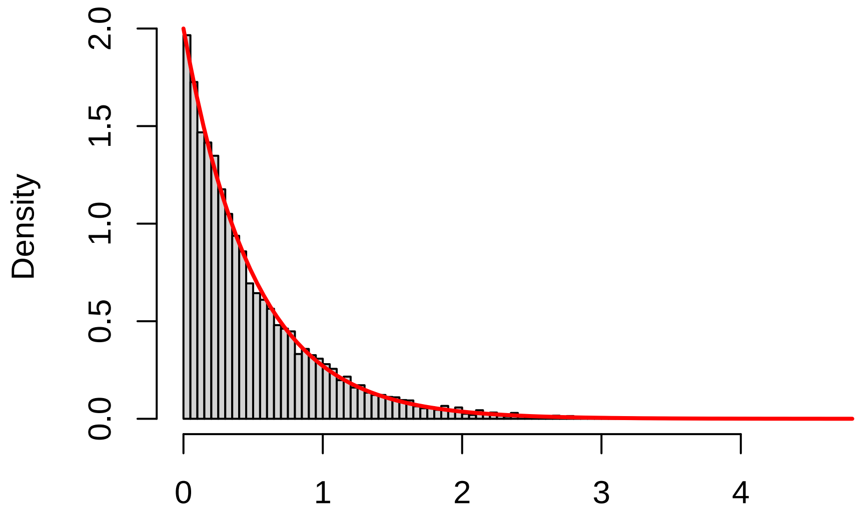
```
n = 10000
a = -1
b = 4
U = runif(n)
X = (b - a) * U + a
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dunif(x, a, b), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.2. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Exp}(\lambda)$. Για $x > 0$, υπολογίζουμε ότι:

$$F(x) = 1 - e^{-\lambda x}, \quad F^{-1}(u) = -\frac{1}{\lambda}\log(1-u).$$

```
n = 10000
lambda = 2
U = runif(n)
X = -log(1 - U)/lambda
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(x, lambda), add = TRUE, col = "red", lwd = 2)
```



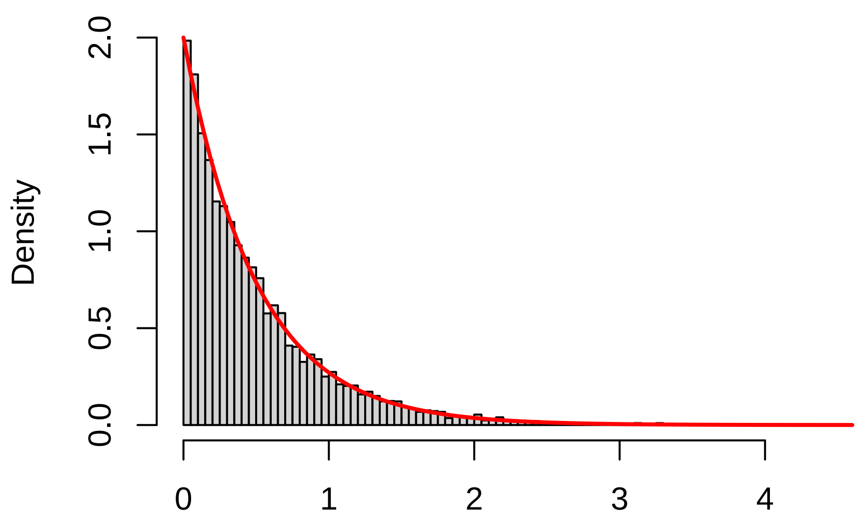
Λήμμα 1.1. Αν $U \sim \text{Unif}[0,1]$, τότε $V = 1 - U \sim \text{Unif}[0,1]$.

Απόδειξη. Για $v \in [0,1]$, υπολογίζουμε ότι:

$$F_V(v) = \mathbb{P}(V \leqslant v) = \mathbb{P}(1 - U \leqslant v) = \mathbb{P}(U \geqslant 1 - v) = 1 - \mathbb{P}(U \leqslant 1 - v) = 1 - (1 - v) = v = F_U(v).$$

□

```
n = 10000
lambda = 2
U = runif(n)
X = -log(U)/lambda
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(x, lambda), add = TRUE, col = "red", lwd = 2)
```



Λήμμα 1.2. Αν $X \sim \text{Exp}(\lambda)$ και $\mu > 0$, τότε οι τυχαίες μεταβλητές $Y = (X \mid X > \mu)$ και $W = X + \mu$ είναι ισόνομες.

Απόδειξη. Αρχικά, γνωρίζουμε ότι $\mathbb{P}(X > \mu) = 1 - F_X(\mu) = e^{-\lambda\mu}$. Για $x > \mu$, υπολογίζουμε ότι:

$$F_Y(x) = \mathbb{P}\left(X \leqslant x \mid X > \mu\right) = \frac{\mathbb{P}\left(X \leqslant x, X > \mu\right)}{\mathbb{P}(X > \mu)} = \frac{F_X(x) - F_X(\mu)}{F_X(\mu)} = \frac{e^{-\lambda\mu} - e^{-\lambda x}}{e^{-\lambda\mu}} = 1 - e^{-\lambda(x-\mu)},$$

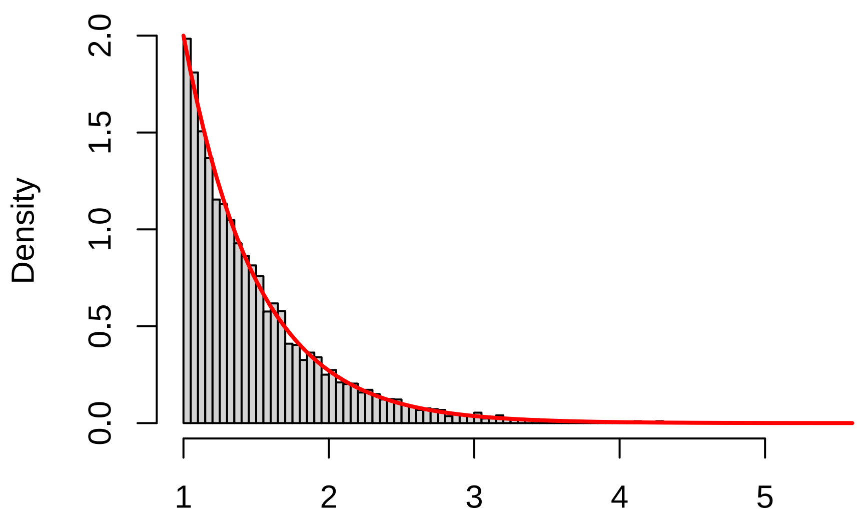
$$F_W(x) = \mathbb{P}\left(W \leqslant x\right) = \mathbb{P}\left(X + \mu \leqslant x\right) = F_X(x - \mu) = 1 - e^{-\lambda(x-\mu)}.$$

□

Σημείωση 1.3. Το παραπάνω λήμμα είναι απόρροια της αμνήμονης ιδιότητας της εκθετικής κατανομής.

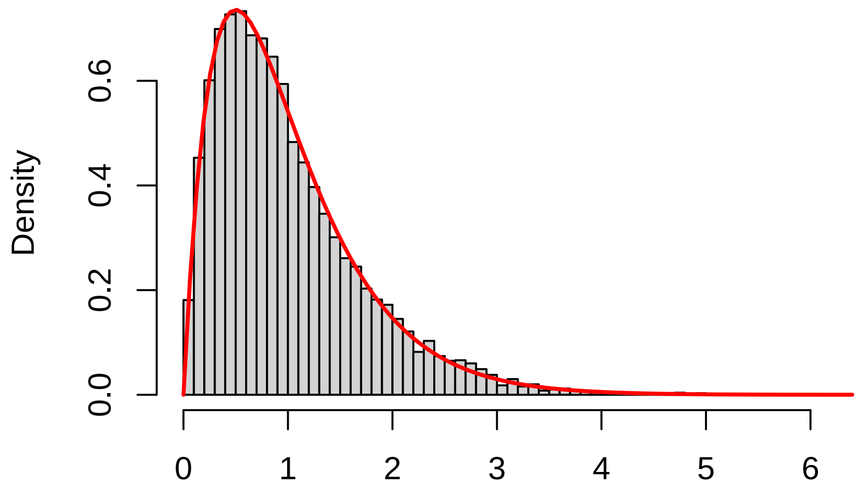
Παράδειγμα 1.3. Έστω $X \sim \mathrm{Exp}(\lambda)$ και $\mu > 0$. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ από τη δεσμευμένη κατανομή της X δεδομένου ότι $X > \mu$.

```
n = 10000
lambda = 2
mu = 1
U = runif(n)
X = mu - log(U)/lambda
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(x - mu, lambda), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.4. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \ldots, X_n \sim \mathrm{Gamma}(k, \lambda)$ για $k \in \mathbb{N}$. Έστω ανεξάρτητες τυχαίες μεταβλητές $Y_1, \ldots, Y_k \sim \mathrm{Exp}(\lambda)$. Τότε, γνωρίζουμε ότι $Y_1 + \cdots + Y_k \sim \mathrm{Gamma}(k, \lambda)$.

```
n = 10000
k = 2
lambda = 2
U = matrix(runif(n * k), n)
Y = -log(U)/lambda
X = rowSums(Y)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dgamma(x, k, lambda), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.5. Για $x \in \mathbb{R}$, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ με συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \frac{\lambda}{2} e^{-\lambda|x-\mu|}.$$

Για $x \leqslant \mu$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \int_{-\infty}^{x} f(y)dy = \int_{-\infty}^{x} \frac{\lambda}{2} e^{-\lambda(\mu-y)} dy = \frac{1}{2} e^{-\lambda(\mu-x)}.$$

Για $x > \mu$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant \mu) + \mathbb{P}(\mu < X \leqslant x) = F(\mu) + \int_{\mu}^{x} \frac{\lambda}{2} e^{-\lambda(y-\mu)} dy = \frac{1}{2} - \frac{1}{2} e^{-\lambda(x-\mu)} + \frac{1}{2} = 1 - \frac{1}{2} e^{-\lambda(x-\mu)}.$$

Για $u \in [0, F(\mu)] = [0, 0.5]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x = \mu + \frac{1}{\lambda} \log(2u).$$

Για $u \in (F(\mu), 1] = (0.5, 1]$, υπολογίζουμε ότι:

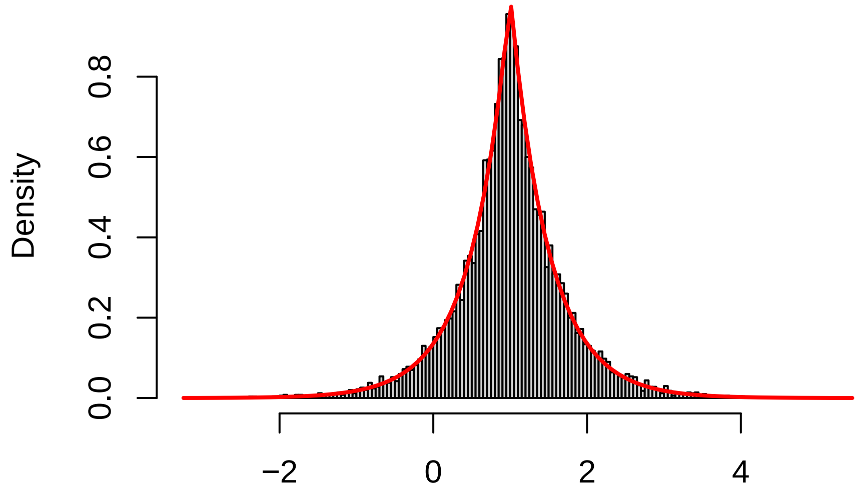
$$F(x) = u \Leftrightarrow x = \mu - \frac{1}{\lambda} \log\left[2(1-u)\right].$$

Επομένως, συμπεραίνουμε ότι:

$$F^{-1}(u) = \begin{cases} \mu + \frac{1}{\lambda} \log(2u), & 0 \leqslant u \leqslant 0.5 \\ \mu - \frac{1}{\lambda} \log\left[2(1-u)\right], & 0.5 < u \leqslant 1 \end{cases}.$$

```
n = 10000
lambda = 2
mu = 1
U = runif(n)
X = ifelse(U <= 0.5, mu + log(2 * U)/lambda, mu - log(2 * (1 - U))/lambda)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
```

```r
curve(dexp(abs(x - mu), lambda)/2, add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.6. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ με συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \begin{cases} \frac{x-2}{2}, & 2 \leqslant x \leqslant 3 \\ \frac{6-x}{6}, & 3 < x \leqslant 6 \end{cases}.$$

Για $x \in [2,3]$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \int_2^x f(y)dy = \int_2^x \frac{y-2}{2}dy = \frac{x^2}{4} - x + 1.$$

Για $x \in (3,6]$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant 3) + \mathbb{P}(3 < X \leqslant x) = F(3) + \int_3^x \frac{6-y}{6}dy = \frac{1}{4} + x - \frac{x^2}{12} - \frac{9}{4} = -\frac{x^2}{12} + x - 2.$$

Για $u \in [0, F(3)] = [0, 0.25]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x^2 - 4x + 4(1-u) = 0 \Leftrightarrow x = \frac{4 \pm \sqrt{16u}}{2} = 2\left(1 \pm \sqrt{u}\right).$$

Η λύση $x = 2\left(1 - \sqrt{u}\right) \in [1,2]$ απορρίπτεται, οπότε παίρνουμε ότι $x = 2\left(1 + \sqrt{u}\right) \in [2,3]$.

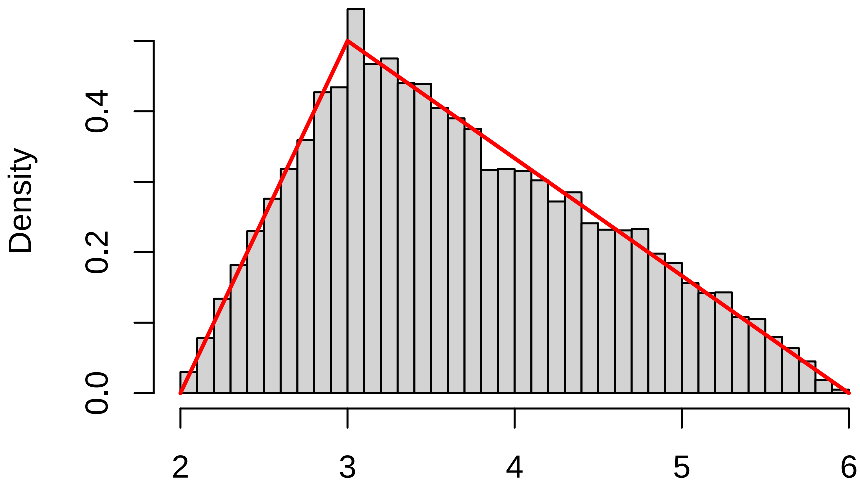
Για $u \in (F(3), 1] = (0.25, 1]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x^2 - 12x + 12(u+2) = 0 \Leftrightarrow x = \frac{12 \pm \sqrt{48(1-u)}}{2} = 2\left[3 \pm \sqrt{3(1-u)}\right].$$

Η λύση $x = 2\left[3 + \sqrt{3(1-u)}\right] \in [6,9)$ απορρίπτεται, οπότε παίρνουμε ότι $x = 2\left[3 - \sqrt{3(1-u)}\right] \in (3,6]$. Επομένως, συμπεραίνουμε ότι:

$$F^{-1}(u) = \begin{cases} 2\left(1 + \sqrt{u}\right), & 0 \leqslant u \leqslant 0.25 \\ 2\left[3 - \sqrt{3(1-u)}\right], & 0.25 < u \leqslant 1 \end{cases}.$$

```
f = function(x) {
    ifelse(x >= 2 & x < 3, (x - 2)/2, ifelse(x >= 3 & x < 6, (6 - x)/6, 0))
}

n = 10000
U = runif(n)
X = ifelse(U <= 0.25, 2 * (1 + sqrt(U)), 2 * (3 - sqrt(3 * (1 - U))))
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(f(x), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.7. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ με συνάρτηση πυκνότητας πιθανότητας $f(x) = 1 - |1 - x|$ για $x \in [0, 2]$. Για $x \in [0, 1]$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \int_0^x f(y)dy = \int_0^x ydy = \frac{x^2}{2}.$$

Για $x \in (1, 2]$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant 1) + \mathbb{P}(1 < X \leqslant x) = F(1) + \int_1^x 2 - ydy = \frac{1}{2} + 2x - \frac{x^2}{2} - \frac{3}{2} = -\frac{x^2}{2} + 2x - 1.$$

Για $u \in [0, F(1)] = [0, 0.5]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x^2 = 2u \Leftrightarrow x = \pm\sqrt{2u}.$$

Η λύση $x = -\sqrt{2u} \in [-1, 0]$ απορρίπτεται, οπότε παίρνουμε ότι $x = \sqrt{2u} \in [0, 1]$.

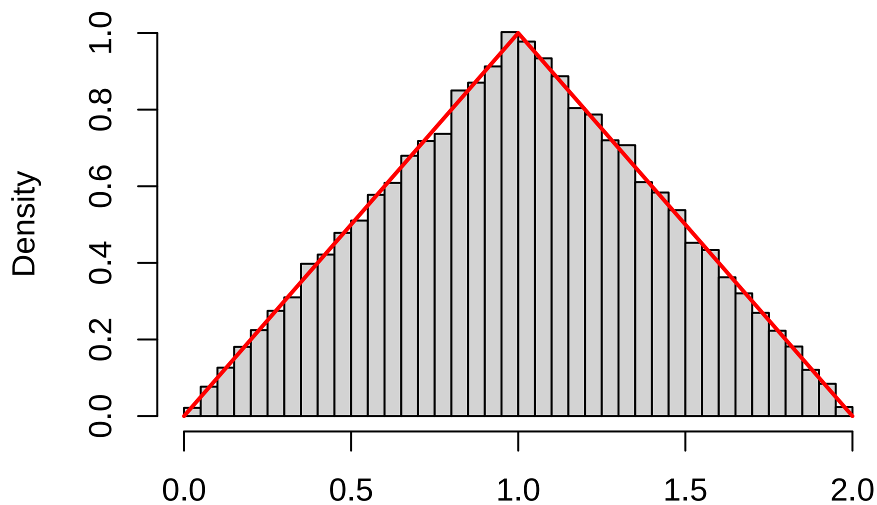
Για $u \in (F(1), 1] = (0.5, 1]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x^2 - 4x + 2(u + 1) = 0 \Leftrightarrow x = \frac{4 \pm \sqrt{8(1 - u)}}{2} = 2 \pm \sqrt{2(1 - u)}.$$

Η λύση $x = 2 + \sqrt{2(1 - u)} \in [2, 3)$ απορρίπτεται, οπότε παίρνουμε ότι $x = 2 - \sqrt{2(1 - u)} \in (1, 2]$. Επομένως, συμπεραίνουμε ότι:

$$F^{-1}(u) = \begin{cases} \sqrt{2u}, & 0 \leqslant u \leqslant 0.5 \\ 2 - \sqrt{2(1-u)}, & 0.5 < u \leqslant 1 \end{cases}.$$

```
n = 50000
U = runif(n)
X = ifelse(U <= 0.5, sqrt(2 * U), 2 - sqrt(2 * (1 - U)))
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(1 - abs(1 - x), add = TRUE, col = "red", lwd = 2)
```



Λήμμα 1.3. Έστω ανεξάρτητες τυχαίες μεταβλητές $U, V \sim \mathrm{Unif}[0,1]$. Τότε, η τυχαία μεταβλητή $X = U + V$ έχει συνάρτηση πυκνότητας πιθανότητας $f(x) = 1 - |1 - x|$ για $x \in [0,2]$.

Απόδειξη. Για $x \in [0,2]$, υπολογίζουμε ότι:

$$F_X(x) = \mathbb{P}(U + V \leqslant x) = \int_0^1 \mathbb{P}(U \leqslant x - v) f_V(v) dv = \int_0^1 F_U(x - v) dv.$$

Παρατηρούμε ότι:

$$F_U(x - v) = \begin{cases} 1, & v \leqslant x - 1 \\ x - v, & x - 1 < v \leqslant x \\ 0, & v > x \end{cases}.$$

Για $x \in [0,1]$, παίρνουμε ότι:

$$F_X(x) = \int_0^x x - v dv = \frac{x^2}{2}.$$

Για $x \in (1,2]$, παίρνουμε ότι:

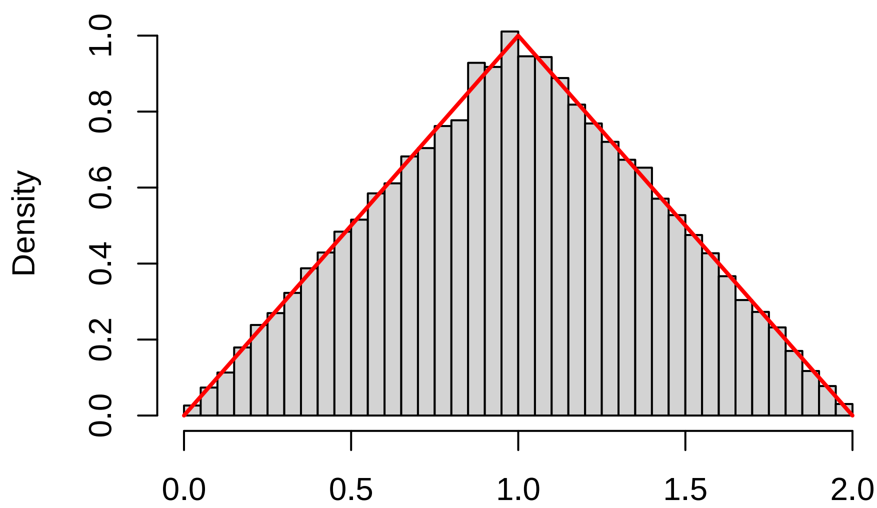
$$F_X(x) = \int_0^{x-1} 1 dv + \int_{x-1}^1 x - v dv = x - 1 + x - \frac{1}{2} - x(x-1) + \frac{(x-1)^2}{2} = -\frac{x^2}{2} + 2x - 1.$$

Επομένως, συμπεραίνουμε ότι:

$$f_X(x) = \begin{cases} x, & 0 \leqslant x \leqslant 1 \\ 2 - x, & 1 < x \leqslant 2 \end{cases}.$$

Παρατηρούμε ότι $f_X(x) = 1 - |1 - x| = f(x)$ για $x \in [0, 2]$. □

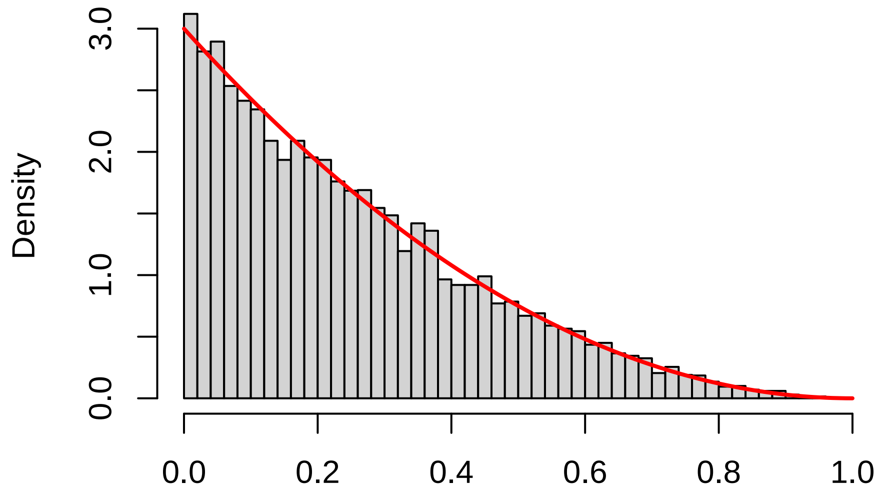
```
n = 50000
U = runif(n)
V = runif(n)
X = U + V
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(1 - abs(1 - x), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.8. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \mathrm{Beta}(1, k)$. Για $x \in [0, 1]$, γνωρίζουμε ότι $f(x) = k(1 - x)^{k-1}$. Υπολογίζουμε ότι:

$$F(x) = 1 - (1 - x)^k, \quad F^{-1}(u) = 1 - (1 - u)^{1/k}.$$

```
n = 10000
k = 3
U = runif(n)
X = 1 - U^(1/k)
hist(X, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = NA)
curve(dbeta(x, 1, k), add = TRUE, col = "red", lwd = 2)
```



Λήμμα 1.4. Έστω ανεξάρτητες τυχαίες μεταβλητές Y_i με κοινό στήριγμα S και συναρτήσεις κατανομής F_i για $i = 1, 2, \dots, k$. Τότε,

i. Η τυχαία μεταβλητή $X = \max\{Y_1, \dots, Y_k\}$ έχει συνάρτηση κατανομής:

$$F(x) = \prod_{i=1}^{k} F_i(x).$$

ii. Η τυχαία μεταβλητή $X = \min\{Y_1, \dots, Y_k\}$ έχει συνάρτηση κατανομής:

$$F(x) = 1 - \prod_{i=1}^{k} \left[1 - F_i(x)\right].$$

Απόδειξη. i. Για $x \in S$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}\left(\max\{Y_1, \dots, Y_k\} \leqslant x\right) = \mathbb{P}\left(Y_1 \leqslant x, \dots, Y_k \leqslant x\right) = \prod_{i=1}^{k} \mathbb{P}(Y_i \leqslant x) = \prod_{i=1}^{k} F_i(x).$$

ii. Για $x \in S$, υπολογίζουμε ότι:

$$\begin{aligned} F(x) &= \mathbb{P}\left(\min\{Y_1, \dots, Y_k\} \leqslant x\right) = 1 - \mathbb{P}\left(\min\{Y_1, \dots, Y_k\} > x\right) \\ &= 1 - \mathbb{P}\left(Y_1 > x, \dots, Y_k > x\right) = 1 - \prod_{i=1}^{k} \mathbb{P}(Y_i > x) = 1 - \prod_{i=1}^{k} \left[1 - F_i(x)\right]. \end{aligned}$$

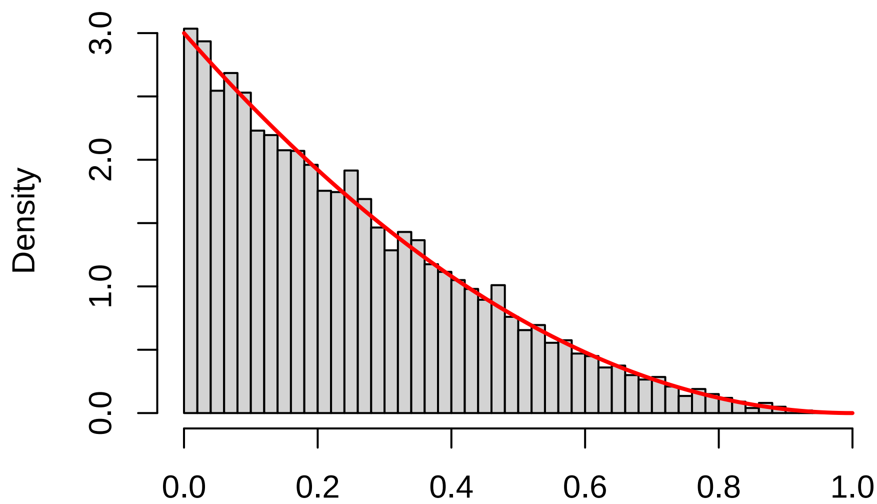
□

Για $x \in [0, 1]$, παρατηρούμε ότι:

$$F(x) = 1 - \prod_{i=1}^{k} (1 - x).$$

Θεωρούμε τις ανεξάρτητες τυχαίες μεταβλητές $Y_1, \dots, Y_k \sim \mathrm{Unif}[0, 1]$ με συνάρτηση κατανομής $F_Y(x) = x$ για $x \in [0, 1]$. Σύμφωνα με το προηγούμενο λήμμα, συμπεραίνουμε ότι η τυχαία μεταβλητή $\min\{Y_1, \dots, Y_k\}$ έχει ως συνάρτηση κατανομής την F.

```r
n = 10000
k = 3
U = matrix(runif(n * k), n)
X = apply(U, 1, min)
hist(X, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = NA)
curve(dbeta(x, 1, k), add = TRUE, col = "red", lwd = 2)
```



Μέθοδος Απόρριψης

Έστω συνάρτηση πυκνότητας πιθανότητας f με φραγμένο στήριγμα $S = [0,1]$ και ανεξάρτητες τυχαίες μεταβλητές $Y \sim \text{Unif}[0,1]$, $U \sim \text{Unif}[0,1]$. Ορίζουμε $M = \max_{x\in[0,1]} f(x)$ και $V = MU \sim \text{Unif}[0,M]$.

Σημείωση 1.4. Εφόσον η f είναι συνάρτηση πυκνότητας πιθανότητας με στήριγμα το $[0,1]$, θα πρέπει αναγκαστικά να ισχύει ότι $M > 1$.

Πρόταση 1.1. i. Το τυχαίο διάνυσμα (Y, V) ακολουθεί τη δισδιάστατη ομοιόμορφη κατανομή στο ορθογώνιο με βάση $[0,1]$ και ύψος $[0,M]$.

ii. Η δεσμευμένη κατανομή του τυχαίου διανύσματος (Y, V) δεδομένου ότι $f(Y) \geqslant V$ είναι η δισδιάστατη ομοιόμορφη στην περιοχή κάτω από το γράφημα της συνάρτησης πυκνότητας πιθανότητας f, δηλαδή στην περιοχή $\{(y,v) \in [0,1] \times [0,M] : f(y) \geqslant v\}$.

iii. Η περιθώρια κατανομή του Y δεδομένου ότι $f(Y) \geqslant V$ έχει συνάρτηση πυκνότητας πιθανότητας την f.

Απόδειξη. i. Για $y \in [0,1]$ και $v \in [0,M]$, υπολογίζουμε ότι:

$$F_{Y,V}(y,v) = \mathbb{P}\left(Y \leqslant y, V \leqslant v\right) = \mathbb{P}(Y \leqslant y)\mathbb{P}(V \leqslant v) = y \cdot \frac{v}{M},$$

$$f_{Y,V}(y,v) = \frac{\partial^2 F_{Y,V}(y,v)}{\partial v \partial y} = 1 \cdot \frac{1}{M} = \frac{1}{\int_0^1 \int_0^M 1 dv dy}.$$

ii. Για $y \in [0,1]$ και $v \in [0,M]$ με $f(y) \geqslant v$, υπολογίζουμε ότι:

$$\mathbb{P}\left[f(Y) \geqslant V\right] = \int_0^1 f_Y(y)\mathbb{P}\left[V \leqslant f(y)\right] dy = \int_0^1 \frac{f(y)}{M} dy = \frac{1}{M}\int_0^1 f(y) dy = \frac{1}{M},$$

$$\mathbb{P}\left[Y \leqslant y, V \leqslant v, f(Y) \geqslant V\right] = \int_0^y f_Y(x) \mathbb{P}\left[V \leqslant v, V \leqslant f(x)\right] dx = \frac{1}{M} \int_0^y \min\{v, f(x)\} dx,$$

$$F_{Y,V|F(Y)\geqslant V}(y, v) = \mathbb{P}\left[Y \leqslant y, V \leqslant v \mid f(Y) \geqslant V\right] = \frac{\mathbb{P}\left[Y \leqslant y, V \leqslant v, f(Y) \geqslant V\right]}{\mathbb{P}\left[f(Y) \geqslant V\right]} = \int_0^y \min\{v, f(x)\} dx,$$

$$f_{Y,V|f(Y)\geqslant V}(y, v) = \frac{\partial^2 F_{Y,V|F(Y)\geqslant V}(y, v)}{\partial v \partial y} = \frac{\partial \min\{v, f(y)\}}{\partial v} = \frac{\partial v}{\partial v} = 1 = \frac{1}{\int_S \int_0^{f(y)} 1 dv dy}.$$

iii. Για $y \in [0, 1]$, υπολογίζουμε ότι:

$$f_{Y|f(Y)\geqslant V}(y) = \int_0^{f(y)} f_{Y,V|f(Y)\geqslant V}(y, v) dv = \int_0^{f(y)} 1 dv = f(y).$$

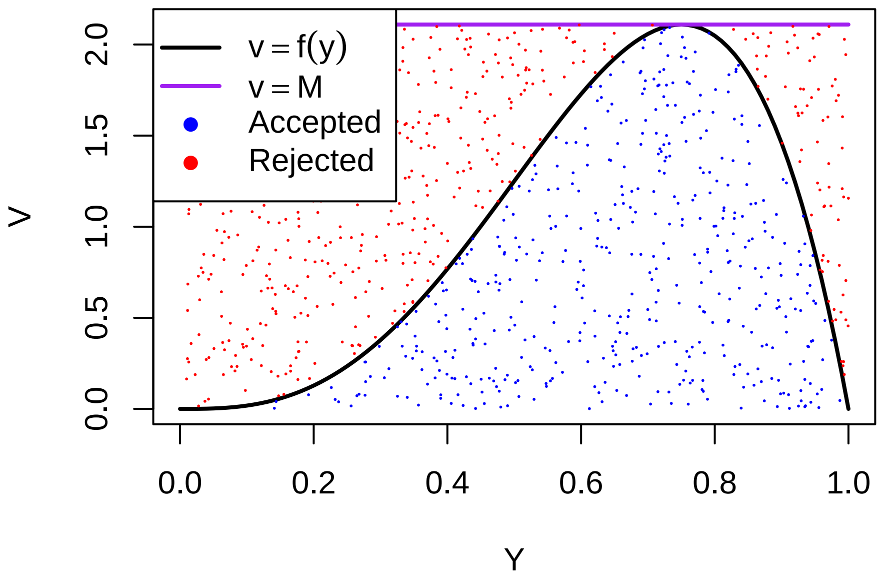
□

Παράδειγμα 1.9. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \ldots, X_n \sim \text{Beta}(4, 2)$. Για $x \in [0, 1]$, γνωρίζουμε ότι $f(x) = 20x^3(1 - x)$. Για $x \in (0, 1)$, υπολογίζουμε ότι $f'(x) = 20x^2(3 - 4x)$. Επομένως, παίρνουμε ότι $f'(x) = 0 \Leftrightarrow x = 3/4$, το οποίο συνεπάγεται ότι $M = f(3/4) = 135/64$.

```r
n = 1000
a = 4
b = 2
M = 135/64
print(M)
```

```
## [1] 2.109375
```

```r
Y = runif(n)
U = runif(n)
V = M * U
I = which(dbeta(Y, a, b) >= V)
J = which(dbeta(Y, a, b) < V)
curve(dbeta(x, a, b), lwd = 2, xlab = "Y", ylab = "V")
curve(M * dunif(x), add = TRUE, col = "purple", lwd = 2)
points(Y[I], V[I], col = "blue", pch = 16, cex = 0.2)
points(Y[J], V[J], col = "red", pch = 16, cex = 0.2)
legend("topleft", c(expression(v == f(y)), expression(v == M), "Accepted", "Rejected"),
    col = c("black", "purple", "blue", "red"), lty = c(1, 1, NA, NA), lwd = c(2,
        2, NA, NA), pch = c(NA, NA, 16, 16), bg = "white")
```



Algorithm 1.1 Μέθοδος Απόρριψης για Φραγμένο Στήριγμα $S = [0, 1]$

Είσοδος: Συνάρτηση πυκνότητας πιθανότητας f και ζητούμενο μέγεθος δείγματος n.

1: Υπολογίζουμε $M = \max_{x\in[0,1]} f(x)$.

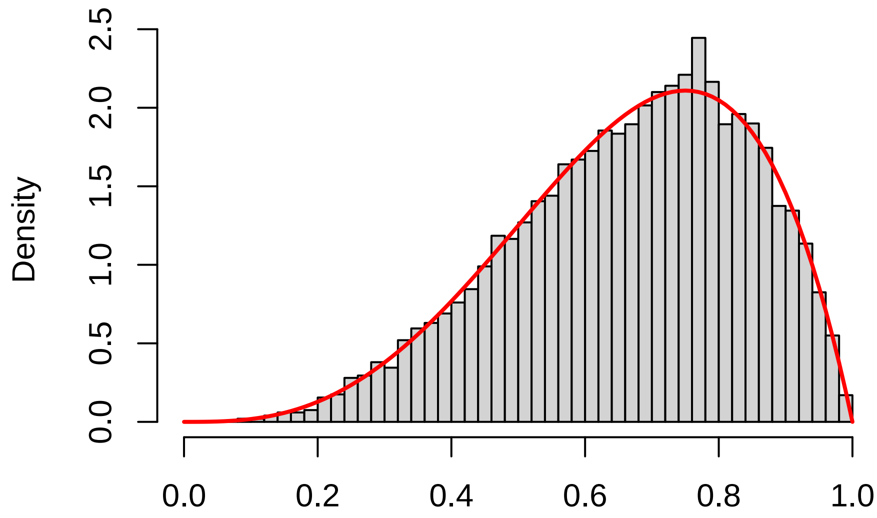
2: Για $i = 1, 2, \ldots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε $Y \sim \text{Unif}[0, 1]$, $U \sim \text{Unif}[0, 1]$ και θέτουμε $V = MU \sim \text{Unif}[0, 1]$.

ii: Αν $f(Y) \geqslant V$, θέτουμε $X_i = Y$. Διαφορετικά, επιστρέφουμε στο βήμα 2.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας f.

```
n = 10000
a = 4
b = 2
M = 135/64
X = numeric(n)
for (i in 1:n) {
    Y = runif(1)
    U = runif(1)
    V = M * U
    while (dbeta(Y, a, b) < V) {
        Y = runif(1)
        U = runif(1)
        V = M * U
    }
    X[i] = Y
}
hist(X, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = NA)
curve(dbeta(x, a, b), add = TRUE, col = "red", lwd = 2)
```



Έστω τώρα συνάρτηση πυκνότητας πιθανότητας f με γενικό στήριγμα S και προτείνουσα συνάρτηση πυκνότητας πιθανότητας g με στήριγμα $S_g \supseteq S$. Θεωρούμε ανεξάρτητες τυχαίες μεταβλητές $Y \sim g$ και $U \sim \text{Unif}[0,1]$. Ορίζουμε $M = \max_{x \in S} \frac{f(x)}{g(x)}$ και $V = Mg(Y)U$.

Σημείωση 1.5. Εφόσον οι f και g είναι συναρτήσεις πυκνότητας πιθανότητας, ισχύει ότι $M > 1$.

Πρόταση 1.2. i. Το τυχαίο διάνυσμα (Y, V) ακολουθεί τη δισδιάστατη ομοιόμορφη κατανομή στην περιοχή κάτω από το γράφημα της συνάρτησης Mg, δηλαδή στην περιοχή $\left\{(y,v) \in S_g \times [0,\infty] : Mg(y) \geqslant v\right\}$.

ii. Η δεσμευμένη κατανομή του τυχαίου διανύσματος (Y, V) δεδομένου ότι $f(Y) \geqslant V$ είναι η δισδιάστατη ομοιόμορφη στην περιοχή κάτω από το γράφημα της συνάρτησης πυκνότητας πιθανότητας f, δηλαδή στην περιοχή $\{(y,v) \in S \times [0,\infty] : f(y) \geqslant v\}$.

iii. Η περιθώρια κατανομή του Y δεδομένου ότι $f(Y) \geqslant V$ έχει συνάρτηση πυκνότητας πιθανότητας την f.

Απόδειξη. i. Για $y \in S_g$ και $v \in [0,\infty]$, υπολογίζουμε ότι:

$$F_{Y,V}(y,v) = \mathbb{P}\left[Y \leqslant y, V \leqslant v\right] = \int_{-\infty}^{y} g(x)\mathbb{P}\left[Mg(x)U \leqslant v\right]dx = \int_{-\infty}^{y} \cancel{g(x)} \cdot \frac{v}{M\cancel{g(x)}}dx = \frac{v}{M}\int_{-\infty}^{y} 1dx,$$

$$f_{Y,V}(y,v) = \frac{\partial^2 F_{Y,V}(y,v)}{\partial v \partial y} = \frac{\partial}{\partial v}\frac{v}{M} = \frac{1}{M} = \frac{1}{\int_{S_g}\int_0^{Mg(y)} 1dvdy}.$$

ii. Για $y \in S$ και $v \in [0,\infty]$ με $f(y) \geqslant v$, υπολογίζουμε ότι:

$$\mathbb{P}\left[f(Y) \geqslant V\right] = \int_S g(y)\mathbb{P}\left[Mg(y)U \leqslant f(y)\right]dy = \int_S \cancel{g(y)} \cdot \frac{f(y)}{M\cancel{g(y)}}dy = \frac{1}{M}\int_S f(y)dy = \frac{1}{M},$$

$$\mathbb{P}\left[Y \leqslant y, V \leqslant v, f(Y) \geqslant V\right] = \int_{-\infty}^{y} g(x)\mathbb{P}\left[Mg(x)U \leqslant v, Mg(x)U \leqslant f(x)\right]dx = \frac{1}{M}\int_{-\infty}^{y} \min\{v, f(x)\}dx,$$

$$F_{Y,V|F(Y)\geqslant V}(y,v) = \mathbb{P}\left[Y \leqslant y, V \leqslant v \mid f(Y) \geqslant V\right] = \frac{\mathbb{P}\left[Y \leqslant y, V \leqslant v, f(Y) \geqslant V\right]}{\mathbb{P}\left[f(Y) \geqslant V\right]} = \int_{-\infty}^{y} \min\{v, f(x)\}dx,$$

$$f_{Y,V|f(Y)\geqslant V}(y,v) = \frac{\partial^2 F_{Y,V|F(Y)\geqslant V}(y,v)}{\partial v \partial y} = \frac{\partial \min\{v, f(y)\}}{\partial v} = \frac{\partial v}{\partial v} = 1 = \frac{1}{\int_S \int_0^{f(y)} 1dvdy}.$$

iii. Για $y \in S$, υπολογίζουμε ότι:

$$f_{Y|f(Y)\geqslant V}(y) = \int_0^{f(y)} f_{Y,V|f(Y)\geqslant V}(y,v)dv = \int_0^{f(y)} 1dv = f(y).$$

$\square$

Παράδειγμα 1.10. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Gamma}(2, 0.5) \equiv \chi^2_4$. Αν θεωρήσουμε μία τυχαία μεταβλητή $X \sim \text{Gamma}(2, 0.5)$, τότε παρατηρούμε ότι $\mathbb{E}(X) = \frac{2}{0.5} = 4$. Έστω τυχαία μεταβλητή $Y \sim \text{Exp}(1/4)$ με προτείνουσα συνάρτηση πυκνότητας πιθανότητας $g(x) = \frac{1}{4}e^{-x/4}$ για $x > 0$. Παρατηρούμε ότι κι αυτή έχει μέση τιμή $\mathbb{E}(Y) = \frac{1}{1/4} = 4$. Ορίζουμε:

$$h(x) = \frac{f(x)}{g(x)} = xe^{-x/4}$$

Υπολογίζουμε ότι:

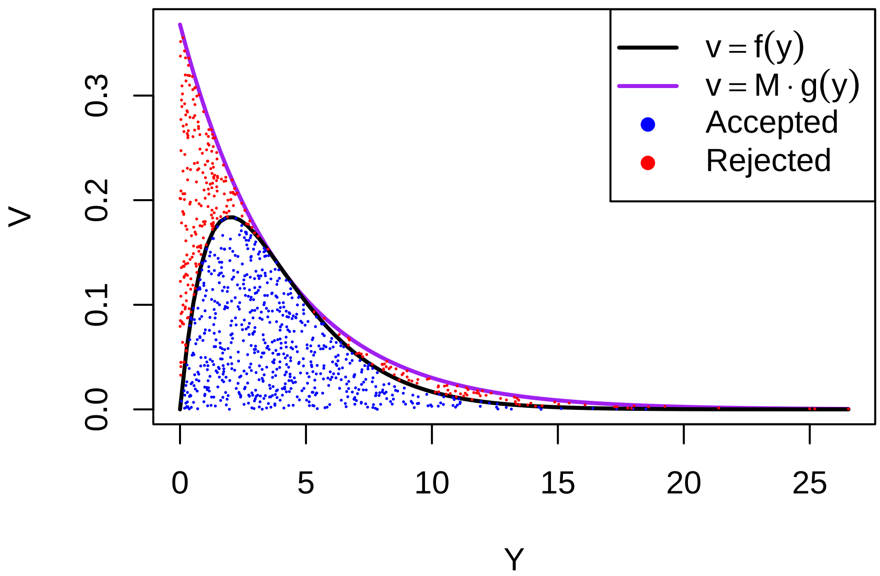
$$h'(x) = \left(1 - \frac{x}{4}\right)e^{-x/4}.$$

Επομένως, παίρνουμε ότι $h'(x) = 0 \Leftrightarrow x = 4$, το οποίο συνεπάγεται $M = h(4) = 4e^{-1}$.

```
n = 1000
a = 2
lambda = 0.5
M = 4 * exp(-1)
print(M)
```

```
## [1] 1.471518
```

```
W = runif(n)
Y = -log(W) * a/lambda
U = runif(n)
V = M * dexp(Y, lambda/a) * U
I = which(dgamma(Y, a, lambda) >= V)
J = which(dgamma(Y, a, lambda) < V)
curve(M * dexp(x, lambda/a), xlim = c(0, max(Y)), col = "purple", lwd = 2, xlab = "Y",
    ylab = "V")
curve(dgamma(x, a, lambda), add = TRUE, lwd = 2)
points(Y[I], V[I], col = "blue", pch = 16, cex = 0.2)
points(Y[J], V[J], col = "red", pch = 16, cex = 0.2)
legend("topright", c(expression(v == f(y)), expression(v == M %.% g(y)), "Accepted",
    "Rejected"), col = c("black", "purple", "blue", "red"), lty = c(1, 1, NA,
    NA), lwd = c(2, 2, NA, NA), pch = c(NA, NA, 16, 16))
```



Algorithm 1.2 Μέθοδος Απόρριψης για Γενικό Στήριγμα

Είσοδος: Συνάρτηση πυκνότητας πιθανότητας f, προτείνουσα συνάρτηση πυκνότητας πιθανότητας g και ζητούμενο μέγεθος δείγματος n.

1: Υπολογίζουμε $M = \max_{x \in \mathbb{R}} \frac{f(x)}{g(x)}$.

2: Για $i = 1, 2, \dots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

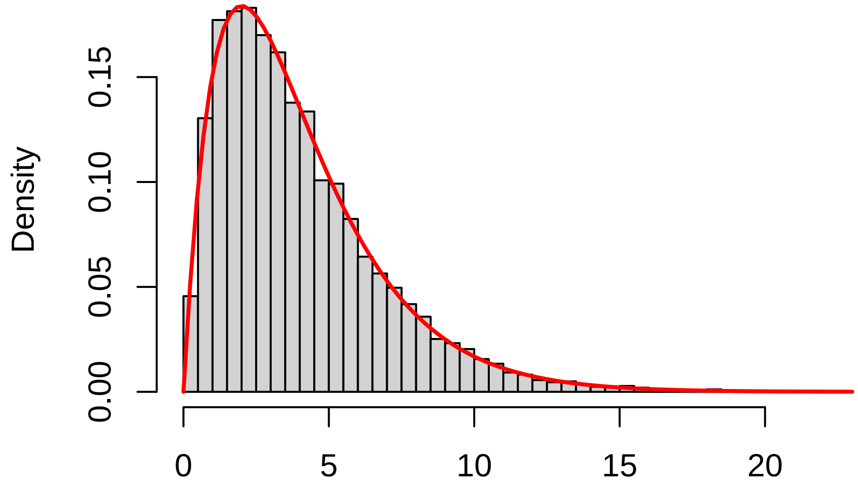
i: Προσομοιώνουμε $Y \sim g$, $U \sim \text{Unif}[0, 1]$ και θέτουμε $V = Mg(Y)U$.

ii: Αν $f(Y) \geqslant V$, θέτουμε $X_i = Y$. Διαφορετικά, επιστρέφουμε στο βήμα 2.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας f.

```
n = 10000
a = 2
lambda = 0.5
M = 4 * exp(-1)
X = numeric(n)
for (i in 1:n) {
    W = runif(1)
    Y = -log(W) * a/lambda
    U = runif(1)
    V = M * dexp(Y, lambda/a) * U
    while (dgamma(Y, a, lambda) < V) {
        W = runif(1)
        Y = -log(W) * a/lambda
        U = runif(1)
        V = M * dexp(Y, lambda/a) * U
    }
    X[i] = Y
}
```

```r
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dgamma(x, a, lambda), add = TRUE, col = "red", lwd = 2)
```



Θεώρημα 1.2. i. Η πιθανότητα αποδοχής του X_i δεδομένου ότι $Y = y$ είναι $\frac{f(y)}{Mg(y)}$.

ii. Η προσομοίωση του X_i απαιτεί πεπερασμένο πλήθος επαναλήψεων με πιθανότητα 1. Το μέσο πλήθος επαναλήψεων μέχρι την προσομοίωση του X_i είναι M.

Απόδειξη. i. Για $y \in S_g$, υπολογίζουμε ότι:

$$\mathbb{P}\left[f(Y) \geqslant V \mid Y = y\right] = \mathbb{P}\left[Mg(y)U \leqslant f(y)\right] = \frac{f(y)}{Mg(y)}.$$

ii. Υπολογίσαμε ότι η πιθανότητα αποδοχής του X_i, δηλαδή η πιθανότητα $\mathbb{P}\left[f(Y) \geqslant V\right]$, ισούται με $\frac{1}{M}$. Εφόσον κάθε προσπάθεια προσομοίωσης του X_i είναι ανεξάρτητη από τις προηγούμενες και καθεμία επιτυγχάνει με πιθανότητα $\frac{1}{M}$, συμπεραίνουμε ότι το πλήθος των προσπαθειών μέχρι την προσομοίωση του X_i ακολουθεί τη γεωμετρική κατανομή με πιθανότητα επιτυχίας $\frac{1}{M}$. Επομένως, το μέσο πλήθος προσπαθειών μέχρι την προσομοίωση του X_i δίνεται από τη μέση τιμή αυτής της γεωμετρικής κατανομής, η οποία είναι M. ☐

Σημείωση 1.6. Η μέθοδος απόρριψης είναι περισσότερο αποδοτική όταν M κοντά στο 1. Σε αυτήν την περίπτωση, απαιτείται μικρό πλήθος προσπαθειών μέχρι την προσομοίωση του X_i.

Παράδειγμα 1.11. Γενικότερα, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \ldots, X_n \sim \text{Gamma}\,(a, \lambda)$. Έστω τυχαία μεταβλητή $Y \sim \text{Exp}(\mu)$ με προτείνουσα συνάρτηση πυκνότητας πιθανότητας $g_\mu(x) = \mu e^{-\mu x}$ για $x > 0$. Ορίζουμε:

$$h_\mu(x) = \frac{f(x)}{g_\mu(x)} = \frac{\frac{\lambda^a}{\Gamma(a)} x^{a-1} e^{-\lambda x}}{\mu e^{-\mu x}} = \frac{\lambda^a}{\mu \Gamma(a)} x^{a-1} e^{-(\lambda-\mu)x}.$$

Για $a > 1$, υπολογίζουμε ότι:

$$\frac{\partial h_\mu(x)}{\partial x} = \frac{\lambda^a}{\Gamma(a)} x^{a-2} e^{-(\lambda-\mu)x} \left[a - 1 - (\lambda - \mu)x\right],$$

$$\frac{\partial h_\mu(x)}{\partial x} = 0 \Leftrightarrow x = \frac{a-1}{\lambda - \mu},$$

$$M(\mu) = \max_{x \in \mathbb{R}} h_\mu(x) = h_\mu\left(\frac{a-1}{\lambda-\mu}\right) = \frac{\lambda^a}{\mu\Gamma(a)}\left(\frac{a-1}{\lambda-\mu}\right)^{a-1} e^{-(a-1)},$$

$$\frac{\partial M(\mu)}{\partial \mu} = \frac{\lambda^a}{\mu\Gamma(a)}\left(\frac{a-1}{\lambda-\mu}\right)^{a-1}\left(-\frac{1}{\mu}+\frac{a-1}{\lambda-\mu}\right) e^{-(a-1)},$$

$$\frac{\partial M(\mu)}{\partial \mu} = 0 \Leftrightarrow \mu = \frac{\lambda}{a},$$

$$M^* = \min_{\mu>0} M(\mu) = M\left(\frac{\lambda}{a}\right) = \frac{a^a}{\Gamma(a)} e^{-(a-1)}.$$

Επομένως, το μ το οποίο ελαχιστοποιεί το μέσο πλήθος των επαναλήψεων που απαιτούνται για την προσομοίωση μίας τυχαίας μεταβλητής από την κατανομή Gamma (a, λ) ισούται με $\frac{\lambda}{a}$ και το ελάχιστο μέσο πλήθος επαναλήψεων που απαιτούνται ισούται με $\frac{a^a}{\Gamma(a)} e^{-(a-1)}$.

Σημείωση 1.7. Γνωρίζουμε ότι $\chi^2(\nu) \equiv \text{Gamma}\left(\frac{\nu}{2}, \frac{1}{2}\right)$. Επομένως, το μ το οποίο ελαχιστοποιεί το μέσο πλήθος των επαναλήψεων που απαιτούνται για την προσομοίωση μίας τυχαίας μεταβλητής από την κατανομή $\chi^2(\nu)$ για $\nu > 2$ με προτείνουσα συνάρτηση πυκνότητας πιθανότητας $g(x) = \mu e^{-\mu x}$ για $x > 0$ ισούται με $\frac{1}{\nu}$ και το ελάχιστο μέσο πλήθος επαναλήψεων που απαιτούνται ισούται με $\left(\frac{\nu}{2}\right)^{\nu/2} \frac{1}{\Gamma(\nu/2)} e^{-(\nu-2)/2}$.

Παράδειγμα 1.12. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \ldots, X_n \sim \mathcal{N}(\mu, \sigma^2)$. Για $x \in \mathbb{R}$, θεωρούμε την προτείνουσα συνάρτηση πυκνότητας πιθανότητας:

$$g_\lambda(x) = \frac{\lambda}{2} e^{-\lambda|x-\mu|}.$$

Ορίζουμε:

$$h_\lambda(x) = \frac{f(x)}{g_\lambda(x)} = \frac{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}}{\frac{\lambda}{2} e^{-\lambda|x-\mu|}} = \sqrt{\frac{2}{\pi\sigma^2}} \frac{1}{\lambda} \exp\left\{\lambda|x-\mu| - \frac{1}{2\sigma^2}(x-\mu)^2\right\}.$$

Εφόσον η συνάρτηση h είναι συμμετρική γύρω από το $x = \mu$, θα μελετήσουμε τη συμπεριφορά της για $x \geqslant \mu$. Υπολογίζουμε ότι:

$$\frac{\partial h_\lambda(x)}{\partial x} = \sqrt{\frac{2}{\pi\sigma^2}} \frac{1}{\lambda}\left(\lambda - \frac{x-\mu}{\sigma^2}\right) \exp\left\{\lambda(x-\mu) - \frac{1}{2\sigma^2}(x-\mu)^2\right\},$$

$$\frac{\partial h_\lambda(x)}{\partial x} = 0 \Leftrightarrow x = \mu + \sigma^2\lambda,$$

$$M(\lambda) = \max_{x \in \mathbb{R}} h_\lambda(x) = h_\lambda\left(\mu + \sigma^2\lambda\right) = \sqrt{\frac{2}{\pi\sigma^2}} \frac{1}{\lambda} e^{\sigma^2\lambda^2/2},$$

$$\frac{\partial M(\lambda)}{\partial \lambda} = \sqrt{\frac{2}{\pi\sigma^2}}\left(\sigma^2 - \frac{1}{\lambda^2}\right) e^{\sigma^2\lambda^2/2},$$

$$\frac{\partial M(\lambda)}{\partial \lambda} = 0 \Leftrightarrow \lambda = \frac{1}{\sigma},$$

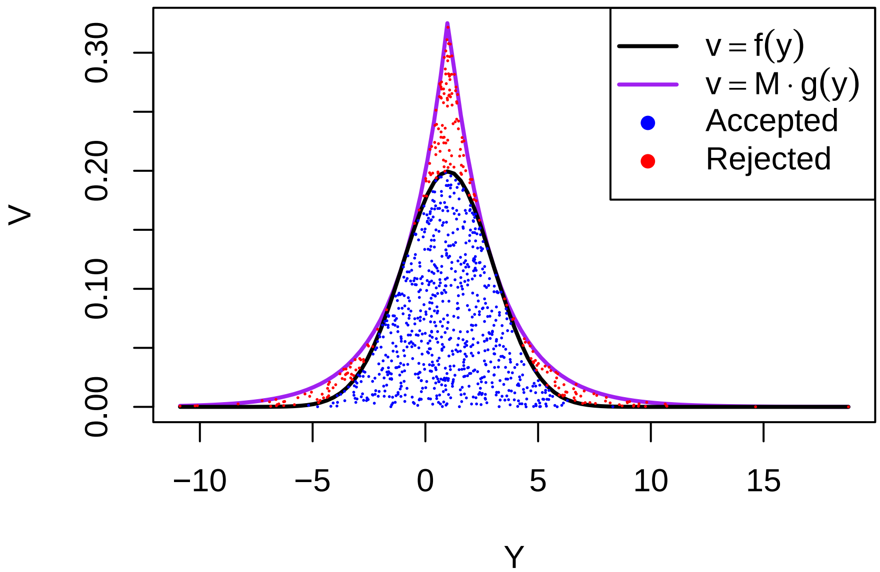
$$M^* = \min_{\lambda>0} M(\lambda) = M\left(\frac{1}{\sigma}\right) = \sqrt{\frac{2e}{\pi}}.$$

Επομένως, το λ το οποίο ελαχιστοποιεί το μέσο πλήθος των επαναλήψεων που απαιτούνται για την προσομοίωση μίας τυχαίας μεταβλητής από την κατανομή $\mathcal{N}(\mu, \sigma^2)$ ισούται με $\frac{1}{\sigma}$ και το ελάχιστο μέσο πλήθος επαναλήψεων που απαιτούνται ισούται με $\sqrt{2e/\pi}$.

```r
n = 1000
mu = 1
sigma = 2
lambda = 1/sigma
M = sqrt(2 * exp(1)/pi)
print(M)
```

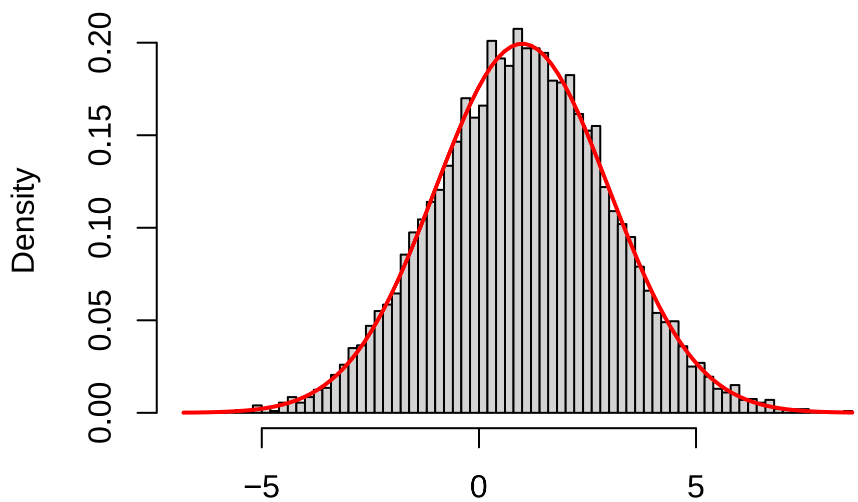
```
## [1] 1.315489
```

```r
W = runif(n)
Y = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
U = runif(n)
V = M * dexp(abs(Y - mu), lambda)/2 * U
I = which(dnorm(Y, mu, sigma) >= V)
J = which(dnorm(Y, mu, sigma) < V)
curve(M * dexp(abs(x - mu), lambda)/2, xlim = range(Y), col = "purple", lwd = 2,
    xlab = "Y", ylab = "V")
curve(dnorm(x, mu, sigma), add = TRUE, lwd = 2)
points(Y[I], V[I], col = "blue", pch = 16, cex = 0.2)
points(Y[J], V[J], col = "red", pch = 16, cex = 0.2)
legend("topright", c(expression(v == f(y)), expression(v == M %.% g(y)), "Accepted",
    "Rejected"), col = c("black", "purple", "blue", "red"), lty = c(1, 1, NA,
    NA), lwd = c(2, 2, NA, NA), pch = c(NA, NA, 16, 16))
```




```r
n = 10000
mu = 1
sigma = 2
lambda = 1/sigma
```

```
M = sqrt(2 * exp(1)/pi)
X = numeric(n)
for (i in 1:n) {
    W = runif(1)
    Y = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y - mu), lambda)/2 * U
    while (dnorm(Y, mu, sigma) < V) {
        W = runif(1)
        Y = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y - mu), lambda)/2 * U
    }
    X[i] = Y
}
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dnorm(x, mu, sigma), add = TRUE, col = "red", lwd = 2)
```



Έστω τυχαία μεταβλητή X με απόλυτα συνεχή συνάρτηση κατανομής G, συνάρτηση πυκνότητας πιθανότητας g, στήριγμα S και $a, b \in S$ με $a < b$. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ από τη δεσμευμένη κατανομή της X δεδομένου ότι $a \leqslant X \leqslant b$. Γνωρίζουμε ότι $\mathbb{P}(a \leqslant X \leqslant b) = G(b) - G(a)$. Για $x \in [a, b]$, υπολογίζουμε ότι:

$$F_{X|a\leqslant X\leqslant b}(x) = \mathbb{P}\left(X \leqslant x \mid a \leqslant X \leqslant b\right) = \frac{\mathbb{P}\left(X \leqslant x, a \leqslant X \leqslant b\right)}{\mathbb{P}(a \leqslant X \leqslant b)} = \frac{\mathbb{P}(a \leqslant X \leqslant x)}{\mathbb{P}(a \leqslant X \leqslant b)} = \frac{G(x) - G(a)}{G(b) - G(a)},$$

$$f_{X|a\leqslant X\leqslant b}(x) = \frac{\partial F_{X|a\leqslant X\leqslant b}(x)}{\partial x} = \frac{g(x)}{G(b) - G(a)}.$$

Παρατηρούμε ότι:

$$\frac{f_{X|a\leqslant X\leqslant b}(x)}{g(x)} = \begin{cases} \frac{1}{G(b)-G(a)}, & x \in [a,b] \\ 0, & x \notin [a,b] \end{cases}, \quad M = \max_{x\in S} \frac{f_{X|a\leqslant X\leqslant b}(x)}{g(x)} = \frac{1}{G(b)-G(a)}.$$

Αν $Y \sim g$ και $U \sim \text{Unif}[0,1]$, τότε:

$$\mathbb{P}\left[f_{X|a\leqslant X\leqslant b}(Y) \geqslant Mg(Y)U \mid Y\right] = \frac{f_{X|a\leqslant X\leqslant b}(Y)}{Mg(Y)} = \begin{cases} 1, & Y \in [a,b] \\ 0, & Y \notin [a,b] \end{cases}.$$

Με άλλα λόγια, αν η παραγμένη τιμή Y από τη συνάρτηση πυκνότητας πιθανότητας g ανήκει στο ζητούμενο διάστημα $[a,b]$, τότε γίνεται δεκτή με πιθανότητα 1. Διαφορετικά, απορρίπτεται με πιθανότητα 1. Επομένως, η προσομοίωση της τυχαίας μεταβλητής U είναι περιττή.

Παράδειγμα 1.13. Έστω $X \sim \text{Gamma}(3, 0.5)$, $a = 2$ και $b = 10$. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από τη δεσμευμένη κατανομή της X δεδομένου ότι $a \leqslant X \leqslant b$.

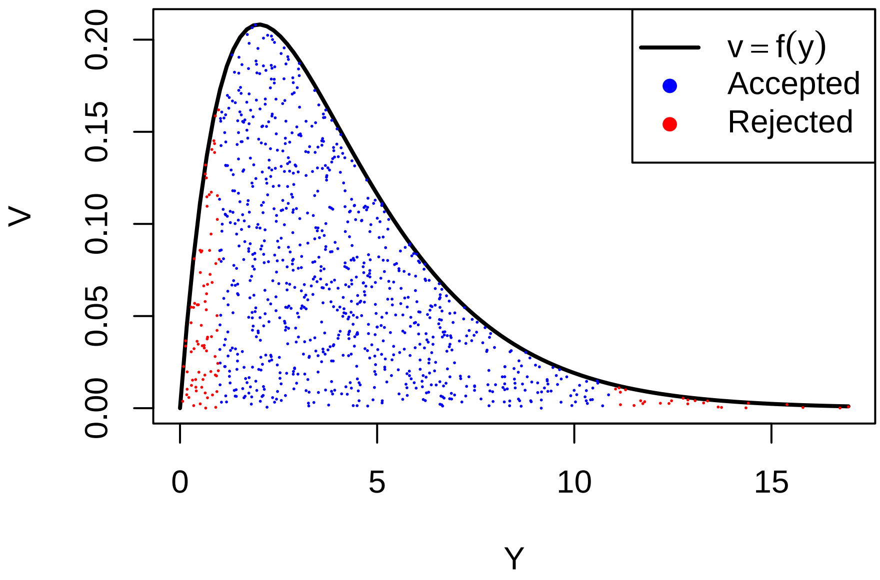
```
n = 1000
k = 2
lambda = 0.5
a = 1
b = 11
P = pgamma(b, k, lambda) - pgamma(a, k, lambda)
print(P)
```

```
## [1] 0.883232
```

```
M = 1/P
print(M)
```

```
## [1] 1.132205
```

```
W = matrix(runif(n * k), n)
R = -log(W)/lambda
Y = rowSums(R)
U = runif(n)
V = M * dgamma(Y, k, lambda) * U
I = which(Y >= a & Y <= b)
J = which(Y < a | Y > b)
curve(M * dgamma(x, k, lambda), xlim = c(0, max(Y)), lwd = 2, xlab = "Y", ylab = "V")
points(Y[I], V[I], col = "blue", pch = 16, cex = 0.2)
points(Y[J], V[J], col = "red", pch = 16, cex = 0.2)
legend("topright", c(expression(v == f(y)), "Accepted", "Rejected"), col = c("black",
    "blue", "red"), lty = c(1, NA, NA), lwd = c(2, NA, NA), pch = c(NA, 16,
    16))
```



Είσοδος: Προτείνουσα συνάρτηση πυκνότητας πιθανότητας g, ζητούμενο διάστημα $[a, b]$ και ζητούμενο μέγεθος δείγματος n.

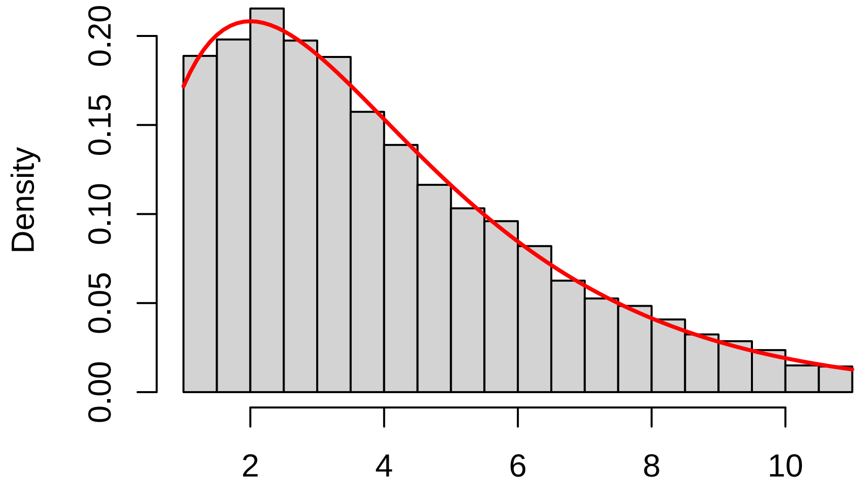
Για $i = 1, 2, \ldots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

1: Προσομοιώνουμε $Y \sim g$.

2: Αν $Y \in [a, b]$, θέτουμε $X_i = Y$. Διαφορετικά, επιστρέφουμε στο βήμα 1.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \ldots, X_n$.

```r
n = 10000
k = 2
lambda = 0.5
a = 1
b = 11
M = 1/(pgamma(b, k, lambda) - pgamma(a, k, lambda))
X = numeric(n)
for (i in 1:n) {
    U = runif(k)
    R = -log(U)/lambda
    Y = sum(R)
    while (Y < a || Y > b) {
        U = runif(k)
        R = -log(U)/lambda
        Y = sum(R)
    }
    X[i] = Y
}
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(M * dgamma(x, k, lambda), add = TRUE, col = "red", lwd = 2)
```



Σημείωση 1.8. Παρατηρούμε ότι το μέσο πλήθος επαναλήψεων μέχρι την προσομοίωση του X_i ισούται με:

$$M = \frac{1}{G(b) - G(a)} = \frac{1}{\mathbb{P}(a \leqslant X \leqslant b)} > 1.$$

Επομένως, η χρήση της συνάρτησης πυκνότητας πιθανότητας g ως προτείνουσας είναι αποδοτική όταν η πιθανότητα $\mathbb{P}(a \leqslant X \leqslant b)$ είναι μεγάλη. Διαφορετικά, θα ήταν πιο αποδοτική η χρήση της ομοιόμορφης κατανομής στο διάστημα $[a, b]$ ως προτείνουσας.

Παράδειγμα 1.14. Θέλουμε να παράγουμε τυχαίο δείγμα $(X_1, Y_1, Z_1), \ldots, (X_n, Y_n, Z_n)$ από την ομοιόμορφη κατανομή στο σύνολο $S = \{(x, y, z) \in \mathbb{R}^3 : x^2 + y^2 \leqslant 2z, x \geqslant y \geqslant z\}$. Παρατηρούμε ότι:

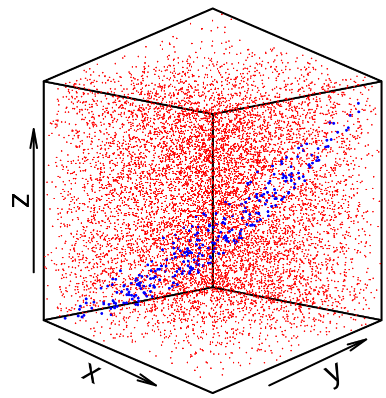
$$x^2 + y^2 \leqslant 2z \leqslant 2x \quad \Rightarrow \quad (x-1)^2 + y^2 \leqslant 1 \quad \Rightarrow \quad x \in [0, 2], \quad y \in [-1, 1],$$

$$x^2 + y^2 \leqslant 2z \leqslant 2y \quad \Rightarrow \quad x^2 + (y-1)^2 \leqslant 1 \quad \Rightarrow \quad x \in [-1, 1], \quad y \in [0, 2],$$

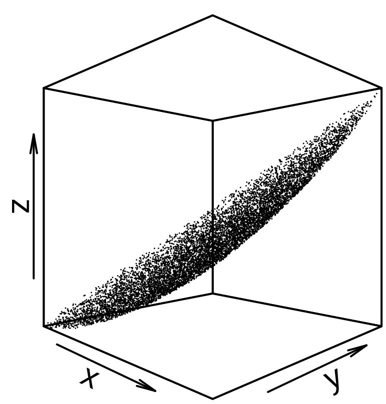
$$0 \leqslant x^2 + y^2 \leqslant 2z \quad \Rightarrow \quad z \geqslant 0.$$

Συμψηφίζοντας όλους τους περιορισμούς, συμπεραίνουμε ότι $S \subseteq [0, 1]^3$. Έστω (X, Y, Z) τυχαίο διάνυσμα που ακολουθεί την ομοιόμορφη κατανομή στον κύβο $[0, 1]^3$ με συνάρτηση πυκνότητας πιθανότητας $g(x, y, z) = 1$ για $x, y, z \in [0, 1]$. Συμπεραίνουμε ότι οι τυχαίες μεταβλητές X, Y, Z είναι ανεξάρτητες και ακολουθούν την κατανομή $\mathrm{Unif}[0, 1]$. Επομένως, αρκεί να παράγουμε τυχαίο δείγμα από τη δεσμευμένη κατανομή του (X, Y, Z) δεδομένου ότι $(X, Y, Z) \in S$.

```
library(plot3D)
n = 10000
X = runif(n)
Y = runif(n)
Z = runif(n)
I = which(X >= Y & Y >= Z & X^2 + Y^2 <= 2 * Z)
J = which(X < Y | Y < Z | X^2 + Y^2 > 2 * Z)
scatter3D(X[J], Y[J], Z[J], phi = 0, theta = 45, col = "red", pch = 16, cex = 0.1)
scatter3D(X[I], Y[I], Z[I], phi = 0, theta = 45, col = "blue", add = TRUE, pch = 16,
    cex = 0.2)
```

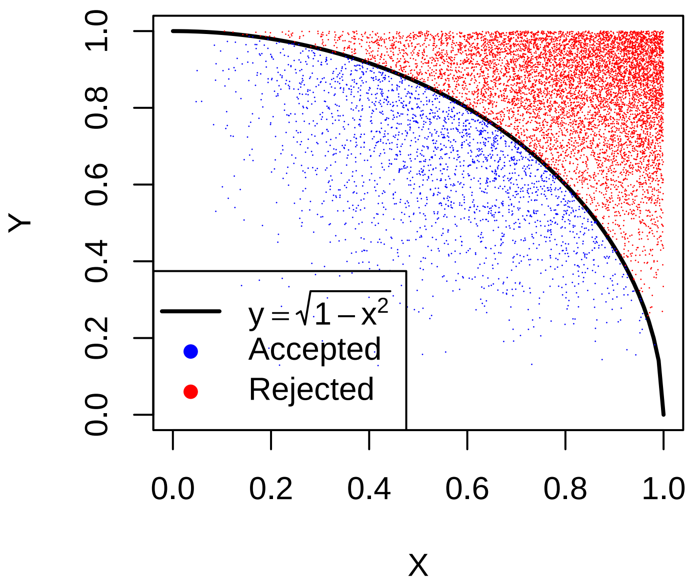



```
library(plot3D)
n = 10000
X = numeric(n)
Y = numeric(n)
Z = numeric(n)
for (i in 1:n) {
    X[i] = runif(1)
    Y[i] = runif(1)
    Z[i] = runif(1)
    while (X[i] < Y[i] || Y[i] < Z[i] || X[i]^2 + Y[i]^2 > 2 * Z[i]) {
        X[i] = runif(1)
        Y[i] = runif(1)
        Z[i] = runif(1)
    }
}
scatter3D(X, Y, Z, colvar = NA, phi = 0, theta = 45, pch = 16, cex = 0.1)
```



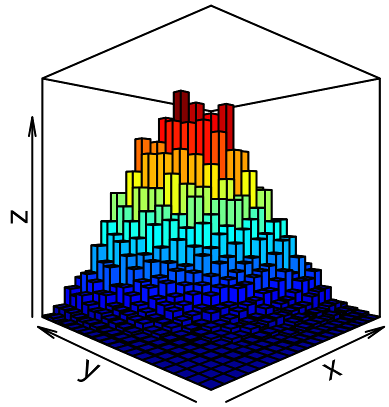
Παράδειγμα 1.15. Θέλουμε να παράγουμε τυχαίο δείγμα $(X_1, Y_1), \ldots, (X_n, Y_n)$ από τη συνάρτηση πυκνότητας πιθανότητας $f(x, y) = cx^2y^3$ για $(x, y) \in S = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leqslant 1, x, y \geqslant 0\} \subseteq [0, 1]^2$. Έστω $X \sim \mathrm{Beta}(3, 1)$ και $Y \sim \mathrm{Beta}(4, 1)$ ανεξάρτητες τυχαίες μεταβλητές με συναρτήσεις πυκνότητας πιθανότητας $f_X(x) = 3x^2$ και $f_Y(y) = 4y^3$ για $x, y \in [0, 1]$. Παρατηρούμε ότι $g(x, y) = f_X(x)f_Y(y) = 12x^2y^3$. Επομένως, αρκεί να παράγουμε τυχαίο δείγμα από τη δεσμευμένη κατανομή του (X, Y) δεδομένου ότι $(X, Y) \in S$. Για $x, y, u, v \in [0, 1]$, υπολογίζουμε ότι $F_X(x) = x^3$, $F_Y(y) = y^4$, $F_X^{-1}(u) = u^{1/3}$ και $F_Y^{-1}(v) = v^{1/4}$.

```r
n = 10000
U = runif(n)
X = U^(1/3)
V = runif(n)
Y = V^(1/4)
I = which(X^2 + Y^2 <= 1)
J = which(X^2 + Y^2 > 1)
curve(sqrt(1 - x^2), xlim = c(0, 1), lwd = 2, xlab = "X", ylab = "Y")
points(X[I], Y[I], col = "blue", pch = 16, cex = 0.1)
points(X[J], Y[J], col = "red", pch = 16, cex = 0.1)
legend("bottomleft", c(expression(y == sqrt(1 - x^2)), "Accepted", "Rejected"),
    col = c("black", "blue", "red"), lty = c(1, NA, NA), lwd = c(2, NA, NA),
    pch = c(NA, 16, 16))
```




```r
library(plot3D)
n = 50000
X = numeric(n)
Y = numeric(n)
for (i in 1:n) {
    U = runif(1)
    V = runif(1)
    X[i] = U^(1/3)
    Y[i] = V^(1/4)
    while (X[i]^2 + Y[i]^2 > 1) {
        U = runif(1)
        V = runif(1)
        X[i] = U^(1/3)
        Y[i] = V^(1/4)
    }
```

```
}
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 315,
    border = 1)
```



Μετασχηματισμός Box-Muller

Έστω ανεξάρτητες τυχαίες μεταβλητές $X, Y \sim \mathcal{N}(0,1)$. Τότε, υπολογίζουμε ότι:

$$f_{X,Y}(x,y) = f_X(x)f_Y(y) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2} \cdot \frac{1}{\sqrt{2\pi}}e^{-y^2/2} = \frac{1}{2\pi}e^{-(x^2+y^2)/2}.$$

Θεωρούμε την αλλαγή σε πολικές συντεταγμένες $D = X^2 + Y^2$, $\Theta = \arctan\frac{Y}{X}$ ή ισοδύναμα $X = \sqrt{D}\cos\Theta$, $Y = \sqrt{D}\sin\Theta$. Για $d > 0$ και $\theta \in [0, 2\pi]$, υπολογίζουμε ότι:

$$J_{X,Y}(d,\theta) = \frac{\partial(x,y)}{\partial(d,\theta)} = \begin{bmatrix} \frac{\partial x}{\partial d} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial d} & \frac{\partial y}{\partial \theta} \end{bmatrix} = \begin{bmatrix} \frac{\cos\theta}{2\sqrt{d}} & -\sqrt{d}\sin\theta \\ \frac{\sin\theta}{2\sqrt{d}} & \sqrt{d}\cos\theta \end{bmatrix},$$

$$\det\left[J_{X,Y}(d,\theta)\right] = \frac{1}{2}\cos^2\theta + \frac{1}{2}\sin^2\theta = \frac{1}{2},$$

$$f_{D,\Theta}(d,\theta) = \left|\det\left[J_{X,Y}(d,\theta)\right]\right| f_{X,Y}\left(\sqrt{d}\cos\theta, \sqrt{d}\sin\theta\right) = \frac{1}{2}\cdot\frac{1}{2\pi}e^{-d/2} = \frac{1}{2}e^{-d/2}\cdot\frac{1}{2\pi}.$$

Επομένως, συμπεραίνουμε ότι οι τυχαίες μεταβλητές D και Θ είναι ανεξάρτητες με $D \sim \mathrm{Exp}(1/2) \equiv \chi^2_2$ και $\Theta \sim \mathrm{Unif}[0, 2\pi]$.

Σημείωση 1.9. Αν $Z \sim \mathcal{N}(0,1)$, $\mu \in \mathbb{R}$ και $\sigma > 0$, τότε $X = \sigma Z + \mu \sim \mathcal{N}\left(\mu, \sigma^2\right)$.

```
n = 10000
mu = 1
sigma = 2
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = sigma * Z + mu
```

Algorithm 1.3 Μετασχηματισμός Box-Muller

Είσοδος: Μέση τιμή μ, τυπική απόκλιση σ και ζητούμενο μέγεθος δείγματος n.

Για $i = 1, 2, \dots, n/2$, επαναλαμβάνουμε τα παρακάτω βήματα:

1: Προσομοιώνουμε $U \sim \text{Unif}[0,1]$ και παίρνουμε $D = -2\log U \sim \text{Exp}(1/2)$.

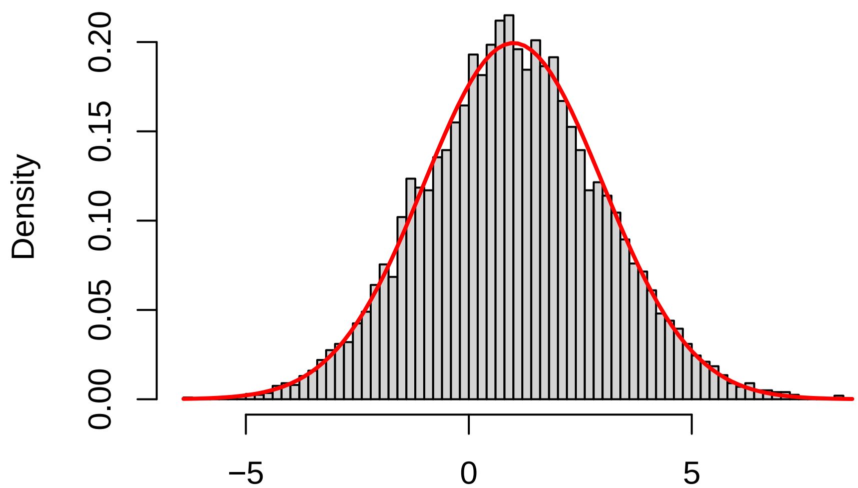
2: Προσομοιώνουμε $V \sim \text{Unif}[0,1]$ και παίρνουμε $\Theta = 2\pi V \sim \text{Unif}[0, 2\pi]$.

3: Θέτουμε $Z_1 = \sqrt{D}\cos\Theta \sim \mathcal{N}(0,1)$ και $Z_2 = \sqrt{D}\sin\Theta \sim \mathcal{N}(0,1)$.

4: Θέτουμε $X_{2i-1} = \sigma Z_1 + \mu \sim \mathcal{N}(\mu, \sigma^2)$ και $X_{2i} = \sigma Z_2 + \mu \sim \mathcal{N}(\mu, \sigma^2)$.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \dots, X_n$.

```r
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dnorm(x, mu, sigma), add = TRUE, col = "red", lwd = 2)
```



Παραγωγή Διακριτών Τυχαίων Μεταβλητών

Έστω διακριτή τυχαία μεταβλητή X με στήριγμα $S = \mathbb{N}$ και συνάρτηση πιθανότητας $p_j = \mathbb{P}(X = j)$ για $j = 0, 1, \dots$. Για $x \in \mathbb{N}$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \sum_{j=0}^{x} p_j,$$

$$F^{-}(u) = \inf\left\{ x \in S : \sum_{j=0}^{x} p_j \geqslant u \right\} = \begin{cases} 0, & 0 \leqslant u \leqslant p_0 \\ 1, & p_0 < u \leqslant p_0 + p_1 \\ 2, & p_0 + p_1 < u \leqslant p_0 + p_1 + p_2 \\ \dots \end{cases}.$$

Παράδειγμα 1.16. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση πιθανότητας

$p_1 = 0.2$, $p_2 = 0.15$, $p_3 = 0.25$, $p_4 = 0.4$. Υπολογίζουμε ότι:

$$F^{-}(u) = \begin{cases} 1, & 0 \leqslant u \leqslant 0.2 \\ 2, & 0.2 < u \leqslant 0.35 \\ 3, & 0.35 < u \leqslant 0.6 \\ 4, & 0.6 < u \leqslant 1 \end{cases}.$$

```
n = 10000
S = 1:4
pmf = c(0.2, 0.15, 0.25, 0.4)
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    j = 1
    cdf = pmf[1]
    while (U > cdf) {
        j = j + 1
        cdf = cdf + pmf[j]
    }
    X[i] = S[j]
}
table(factor(X, levels = S))/n
```

```
##
##      1      2      3      4
## 0.2010 0.1511 0.2511 0.3968
```

Σημείωση 1.10. Επειδή ο αλγόριθμος διατρέχει το στήριγμα της τυχαίας μεταβλητής X από την αρχή μέχρι το τέλος και η X είναι πιο πιθανό να πάρει τις τιμές στις οποίες αντιστοιχεί η μεγαλύτερη πιθανότητα, είναι πιο υπολογιστικά αποδοτικό πρώτα να κατατάξουμε τη συνάρτηση πιθανότητας της τυχαίας μεταβλητής σε φθίνουσα σειρά και μετά να προσομοιώσουμε από αυτή.

```
n = 10000
S = 1:4
pmf = c(0.2, 0.15, 0.25, 0.4)
I = order(pmf, decreasing = TRUE)
pmf = pmf[I]
S = I[S]
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    j = 1
    cdf = pmf[1]
    while (U > cdf) {
```

```
        j = j + 1
        cdf = cdf + pmf[j]
    }
    X[i] = S[j]
}
table(factor(X, levels = S))/n
```

```
##
##      4      3      1      2
## 0.4027 0.2454 0.1946 0.1573
```

Παράδειγμα 1.17. Θέλουμε να παράγουμε ένα πεπερασμένο μονοπάτι $X_1, X_2, \dots, X_n$ από μία Μαρκοβιανή αλυσίδα διακριτού χρόνου με πεπερασμένο χώρο καταστάσεων $S = \{1, 2, \dots, m\}$, αρχική κατανομή $a = [a_k]$ και πίνακα πιθανοτήτων μετάβασης $P = \left[p_{k,\ell}\right]$. Γνωρίζουμε ότι:

$$\begin{aligned}\mathbb{P}\left(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n\right) &= \mathbb{P}(X_1 = x_1)\mathbb{P}\left(X_2 = x_2 \mid X_1 = x_1\right) \cdots \mathbb{P}\left(X_n = x_n \mid X_{n-1} = x_{n-1}\right) \\ &= a_{x_1} p_{x_1,x_2} \cdots p_{x_{n-1},x_n}.\end{aligned}$$

Είσοδος: Χώρος καταστάσεων S, αρχική κατανομή a, πίνακας πιθανοτήτων μετάβασης P και ζητούμενο μήκος μονοπατιού n.

1: Προσομοιώνουμε τη X_1 από την αρχική κατανομή a.

2: Για $i = 2, 3, \dots, n$, επαναλαμβάνουμε το παρακάτω βήμα:

i: Προσομοιώνουμε τη X_i από τη συνάρτηση πιθανότητας η οποία δίνεται από τη γραμμή X_{i-1} του πίνακα πιθανοτήτων μετάβασης P.

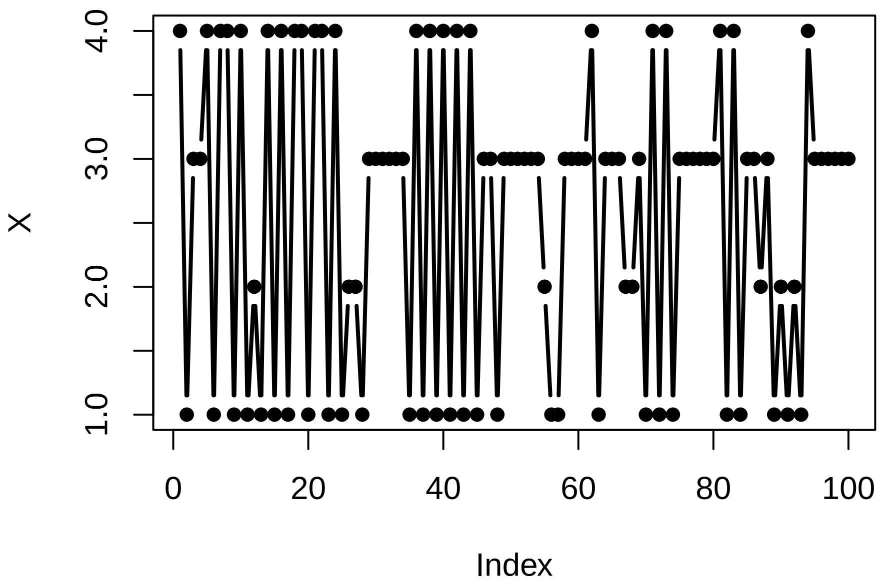
Έξοδος: Μονοπάτι $X_1, X_2, \dots, X_n$.

```
n = 100
m = 4
S = 1:m
a = c(0.2, 0.15, 0.25, 0.4)
P = rbind(c(0.1, 0.2, 0.3, 0.4), c(0.2, 0.5, 0.2, 0.1), c(0.2, 0.1, 0.6, 0.1),
    c(0.7, 0.1, 0.1, 0.1))
rownames(P) = S
colnames(P) = S
print(P)
```

```
##     1   2   3   4
## 1 0.1 0.2 0.3 0.4
## 2 0.2 0.5 0.2 0.1
## 3 0.2 0.1 0.6 0.1
## 4 0.7 0.1 0.1 0.1
```

```
X = numeric(n)
U = runif(1)
```

```
j = 1
cdf = a[1]
while (U > cdf) {
    j = j + 1
    cdf = cdf + a[j]
}
X[1] = S[j]
for (i in 2:n) {
    pmf = P[X[i - 1], ]
    U = runif(1)
    j = 1
    cdf = pmf[1]
    while (U > cdf) {
        j = j + 1
        cdf = cdf + pmf[j]
    }
    X[i] = S[j]
}
plot(X, type = "b", pch = 16, lwd = 2)
```



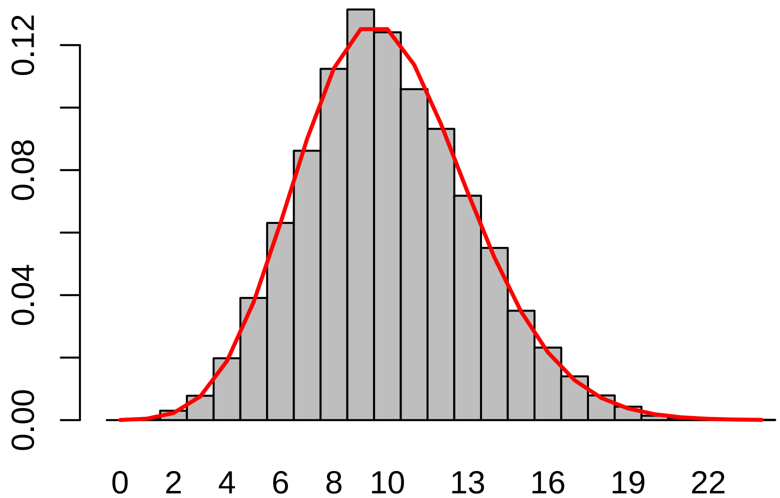
Παράδειγμα 1.18. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Poisson}(\lambda)$. Για $j = 0, 1, \dots$, γνωρίζουμε ότι:

$$p_j = e^{-\lambda} \frac{\lambda^j}{j!}.$$

Παρατηρούμε ότι:

$$p_0 = e^{-\lambda}, \quad p_{j+1} = \frac{\lambda}{j+1} p_j.$$

```r
n = 10000
lambda = 10
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    pmf = exp(-lambda)
    cdf = pmf
    while (U > cdf) {
        X[i] = X[i] + 1
        pmf = pmf * lambda/X[i]
        cdf = cdf + pmf
    }
}
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dpois(0:max(X), lambda), col = "red", lwd = 2)
```



Έστω διαδικασία Poisson $\{N(t) : t \geqslant 0\}$ ρυθμού λ με ενδιάμεσους χρόνους ανανέωσης $Y_1, Y_2, \dots \sim \text{Exp}(\lambda)$. Γνωρίζουμε ότι:

$$N(1) = \sup\left\{k \in \mathbb{N} : \sum_{j=1}^{k} Y_j \leqslant 1\right\} = \inf\left\{k \in \mathbb{N} : \sum_{j=1}^{k+1} Y_j > 1\right\} \sim \text{Poisson}(\lambda).$$

Είσοδος: Ρυθμός λ και ζητούμενο μέγεθος δείγματος n.

Για $i = 1, 2, \dots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

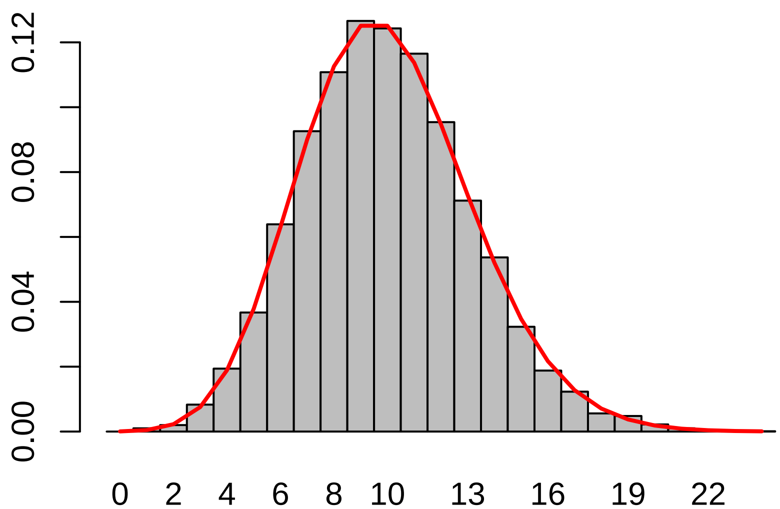
1: Θέτουμε $S \leftarrow 0$ και $k \leftarrow 0$.

2: Προσομοιώνουμε $U \sim \text{Unif}[0, 1]$, παίρνουμε $Y = -\frac{1}{\lambda} \log U \sim \text{Exp}(\lambda)$ και θέτουμε $S \leftarrow S + Y$.

3: Αν $S > 1$, τότε παίρνουμε $X_i = k$. Διαφορετικά, θέτουμε $k \leftarrow k + 1$ και επιστρέφουμε στο βήμα 2.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από την κατανομή Poisson(λ).

```
n = 10000
lambda = 10
X = numeric(n)
for (i in 1:n) {
    S = 0
    while (S <= 1) {
        U = runif(1)
        Y = -log(U)/lambda
        S = S + Y
        if (S <= 1) {
            X[i] = X[i] + 1
        }
    }
}
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dpois(0:max(X), lambda), col = "red", lwd = 2)
```



Παράδειγμα 1.19. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \mathrm{Unif}\{1, 2, \dots, k\}$. Για $x \in S$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \frac{x}{k},$$

$$F^{-}(u) = \inf\left\{x \in S : u \leqslant \frac{x}{k}\right\} = \inf\{x \in S : x \geqslant ku\} = \lfloor ku \rfloor + 1.$$

```
n = 10000
k = 5
U = runif(n)
X = floor(k * U) + 1
table(factor(X, levels = 1:k))/n
```

```
##
##      1      2      3      4      5
## 0.2010 0.2017 0.2005 0.1903 0.2065
```

Παράδειγμα 1.20. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Unif}\{a, a+1, \dots, b\}$. Για $x \in S$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \frac{x-a+1}{b-a+1},$$

$$F^{-}(u) = \inf\left\{x \in S : u \leqslant \frac{x-a+1}{b-a+1}\right\} = \inf\{x \in S : x \geqslant (b-a+1)u + a - 1\} = \lfloor (b-a+1)u \rfloor + a.$$

```
n = 10000
a = -1
b = 3
U = runif(n)
X = floor((b - a + 1) * U) + a
table(factor(X, levels = a:b))/n
```

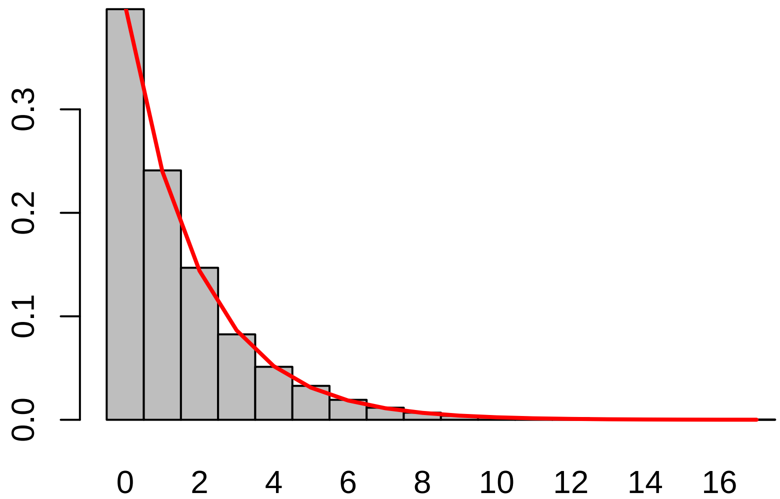
```
##
##     -1      0      1      2      3
## 0.2010 0.2017 0.2005 0.1903 0.2065
```

Παράδειγμα 1.21. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Geom}(p)$ με συνάρτηση πιθανότητας $p_j = p(1-p)^j$ για $j = 0, 1, \dots$. Για $x \in \mathbb{N}$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = p \sum_{j=0}^{x} (1-p)^j = p \cdot \frac{1-(1-p)^{x+1}}{1-(1-p)} = 1 - (1-p)^{x+1},$$

$$\begin{aligned} F^{-}(u) &= \min\left\{x \in \mathbb{N} : 1-(1-p)^{x+1} \geqslant u\right\} = \min\left\{x \in \mathbb{N} : (x+1)\log(1-p) \leqslant \log(1-u)\right\} \\ &= \min\left\{x \in \mathbb{N} : x \geqslant \frac{\log(1-u)}{\log(1-p)} - 1\right\} = \left\lfloor \frac{\log(1-u)}{\log(1-p)} \right\rfloor. \end{aligned}$$

```
n = 10000
p = 0.4
U = runif(n)
X = floor(log(U)/log(1 - p))
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dgeom(0:max(X), p), col = "red", lwd = 2)
```



Παράδειγμα 1.22. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Bernoulli}(p)$. Υπολογίζουμε ότι:

$$F^{-}(u) = \begin{cases} 0, & 0 \leqslant u \leqslant 1-p \\ 1, & 1-p < u \leqslant 1 \end{cases} = \begin{cases} 1, & 0 \leqslant 1-u < p \\ 0, & p \leqslant 1-u \leqslant 1 \end{cases}.$$

```
n = 10000
p = 0.4
U = runif(n)
X = as.numeric(U < p)
table(factor(X, levels = 0:1))/n
```

```
##
##      0      1
## 0.5973 0.4027
```

Παράδειγμα 1.23. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{Bin}(k, p)$. Για $j = 0, 1, \dots, k$, γνωρίζουμε ότι:

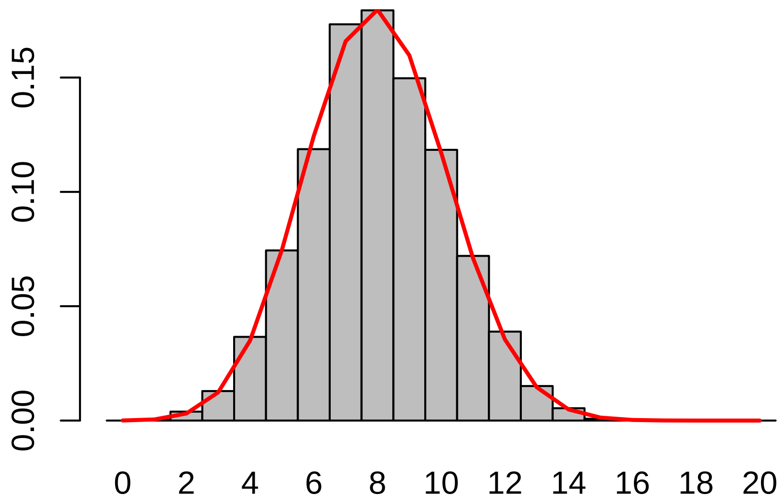
$$p_j = \binom{k}{j} p^j (1-p)^{k-j}.$$

Παρατηρούμε ότι:

$$p_0 = (1-p)^k, \quad p_{j+1} = \frac{k-j}{j+1} \frac{p}{1-p} p_j.$$

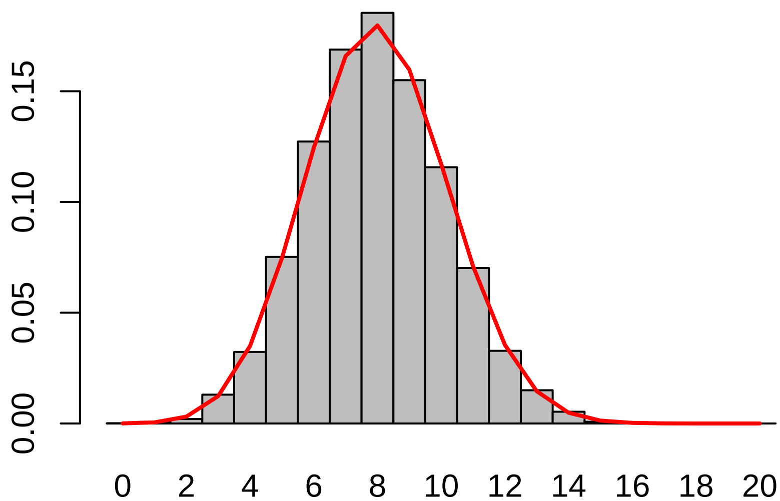
```
n = 10000
k = 20
p = 0.4
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    pmf = (1 - p)^k
    cdf = pmf
    while (U > cdf) {
        pmf = pmf * (k - X[i])/(X[i] + 1) * p/(1 - p)
```

```
        cdf = cdf + pmf
        X[i] = X[i] + 1
    }
}
barplot(table(factor(X, levels = 0:k))/n, space = 0)
lines(0:k + 0.5, dbinom(0:k, k, p), col = "red", lwd = 2)
```



Έστω ανεξάρτητες τυχαίες μεταβλητές $Y_1, \dots, Y_k \sim \text{Bernoulli}(p)$. Τότε, $Y_1 + \dots + Y_k \sim \text{Bin}(k, p)$.

```
n = 10000
k = 20
p = 0.4
U = matrix(runif(n * k), n)
Y = U < p
X = rowSums(Y)
barplot(table(factor(X, levels = 0:k))/n, space = 0)
lines(0:k + 0.5, dbinom(0:k, k, p), col = "red", lwd = 2)
```

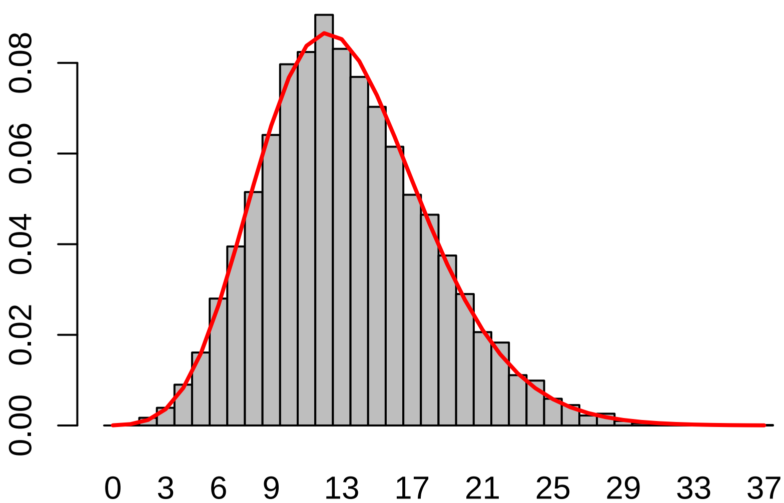


Παράδειγμα 1.24. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, \dots, X_n \sim \text{NegBin}(k, p)$ με συνάρτηση πιθανό-

τητας $p_j = \binom{j+k-1}{j} p^k (1-p)^j$ για $j = 0, 1, \ldots$. Παρατηρούμε ότι:

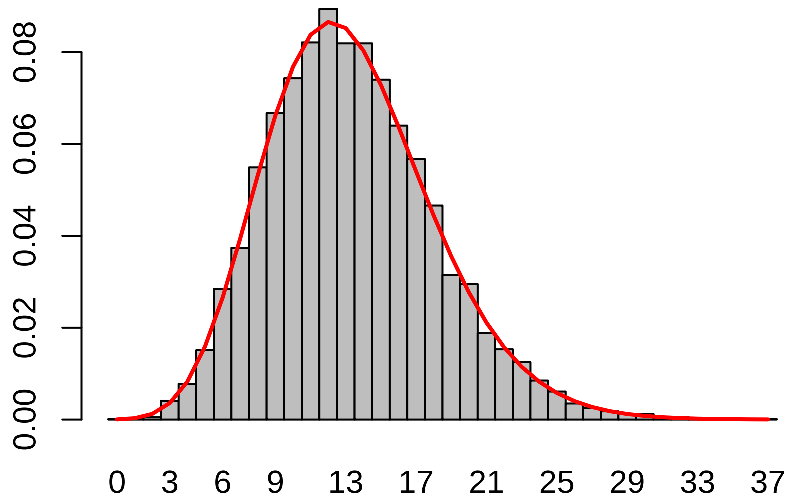
$$p_0 = p^k, \quad p_{j+1} = \frac{j+k}{j+1}(1-p)p_j.$$

```
n = 10000
k = 20
p = 0.6
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    pmf = p^k
    cdf = pmf
    while (U > cdf) {
        pmf = pmf * (1 - p) * (X[i] + k)/(X[i] + 1)
        cdf = cdf + pmf
        X[i] = X[i] + 1
    }
}
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dnbinom(0:max(X), k, p), col = "red", lwd = 2)
```



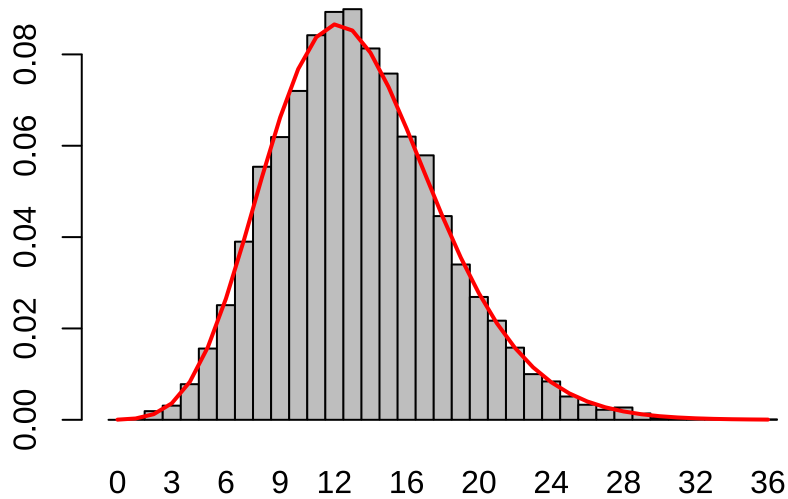
Έστω ανεξάρτητες τυχαίες μεταβλητές $Y_1, \ldots, Y_k \sim \text{Geom}(p)$. Τότε, $Y_1 + \cdots + Y_k \sim \text{NegBin}(k, p)$.

```
n = 10000
k = 20
p = 0.6
U = matrix(runif(n * k), n)
Y = floor(log(U)/log(1 - p))
X = rowSums(Y)
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dnbinom(0:max(X), k, p), col = "red", lwd = 2)
```



Γνωρίζουμε ότι η αρνητική διωνυμική κατανομή αναπαριστά το πλήθος αποτυχιών μέχρι την k-οστή επιτυχία σε ανεξάρτητες δοκιμές Bernoulli με κοινή πιθανότητα επιτυχίας p.

```
n = 10000
k = 20
p = 0.6
X = numeric(n)
for (i in 1:n) {
    success = 0
    while (success < k) {
        U = runif(1)
        Y = as.numeric(U < p)
        if (Y == 0) {
            X[i] = X[i] + 1
        } else {
            success = success + 1
        }
    }
}
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dnbinom(0:max(X), k, p), col = "red", lwd = 2)
```



Παράδειγμα 1.25. Θέλουμε να παράγουμε τυχαίο δείγμα $X^{(1)}, \dots, X^{(n)} \sim \text{Multinomial}(m, p_1, \dots, p_k)$. Έστω ανεξάρτητες τυχαίες μεταβλητές $Y_1, Y_2, \dots, Y_m$ με συνάρτηση πιθανότητας $p = [p_j]$. Τότε, γνωρίζουμε ότι:

$$\left(\sum_{\ell=1}^{m} \mathbb{1}_{\{Y_\ell = 1\}}, \dots, \sum_{\ell=1}^{m} \mathbb{1}_{\{Y_\ell = k\}} \right) \sim \text{Multinomial}(m, p_1, \dots, p_k).$$

```
n = 10000
m = 50
p = c(0.3, 0.1, 0.4, 0.2)
k = length(p)
X = matrix(0, n, k)
for (i in 1:n) {
    Y = numeric(m)
    for (j in 1:m) {
        U = runif(1)
        l = 1
        cdf = p[1]
        while (U > cdf) {
            l = l + 1
            cdf = cdf + p[l]
        }
        Y[j] = l
    }
    X[i, ] = table(factor(Y, levels = 1:k))
}
colMeans(X)
```

```
## [1] 15.0163  4.9851 20.0457  9.9529
```

Έστω $X = (X_1, X_2, \dots, X_k) \sim \text{Multinomial}(m, p_1, \dots, p_k)$. Τότε, παρατηρούμε ότι:

$$\begin{aligned}\mathbb{P}(X = x) &= \mathbb{P}\left(X_1 = x_1, X_2 = x_2, \dots, X_k = x_k\right) \\ &= \mathbb{P}(X_1 = x_1)\mathbb{P}\left(X_2 = x_2 \mid X_1 = x_1\right) \cdots \mathbb{P}\left(X_n = x_n \mid X_1 = x_1, \dots, X_{n-1} = x_{n-1}\right).\end{aligned}$$

Για $x_1 \in \{0, 1, \dots, m\}$, υπολογίζουμε σύμφωνα με το πολυωνυμικό θεώρημα ότι:

$$\begin{aligned}\mathbb{P}(X_1 = x_1) &= \sum_{x_2+\cdots+x_n=m-x_1} \mathbb{P}\left(X = x\right) = \sum_{x_2+\cdots+x_n=m-x_1} \frac{m!}{x_1!x_2!\cdots x_k!} p_1^{x_1} p_2^{x_2} \cdots p_k^{x_k} \\ &= \frac{m!}{x_1!(m-x_1)!} p_1^{x_1} \sum_{x_2+\cdots+x_n=m-x_1} \frac{(m-x_1)!}{x_2!\cdots x_k!} p_2^{x_2} \cdots p_k^{x_k} = \binom{m}{x_1} p_1^{x_1} \left(p_2 + \cdots + p_k\right)^{m-x_1} \\ &= \binom{m}{x_1} p_1^{x_1} (1-p_1)^{m-x_1},\end{aligned}$$

δηλαδή $X_1 \sim \text{Bin}(m, p_1)$. Για $x_2 \in \{0, 1, \dots, m - x_1\}$, υπολογίζουμε ότι:

$$\begin{aligned}\mathbb{P}(X_1 = x_1, X_2 = x_2) &= \sum_{x_3+\cdots+x_n=m-x_1-x_2} \mathbb{P}\left(X = x\right) = \sum_{x_3+\cdots+x_n=m-x_1-x_2} \frac{m!}{x_1!x_2!\cdots x_k!} p_1^{x_1} p_2^{x_2} \cdots p_k^{x_k} \\ &= \frac{m!}{x_1!x_2!(m-x_1-x_2)!} p_1^{x_1} p_2^{x_2} \sum_{x_3+\cdots+x_n=m-x_1-x_2} \frac{(m-x_1-x_2)!}{x_3!\cdots x_k!} p_3^{x_3} \cdots p_k^{x_k} \\ &= \frac{m!}{x_1!x_2!(m-x_1-x_2)!} p_1^{x_1} p_2^{x_2} \left(p_3 + \cdots + p_k\right)^{m-x_1-x_2} \\ &= \frac{m!}{x_1!x_2!(m-x_1-x_2)!} p_1^{x_1} p_2^{x_2} (1-p_1-p_2)^{m-x_1-x_2},\end{aligned}$$

$$\begin{aligned}\mathbb{P}\left(X_2 = x_2 \mid X_1 = x_1\right) &= \frac{\mathbb{P}\left(X_1 = x_1, X_2 = x_2\right)}{\mathbb{P}(X_1 = x_1)} = \frac{\frac{m!}{x_1!x_2!(m-x_1-x_2)!} p_1^{x_1} p_2^{x_2} (1-p_1-p_2)^{m-x_1-x_2}}{\frac{m!}{x_1!(m-x_1)!} p_1^{x_1} (1-p_1)^{m-x_1}} \\ &= \binom{m-x_1}{x_2} \left(\frac{p_2}{1-p_1}\right)^{x_2} \left(1 - \frac{p_2}{1-p_1}\right)^{m-x_1-x_2},\end{aligned}$$

δηλαδή $(X_2 \mid X_1 = x_1) \sim \text{Bin}\left(m - x_1, \frac{p_2}{1-p_1}\right)$. Για $\ell = 2, 3, \dots, k-2$, υπολογίζουμε ότι:

$$(X_{\ell+1} \mid X_1 = x_1, \dots, X_\ell = x_\ell) \sim \text{Bin}\left(m - x_1 - \cdots - x_\ell, \frac{p_{\ell+1}}{1 - p_1 - \cdots - p_\ell}\right).$$

Τέλος, παρατηρούμε ότι:

$$\mathbb{P}\left(X_k = x_k \mid X_1 = x_1, \dots, X_{k-1} = x_{k-1}\right) = \begin{cases} 1, & x_k = m - x_1 - \cdots - x_{k-1} \\ 0, & x_k \neq m - x_1 - \cdots - x_{k-1} \end{cases}.$$

```
n = 10000
m = 50
p = c(0.3, 0.1, 0.4, 0.2)
k = length(p)
X = matrix(0, n, k)
```

```
for (i in 1:n) {
    trials = m
    prob = 1
    for (l in 1:(k - 1)) {
        U = runif(trials)
        X[i, l] = sum(U < p[l]/prob)
        trials = trials - X[i, l]
        prob = prob - p[l]
    }
    X[i, k] = trials
}
colMeans(X)
```

```
## [1] 15.0006  4.9823 20.0646  9.9525
```

Σημείωση 1.11. Η πρώτη μέθοδος προσομοίωσης είναι περισσότερο αποδοτική όταν $m \ll k$, ενώ η δεύτερη μέθοδος προσομοίωσης είναι περισσότερο αποδοτική όταν $k \ll m$.

Παράδειγμα 1.26. Θέλουμε να παράγουμε τυχαίο δείγμα $X^{(1)}, \ldots, X^{(n)} \sim \text{Hypergeom}(m, r_1, \ldots, r_k)$ με συνάρτηση πιθανότητας:

$$\mathbb{P}(X = x) = \mathbb{P}\left(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k\right) = \frac{\binom{r_1}{x_1}\binom{r_2}{x_2}\cdots\binom{r_k}{x_k}}{\binom{r_1+r_2+\cdots+r_k}{m}}.$$

```
n = 10000
m = 50
r = c(30, 10, 40, 20)
k = length(r)
X = matrix(0, n, k)
for (i in 1:n) {
    count = r
    total = sum(r)
    for (j in 1:m) {
        pmf = count/total
        U = runif(1)
        l = 1
        cdf = pmf[1]
        while (U > cdf) {
            l = l + 1
            cdf = cdf + pmf[l]
        }
        X[i, l] = X[i, l] + 1
        count[l] = count[l] - 1
        total = total - 1
    }
```

```
}
colMeans(X)
```

```
## [1] 15.0137  4.9887 20.0386  9.9590
```

Μέθοδος Σύνθεσης

Έστω τυχαία μεταβλητή X με συνάρτηση κατανομής $F_X(x)$ και απόλυτα συνεχής τυχαία μεταβλητή Y με συνάρτηση πυκνότητας πιθανότητας $f_Y(y)$. Γνωρίζουμε ότι:

$$F_X(x) = \mathbb{P}(X \leqslant x) = \int_{\mathbb{R}} \mathbb{P}(X \leqslant x \mid Y = y) f_Y(y) dy = \int_{\mathbb{R}} F_{X|Y}(x \mid y) f_Y(y) dy.$$

Παράδειγμα 1.27. Για $x \in [0, 1]$, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση κατανομής:

$$F(x) = \int_0^{\infty} x^y e^{-y} dy.$$

Θεωρούμε τη συνάρτηση πυκνότητας πιθανότητας $g(y) = e^{-y}$ για $y > 0$, δηλαδή θεωρούμε μία τυχαία μεταβλητή $Y \sim \mathrm{Exp}(1)$. Για $x \in [0, 1]$, υπολογίζουμε ότι:

$$F(x) = \mathbb{P}(X \leqslant x) = \int_0^{\infty} \mathbb{P}\left(X \leqslant x \mid Y = y\right) g(y) dy,$$

το οποίο συνεπάγεται ότι $F_{X|Y}(x \mid y) = \mathbb{P}\left(X \leqslant x \mid Y = y\right) = x^y$ και $F^{-1}_{X|Y}(u \mid y) = u^{1/y}$.

Είσοδος: Συναρτήσεις κατανομής F_Y, $F_{X|Y}$ και ζητούμενο μέγεθος δείγματος n.

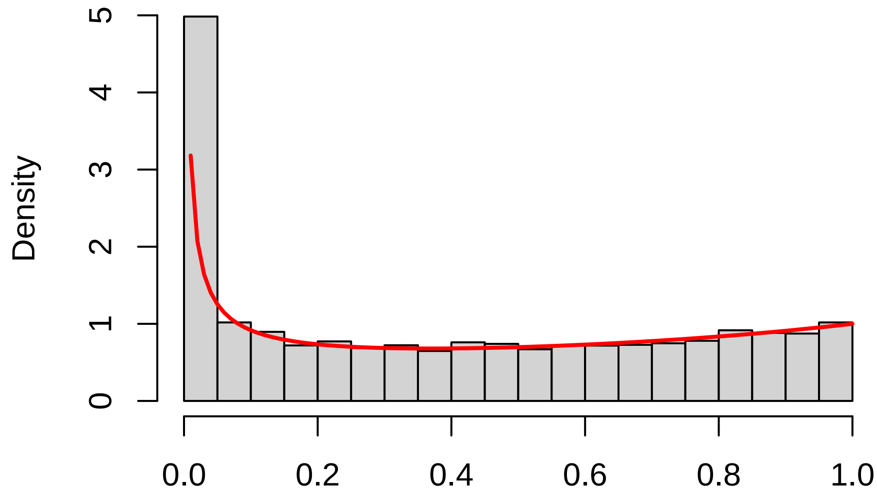
Για $i = 1, 2, \dots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

1: Προσομοιώνουμε $U \sim \mathrm{Unif}[0, 1]$ και παίρνουμε $Y = F_Y^{-1}(U)$.

2: Προσομοιώνουμε $V \sim \mathrm{Unif}[0, 1]$ και παίρνουμε $X_i = F^{-}_{X|Y}(V \mid Y)$.

Έξοδος: Τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση κατανομής F_X.

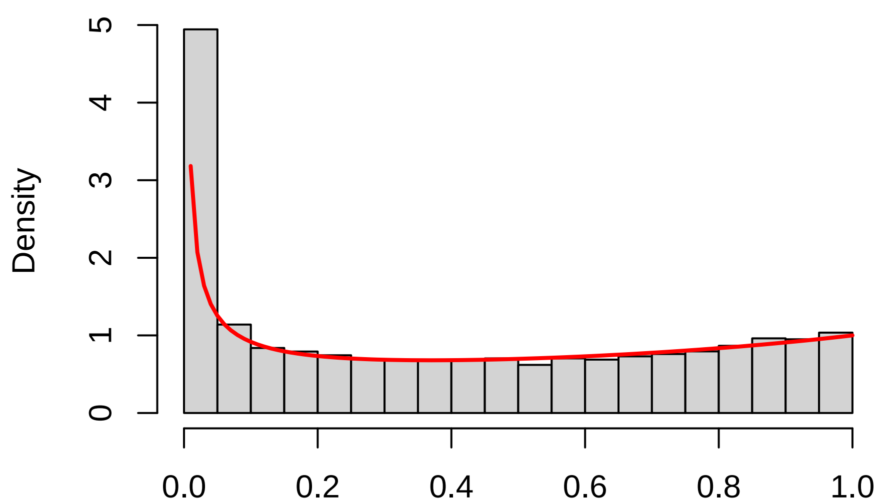
```
n = 10000
U = runif(n)
Y = -log(U)
V = runif(n)
X = V^(1/Y)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(1/(x * (1 - log(x))^2), add = TRUE, col = "red", lwd = 2)
```



Για $x \in [0,1]$, μπορούμε απευθείας να υπολογίσουμε ότι:

$$F(x) = \int_0^{\infty} \left(\frac{e}{x}\right)^{-y} dy = \left[-\frac{1}{\log\frac{e}{x}} \left(\frac{e}{x}\right)^{-y}\right]_{y=0}^{\infty} = \frac{1}{1-\log x}, \quad F^{-1}(u) = e^{1-1/u}, \quad f(x) = \frac{1}{x\,(1-\log x)^2}.$$

```
n = 10000
U = runif(n)
X = exp(1 - 1/U)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(1/(x * (1 - log(x))^2), add = TRUE, col = "red", lwd = 2)
```



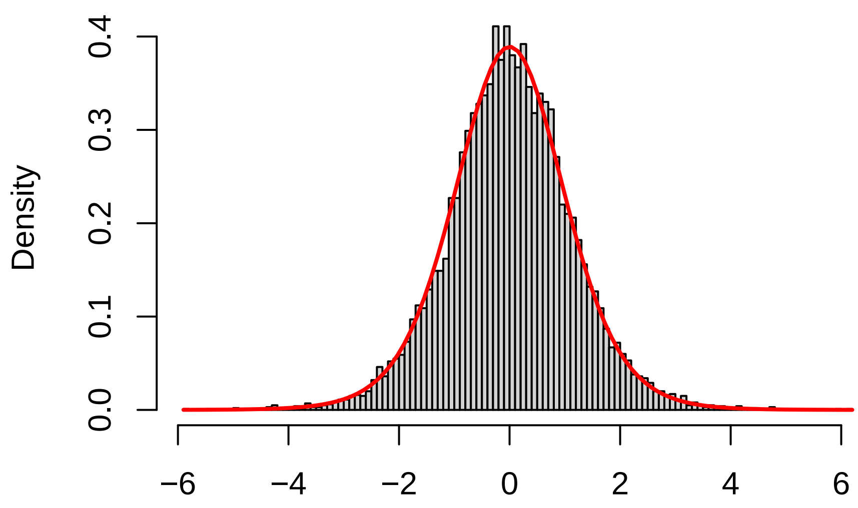
Παράδειγμα 1.28. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n \sim t_\nu$. Έστω ανεξάρτητες τυχαίες μεταβλητές $Z \sim \mathcal{N}(0,1)$ και $Y \sim \chi^2(\nu) \equiv \text{Gamma}\left(\frac{\nu}{2}, \frac{1}{2}\right)$. Τότε, γνωρίζουμε ότι:

$$X = \frac{Z}{\sqrt{Y/\nu}} \sim t_\nu.$$

Παρατηρούμε ότι:

$$(X \mid Y = y) = \frac{Z}{\sqrt{y/\nu}} \sim \mathcal{N}\left(0, \frac{\nu}{y}\right).$$

```r
n = 10000
nu = 10
M = (nu/2)^(nu/2)/gamma(nu/2) * exp(-(nu - 2)/2)
Y = numeric(n)
for (i in 1:n) {
    W = runif(1)
    Y[i] = -nu * log(W)
    U = runif(1)
    V = M * dexp(Y[i], 1/nu) * U
    while (dchisq(Y[i], nu) < V) {
        W = runif(1)
        Y[i] = -nu * log(W)
        U = runif(1)
        V = M * dexp(Y[i], 1/nu) * U
    }
}
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = sqrt(nu/Y) * Z
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dt(x, nu), add = TRUE, col = "red", lwd = 2)
```

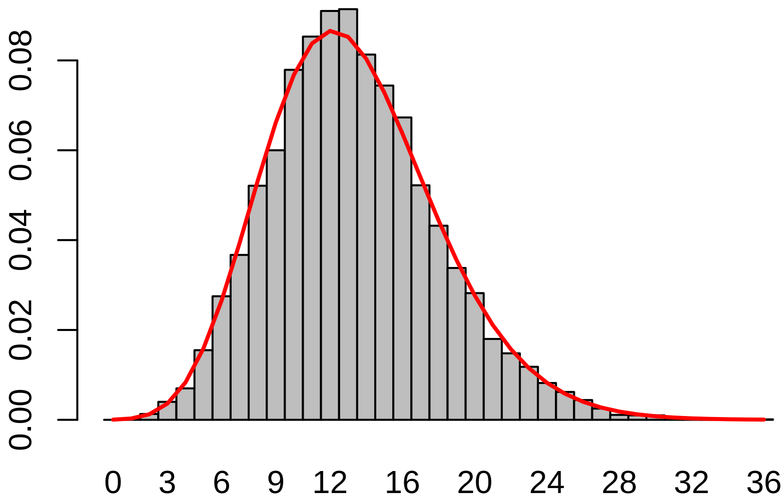


Παράδειγμα 1.29. Έστω διαδικασία Poisson $\{N(t) : t \geqslant 0\}$ ρυθμού 1 η οποία καταμετρά το πλήθος των ρίψεων ενός νομίσματος μέχρι τη χρονική στιγμή t. Για $p \in [0, 1]$, το νόμισμα φέρνει “Κ” με πιθανότητα p και “Γ” με πιθανότητα $1 - p$. Θεωρούμε τις διαδικασίες Poisson $\{N_{\mathrm{K}}(t) : t \geqslant 0\}$ ρυθμού p και $\{N_{\Gamma}(t) : t \geqslant 0\}$ ρυθμού $1 - p$, οι οποίες καταμετρούν το πλήθος των “Κ” και των “Γ” αντίστοιχα μέχρι τη χρονική στιγμή t. Γνωρίζουμε ότι οι διαδικασίες $\{N_{\mathrm{K}}(t) : t \geqslant 0\}$ και $\{N_{\Gamma}(t) : t \geqslant 0\}$ είναι ανεξάρτητες και αποτελούν μία διάσπαση της διαδικασίας $\{N(t) : t \geqslant 0\}$. Το πλήθος των γεγονότων στη διαδικασία $\{N_{\Gamma}(t) : t \geqslant 0\}$ μέχρι το

k-οστό γεγονός στη διαδικασία $\{N_{\mathrm{K}}(t) : t \geqslant 0\}$ γνωρίζουμε ότι ακολουθεί την κατανομή $\mathrm{NegBin}(k, p)$. Όμως, το πλήθος $N_\Gamma(t)$ των "Γ" μέχρι τη χρονική στιγμή t ακολουθεί την κατανομή Poisson $((1-p)t)$ και ο χρόνος S_k μέχρι το k-οστό "Κ" ακολουθεί την κατανομή $\mathrm{Gamma}(k, p)$. Για $j = 0, 1, \ldots$, επαληθεύουμε ότι:

$$
\begin{aligned}
\mathbb{P}\left(N_\Gamma\left(S_k\right) = j\right) &= \int_0^\infty \mathbb{P}\left(N_\Gamma\left(S_k\right) = j \mid S_k = y\right) f_{S_k}(y) dy = \int_0^\infty \mathbb{P}\left(N_\Gamma\left(y\right) = j \mid S_k = y\right) f_{S_k}(y) dy \\
&= \int_0^\infty \mathbb{P}\left(N_\Gamma\left(y\right) = j\right) f_{S_k}(y) dy = \int_0^\infty e^{-(1-p)y} \frac{[(1-p)y]^j}{j!} \frac{p^k}{(k-1)!} y^{k-1} e^{-py} dy \\
&= \frac{1}{j!(k-1)!} p^k (1-p)^j \int_0^\infty y^{j+k-1} e^{-y} dy = \frac{1}{j!(k-1)!} p^k (1-p)^j \cdot \frac{(j+k-1)!}{1^{j+k}} \\
&= \binom{j+k-1}{j} p^k (1-p)^j.
\end{aligned}
$$

```r
n = 10000
k = 20
p = 0.6
U = matrix(runif(n * k), n)
R = -log(U)/p
Y = rowSums(R)
X = numeric(n)
for (i in 1:n) {
    S = 0
    while (S <= Y[i]) {
        U = runif(1)
        R = -log(U)/(1 - p)
        S = S + R
        if (S <= Y[i]) {
            X[i] = X[i] + 1
        }
    }
}
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, dnbinom(0:max(X), k, p), col = "red", lwd = 2)
```



Έστω τυχαία μεταβλητή X με συνάρτηση κατανομής $F_X(x)$ και διακριτή τυχαία μεταβλητή Y με συνάρτηση πιθανότητας $w = [w_j]$. Επιπλέον, ορίζουμε τις δεσμευμένες συναρτήσεις κατανομής $F_j(x) = F_{X|Y}(x \mid j)$ για $j = 0, 1, \ldots$. Γνωρίζουμε ότι:

$$F_X(x) = \mathbb{P}(X \leqslant x) = \sum_{j=0}^{\infty} \mathbb{P}(Y = j)\mathbb{P}(X \leqslant x \mid Y = j) = \sum_{j=0}^{\infty} w_j F_{X|Y}(x \mid j) = \sum_{j=0}^{\infty} w_j F_j(x).$$

Η F_X καλείται μίξη κατανομών.

Σημείωση 1.12. i. Αν οι $F_0, F_1, \ldots$ είναι απόλυτα συνεχείς συναρτήσεις κατανομής με αντίστοιχες συναρτήσεις πυκνότητας πιθανότητας $f_0, f_1, \ldots$, τότε η F είναι απόλυτα συνεχής συνάρτηση κατανομής με αντίστοιχη συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \sum_{j=0}^{\infty} w_j f_j(x).$$

ii. Αν οι $F_0, F_1, \ldots$ είναι βαθμωτές συναρτήσεις κατανομής με αντίστοιχες συναρτήσεις πιθανότητας $p^{(0)}, p^{(1)}, \ldots$, τότε η F είναι βαθμωτή συνάρτηση κατανομής με αντίστοιχη συνάρτηση πιθανότητας:

$$p_\ell = \sum_{j=0}^{\infty} w_j p_\ell^{(j)}.$$

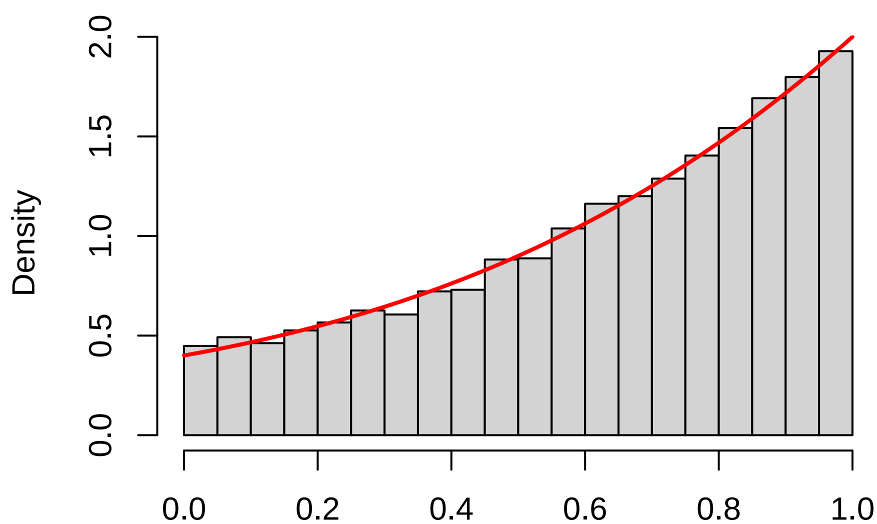
Παράδειγμα 1.30. Για $x \in [0, 1]$, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ με συνάρτηση κατανομής:

$$F(x) = \sum_{j=1}^{k} w_j x^j,$$

όπου $w = [w_j]$ συνάρτηση πιθανότητας. Παρατηρούμε ότι οι συναρτήσεις κατανομής $F_j(x) = x^j$ αντιστοιχούν στις κατανομές $\text{Beta}(j, 1)$ και υπολογίζουμε ότι $F_j^{-1}(u) = u^{1/j}$.

```
n = 10000
S = 1:4
w = c(0.4, 0.3, 0.2, 0.1)
```

```
Y = numeric(n)
for (i in 1:n) {
    U = runif(1)
    j = 1
    cdf = w[1]
    while (U > cdf) {
        j = j + 1
        cdf = cdf + w[j]
    }
    Y[i] = S[j]
}
U = runif(n)
X = U^(1/Y)
hist(X, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = NA)
curve(w[1] * dbeta(x, 1, 1) + w[2] * dbeta(x, 2, 1) + w[3] * dbeta(x, 3, 1) +
    w[4] * dbeta(x, 4, 1), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.31. Για $j = 0, 1, ...,$ θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ με συνάρτηση πιθανότητας:

$$p_j = \frac{1}{2^{j+2}} + \frac{1}{3^{j+1}}.$$

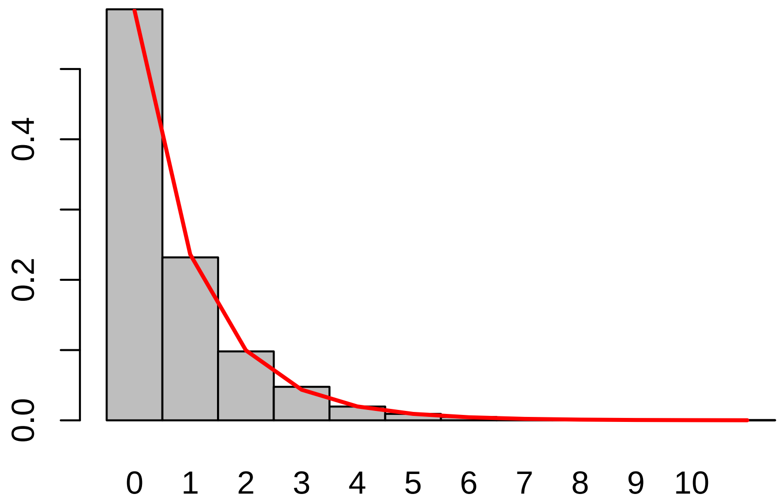
Παρατηρούμε ότι:

$$p_j = \underbrace{\frac{1}{2}}_{w_1} \cdot \underbrace{\left(\frac{1}{2}\right)^j \frac{1}{2}}_{p_j^{(1)}} + \underbrace{\frac{1}{2}}_{w_2} \cdot \underbrace{\left(\frac{1}{3}\right)^j \frac{2}{3}}_{p_j^{(2)}}.$$

Επιπλέον, παρατηρούμε ότι η συνάρτηση πιθανότητας $p^{(1)}$ αντιστοιχεί στη γεωμετρική κατανομή με πιθανότητα επιτυχίας $\frac{1}{2}$, ενώ η συνάρτηση πιθανότητας $p^{(2)}$ αντιστοιχεί στη γεωμετρική κατανομή με πιθανότητα επιτυχίας $\frac{2}{3}$.

```
n = 10000
w = c(0.5, 0.5)
```

```
p = c(0.5, 2/3)
U = runif(n)
Y = ifelse(U < w[1], 1, 2)
V = runif(n)
X = floor(log(V)/log(1 - p[Y]))
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0)
lines(0:max(X) + 0.5, w[1] * dgeom(0:max(X), p[1]) + w[2] * dgeom(0:max(X),
    p[2]), col = "red", lwd = 2)
```



Λήμμα 1.5. Αν $X \sim \text{Exp}(\lambda)$ και $\mu \in \mathbb{R}$, τότε η τυχαία μεταβλητή $W_1 = \mu - X$ έχει συνάρτηση πυκνότητας πιθανότητας $f_{W_1}(x) = \lambda e^{-\lambda(\mu - x)}$ για $x < \mu$.

Απόδειξη. Για $x < \mu$, υπολογίζουμε ότι:

$$F_{W_1}(x) = \mathbb{P}\left(W_1 \leqslant x\right) = \mathbb{P}\left(X \geqslant \mu - x\right) = e^{-\lambda(\mu - x)}, \quad f_{W_1}(x) = \frac{\partial F_{W_1}(x)}{\partial x} = \lambda e^{-\lambda(\mu - x)}.$$

□

Σημείωση 1.13. Έχουμε δείξει ότι η τυχαία μεταβλητή $W_2 = \mu + X$ έχει συνάρτηση πυκνότητας πιθανότητας $f_{W_2}(x) = \lambda e^{-\lambda(x - \mu)}$ για $x \geqslant \mu$.

Παράδειγμα 1.32. Για $x \in \mathbb{R}$, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ με συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \frac{\lambda}{2} e^{-\lambda|x - \mu|}.$$

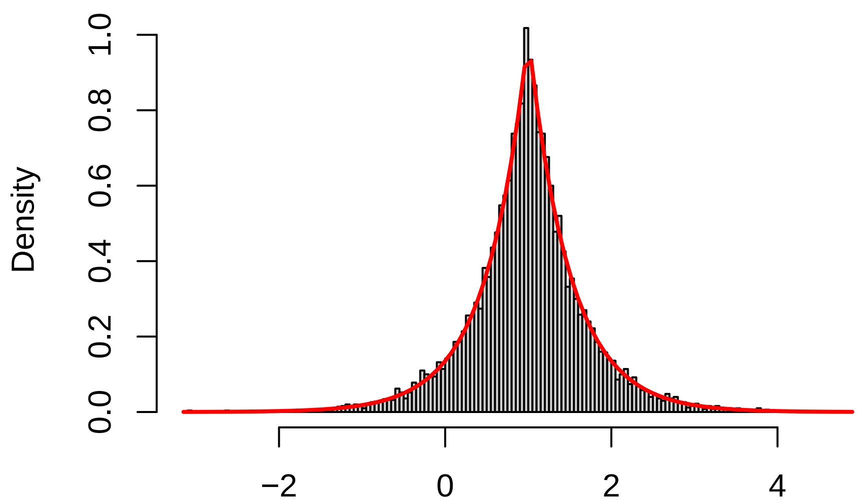
Παρατηρούμε ότι:

$$f(x) = \begin{cases} \frac{1}{2}\lambda e^{-\lambda(\mu - x)}, & x < \mu \\ \frac{1}{2}\lambda e^{-\lambda(x - \mu)}, & x \geqslant \mu \end{cases}.$$

Επομένως, παίρνουμε ότι:

$$f(x) = \frac{1}{2} f_{W_1}(x) + \frac{1}{2} f_{W_2}(x).$$

```
n = 10000
lambda = 2
mu = 1
U = runif(n)
Y = ifelse(U < 0.5, 1, 2)
V = runif(n)
X = ifelse(Y == 1, mu + log(V)/lambda, mu - log(V)/lambda)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(abs(x - mu), lambda)/2, add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.33. Θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \dots, X_n$ με συνάρτηση κατανομής:

$$F(x) = \begin{cases} \frac{1-e^{-2x}+2x}{3}, & 0 \leqslant x \leqslant 1 \\ \frac{3-e^{-2x}}{3}, & x > 1 \end{cases}.$$

Για $x \geqslant 0$, θεωρούμε τις συναρτήσεις κατανομής:

$$F_1(x) = 1 - e^{-2x}, \quad F_2(x) = \begin{cases} x, & 0 \leqslant x \leqslant 1 \\ 1, & x > 1 \end{cases}.$$

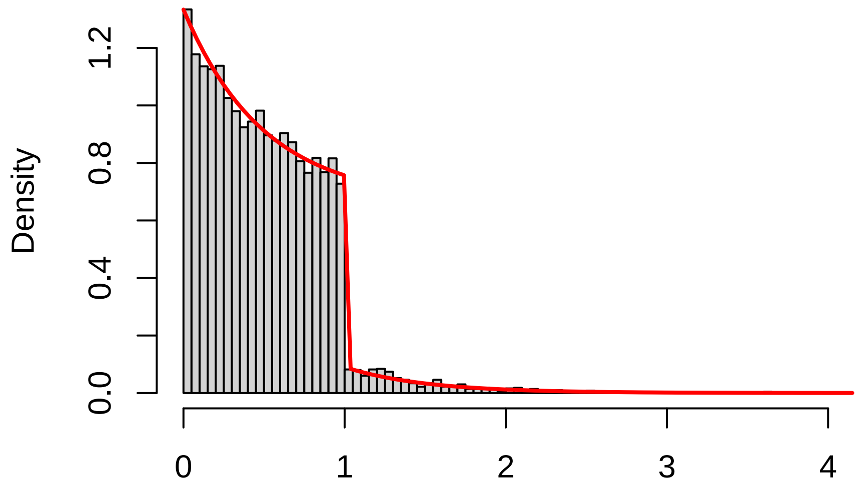
Παρατηρούμε ότι:

$$F(x) = \frac{1}{3}F_1(x) + \frac{2}{3}F_2(x).$$

Επιπλέον, παρατηρούμε ότι η F_1 είναι η συνάρτηση κατανομής της εκθετικής κατανομής με παράμετρο 2, ενώ η F_2 είναι η συνάρτηση κατανομής της ομοιόμορφης κατανομής στο διάστημα $[0, 1]$.

```
n = 10000
w = c(1/3, 2/3)
lambda = 2
U = runif(n)
Y = ifelse(U < w[1], 1, 2)
V = runif(n)
```

```
X = ifelse(Y == 1, -log(V)/lambda, V)
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(w[1] * dexp(x, lambda) + w[2] * dunif(x), add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 1.34. Για $x \geqslant 0$, θέλουμε να παράγουμε τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ με συνάρτηση κατανομής:

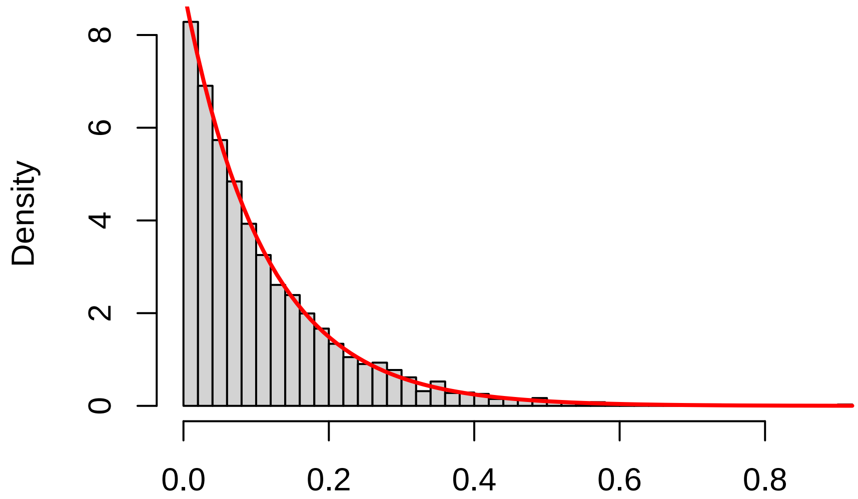
$$F(x) = \frac{2 - e^{-9x}}{2}.$$

Θεωρούμε τις συναρτήσεις κατανομής $F_1(x) = 1 - e^{-9x}$ και $F_2(x) = 1$. Παρατηρούμε ότι:

$$F(x) = \frac{1}{2}F_1(x) + \frac{1}{2}F_2(x).$$

Επιπλέον, παρατηρούμε ότι η F_1 είναι η συνάρτηση κατανομής της εκθετικής κατανομής με παράμετρο 9, ενώ η F_2 είναι η συνάρτηση κατανομής της εκφυλισμένης τυχαίας μεταβλητής $Y = 0$.

```
n = 10000
w = c(0.5, 0.5)
lambda = 9
U = runif(n)
Y = ifelse(U < w[1], 1, 2)
V = runif(n)
X = ifelse(Y == 1, -log(V)/lambda, 0)
hist(X[Y == 1], "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(x, lambda), add = TRUE, col = "red", lwd = 2)
```



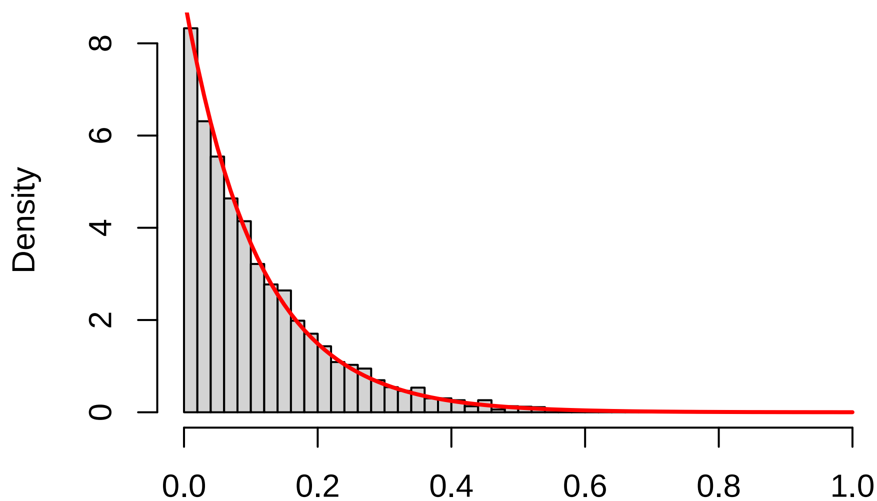
Παρατηρούμε ότι $F(0) = 0.5$. Για $u \in [0, 0.5]$, έπεται ότι $F^{-}(u) = 0$. Για $u \in (0.5, 1]$, υπολογίζουμε ότι:

$$F(x) = u \Leftrightarrow x = -\frac{1}{9} \log\left[2(1-u)\right].$$

Επομένως, παίρνουμε ότι:

$$F^{-}(u) = \begin{cases} 0, & 0 \leqslant u \leqslant 0.5 \\ -\frac{1}{9}\log\left[2(1-u)\right], & 0.5 < u \leqslant 1 \end{cases}.$$

```
n = 10000
U = runif(n)
X = ifelse(U <= 0.5, 0, -log(2 * (1 - U))/lambda)
hist(X[U > 0.5], "FD", freq = FALSE, main = NA, xlab = NA)
curve(dexp(x, lambda), add = TRUE, col = "red", lwd = 2)
```



2 Μέθοδος Monte Carlo

Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_0^1 g(x)dx.$$

Γνωρίζουμε ότι η τυχαία μεταβλητή $U \sim \text{Unif}[0,1]$ έχει συνάρτηση πυκνότητας πιθανότητας $f(x) = 1$ για $x \in [0,1]$. Επομένως, παρατηρούμε ότι:

$$I = \int_0^1 g(x)f(x)dx = \mathbb{E}\left[g(U)\right].$$

Έστω τυχαίο δείγμα $U_1, U_2, \ldots, U_n$ από την κατανομή $\text{Unif}[0,1]$. Σύμφωνα με τον ισχυρό νόμο των μεγάλων αριθμών, γνωρίζουμε ότι:

$$\frac{1}{n}\sum_{i=1}^{n} g(U_i) \xrightarrow{\text{a.s.}} \mathbb{E}\left[g(U)\right] = I.$$

Παράδειγμα 2.1. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_0^1 e^{e^x} dx.$$

```
n = 1e+05
U = runif(n)
I = mean(exp(exp(U)))
print(I)
```

```
## [1] 6.318484
```

Παράδειγμα 2.2. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_0^1 \int_0^1 e^{(x+y)^2} dxdy.$$

Θεωρούμε τις ανεξάρτητες τυχαίες μεταβλητές $U, V \sim \text{Unif}[0,1]$ με συνάρτηση πυκνότητας πιθανότητας $f_{U,V}(u,v) = f_U(u)f_V(v) = 1$ για $u, v \in [0,1]$. Τότε, παρατηρούμε ότι $I = \mathbb{E}\left[e^{(U+V)^2}\right]$.

```
n = 1e+05
U = runif(n)
V = runif(n)
I = mean(exp((U + V)^2))
print(I)
```

```
## [1] 4.886297
```

Γενικότερα, θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_S g(x)dx.$$

Θεωρούμε την τυχαία μεταβλητή X με συνάρτηση πυκνότητας πιθανότητας $f(x)$ και στήριγμα S. Αν θέσουμε $h(x) = \frac{g(x)}{f(x)}$, τότε παρατηρούμε ότι:

$$I = \int_S \frac{g(x)}{f(x)} \cdot f(x) dx = \mathbb{E}\left[h(X)\right].$$

Έστω τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας $f(x)$. Σύμφωνα με τον ισχυρό νόμο των μεγάλων αριθμών, γνωρίζουμε ότι:

$$\frac{1}{n}\sum_{i=1}^{n} h(X_i) \xrightarrow{\text{a.s.}} \mathbb{E}\left[h(X)\right] = I.$$

Παράδειγμα 2.3. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_{-2}^{2} e^{x+x^2} dx.$$

Θεωρούμε την τυχαία μεταβλητή $X \sim \text{Unif}[-2, 2]$ με συνάρτηση πυκνότητας πιθανότητας $f(x) = \frac{1}{4}$ για $x \in [-2, 2]$. Τότε, παρατηρούμε ότι:

$$I = \int_{-2}^{2} 4e^{x+x^2} \frac{1}{4} dx = \mathbb{E}\left(4e^{X+X^2}\right).$$

```
n = 1e+05
U = runif(n)
X = 4 * U - 2
I = mean(4 * exp(X + X^2))
print(I)
```

```
## [1] 93.76997
```

Παράδειγμα 2.4. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_{0}^{\infty} \frac{x}{(1+x^2)^2} dx.$$

Θεωρούμε την τυχαία μεταβλητή $X \sim \text{Exp}(1)$ με συνάρτηση πυκνότητας πιθανότητας $f(x) = e^{-x}$ για $x > 0$. Τότε, παρατηρούμε ότι:

$$I = \int_{0}^{\infty} \frac{xe^x}{(1+x^2)^2} e^{-x} dx = \mathbb{E}\left[\frac{Xe^X}{(1+X^2)^2}\right].$$

```
n = 1e+05
U = runif(n)
X = -log(U)
I = mean(X * exp(X)/(1 + X^2)^2)
print(I)
```

```
## [1] 0.4993482
```

Παράδειγμα 2.5. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_{-\infty}^{\infty} x^4 e^{-x^2} dx.$$

Θεωρούμε την τυχαία μεταβλητή $X \sim \mathcal{N}(0, 0.5)$ με συνάρτηση πυκνότητας πιθανότητας $f(x) = \frac{1}{\sqrt{\pi}} e^{-x^2}$ για $x \in \mathbb{R}$. Τότε, παρατηρούμε ότι:

$$I = \int_{-\infty}^{\infty} \sqrt{\pi} x^4 \frac{1}{\sqrt{\pi}} e^{-x^2} dx = \mathbb{E}\left(\sqrt{\pi} X^4\right).$$

```
n = 1e+05
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = Z/sqrt(2)
I = mean(sqrt(pi) * X^4)
print(I)
```

```
## [1] 1.328683
```

Παράδειγμα 2.6. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_{2}^{\infty} e^{-x^2/2} \sin(2\pi x) dx.$$

Θεωρούμε την τυχαία μεταβλητή X με συνάρτηση πυκνότητας πιθανότητας $f(x) = 4e^{-4(x-2)}$ για $x > 2$. Τότε, παρατηρούμε ότι:

$$I = \int_{2}^{\infty} \frac{1}{4} e^{-x^2/2+4x-8} \sin(2\pi x) \cdot 4e^{-4(x-2)} dx = \mathbb{E}\left[\frac{1}{4} e^{-(X-4)^2/2} \sin(2\pi X)\right].$$

```
n = 1e+05
U = runif(n)
X = 2 - log(U)/4
I = mean(exp(-(X - 4)^2/2) * sin(2 * pi * X)/4)
print(I)
```

```
## [1] 0.01963348
```

Παράδειγμα 2.7. Θέλουμε να προσεγγίσουμε το ολοκλήρωμα:

$$I = \int_{0}^{\infty} \int_{0}^{x} e^{-(x+y)} dy dx.$$

Θεωρούμε τις τυχαίες μεταβλητές X, Y με $X \sim \text{Exp}(1)$ και $(Y \mid X = x) \sim \text{Unif}[0, x]$. Για $x > 0$ και $y \in [0, x]$, παρατηρούμε ότι:

$$f_{X,Y}(x,y) = f_X(x) f_{Y|X}(y \mid x) = e^{-x}\frac{1}{x}.$$

Επομένως, υπολογίζουμε ότι

$$I = \int_0^{\infty}\int_0^{x} xe^{-y}\frac{e^{-x}}{x}dydx = \mathbb{E}\left(Xe^{-Y}\right).$$

```
n = 1e+05
U = runif(n)
X = -log(U)
V = runif(n)
Y = X * V
I = mean(X * exp(-Y))
print(I)
```

```
## [1] 0.4984331
```

Σημείωση 2.1. Γνωρίζουμε ότι $\mathbb{P}(A) = \mathbb{E}\left(\mathbb{1}_A\right)$.

Παράδειγμα 2.8. Θέλουμε να προσεγγίσουμε την τιμή της σταθεράς π. Θεωρούμε δύο ανεξάρτητες τυχαίες μεταβλητές $U, V \sim \text{Unif}[0,1]$. Τότε, υπολογίζουμε ότι:

$$\begin{aligned}
\mathbb{P}\left(U^2 + V^2 \leqslant 1\right) = \mathbb{P}\left(U \leqslant \sqrt{1-V^2}\right) &= \int_0^1 \int_0^{\sqrt{1-v^2}} 1dudv \\
&= \int_0^1 \sqrt{1-v^2}dv \overset{v=\sin x}{=} \int_0^{\pi/2} \sqrt{1-\sin^2 x}\cos x dx = \int_0^{\pi/2} \cos^2 x dx \\
&= \int_0^{\pi/2} \frac{1+\cos(2x)}{2}dx = \left[\frac{x}{2} + \frac{\sin(2x)}{4}\right]_{x=0}^{\pi/2} = \frac{\pi}{4}.
\end{aligned}$$

Επομένως, παρατηρούμε ότι:

$$\pi = 4\mathbb{P}\left(U^2 + V^2 \leqslant 1\right) = \mathbb{E}\left(4 \cdot \mathbb{1}_{\{U^2+V^2 \leqslant 1\}}\right).$$

```
n = 1e+06
U = runif(n)
V = runif(n)
mean(4 * (U^2 + V^2 <= 1))
```

```
## [1] 3.141728
```

Παράδειγμα 2.9. Έστω τυχαίο δείγμα $X_1, \ldots, X_k \sim \mathcal{N}\left(\mu, \sigma^2\right)$. Ορίζουμε:

$$\overline{X} = \frac{1}{k}\sum_{j=1}^{k} X_j, \quad S^2 = \frac{1}{k-1}\sum_{j=1}^{k}\left(X_j - \overline{X}\right)^2.$$

Γνωρίζουμε ότι το $100(1-\alpha)\%$ διάστημα εμπιστοσύνης ίσων ουρών για την παράμετρο μ ισούται με:

$$I(X) = \left[\overline{X} - t_{k-1;\alpha/2}\frac{S}{\sqrt{k}}, \overline{X} + t_{k-1;\alpha/2}\frac{S}{\sqrt{k}}\right].$$

Επιπλέον, γνωρίζουμε από κατασκευή ότι $\mathbb{P}[\mu \in I(X)] = 1-\alpha$. Θέλουμε να επαληθεύσουμε ότι το διάστημα $I(X)$ έχει κάλυψη $100(1-\alpha)\%$ για την παράμετρο μ. Θεωρούμε τυχαία δείγματα $X^{(1)}, \dots, X^{(n)}$ από την κατανομή $\mathcal{N}(\mu, \sigma^2)$ και κατασκευάζουμε τα αντίστοιχα διαστήματα εμπιστοσύνης $I^{(1)}, \dots, I^{(n)}$. Σύμφωνα με τον ισχυρό νόμο των μεγάλων αριθμών, θα πρέπει να ισχύει ότι:

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{\mu \in I^{(i)}\}} \xrightarrow{\text{a.s}} \mathbb{E}\left[\mathbb{1}_{\{\mu \in I(X)\}}\right] = \mathbb{P}[\mu \in I(X)] = 1-\alpha.$$

```
n = 10000
k = 10
mu = 1
sigma = 2
alpha = 0.05
U = matrix(runif(k * n/2), n/2)
D = -2 * log(U)
V = matrix(runif(k * n/2), n/2)
Theta = 2 * pi * V
Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
X = sigma * Z + mu
Xbar = rowMeans(X)
S = apply(X, 1, sd)
I = cbind(Xbar - qt(alpha/2, k - 1, lower.tail = FALSE) * S/sqrt(k), Xbar +
    qt(alpha/2, k - 1, lower.tail = FALSE) * S/sqrt(k))
100 * mean(I[, 1] <= mu & mu <= I[, 2])
```

```
## [1] 94.78
```

Παράδειγμα 2.10. Έστω τυχαίο δείγμα $X_1, \dots, X_k \sim \mathcal{N}(\mu, \sigma^2)$. Γνωρίζουμε ότι η στατιστική συνάρτηση του μονόπλευρου ελέγχου υποθέσεων $H_0 : \mu = \mu_0$ vs. $H_1 : \mu < \mu_0$ ισούται με:

$$T(X) = \frac{\overline{X} - \mu_0}{S/\sqrt{k}}.$$

Επιπλέον, γνωρίζουμε ότι $T(X) \sim t_{k-1}$ υπό την υπόθεση H_0. Ορίζουμε το p-value $p(X) = F_{t_{k-1}}(T(X))$. Απορρίπτουμε την H_0 σε επίπεδο στατιστικής σημαντικότητας α αν $T(X) < -t_{k-1;\alpha}$ ή $p(X) < \alpha$. Η πιθανότητα σφάλματος τύπου I ισούται με $\mathbb{P}_{\mu_0}\left[T(X) < -t_{k-1;\alpha}\right] = \alpha$ και η ισχύς με $\beta(\mu) = \mathbb{P}_{\mu}\left[T(X) < -t_{k-1;\alpha}\right]$. Θέλουμε να μελετήσουμε την κατανομή της στατιστικής συνάρτησης $T(X)$ και του p-value $p(X)$ κάτω από τις υποθέσεις H_0 και H_1. Θεωρούμε τυχαία δείγματα $X^{(1)}, \dots, X^{(n)}$ από την κατανομή $\mathcal{N}(\mu, \sigma^2)$ και πραγματοποιούμε τους αντίστοιχους ελέγχους υποθέσεων. Επιπλέον, θέλουμε να επαληθεύσουμε ότι ο έλεγχος έχει πιθανότητα σφάλματος τύπου I ίση με α ανεξαρτήτως των n, k, μ και σ. Τέλος, θέλουμε να μελετήσουμε τη μεταβολή της ισχύος του ελέγχου για διάφορες τιμές των k, μ, σ και α.

Πρόταση 2.1. Αν η τυχαία μεταβλητή X έχει απόλυτα συνεχή συνάρτηση κατανομής F, τότε ισχύει ότι $U = F(X) \sim \text{Unif}[0, 1]$.

Απόδειξη. Υπολογίζουμε ότι:

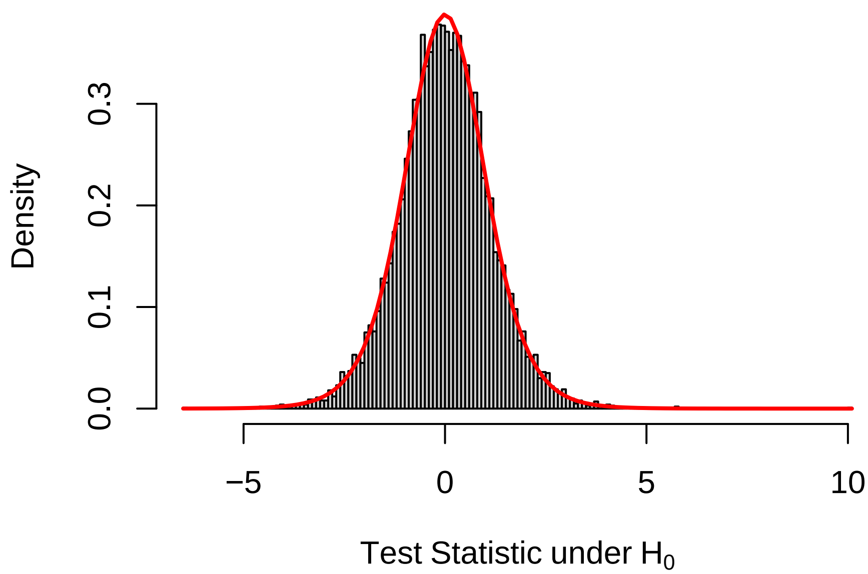
$$F_U(u) = \mathbb{P}\left[F(X) \leqslant u\right] = \mathbb{P}\left[X \leqslant F^{-1}(u)\right] = F\left(F^{-1}(u)\right) = u.$$

□

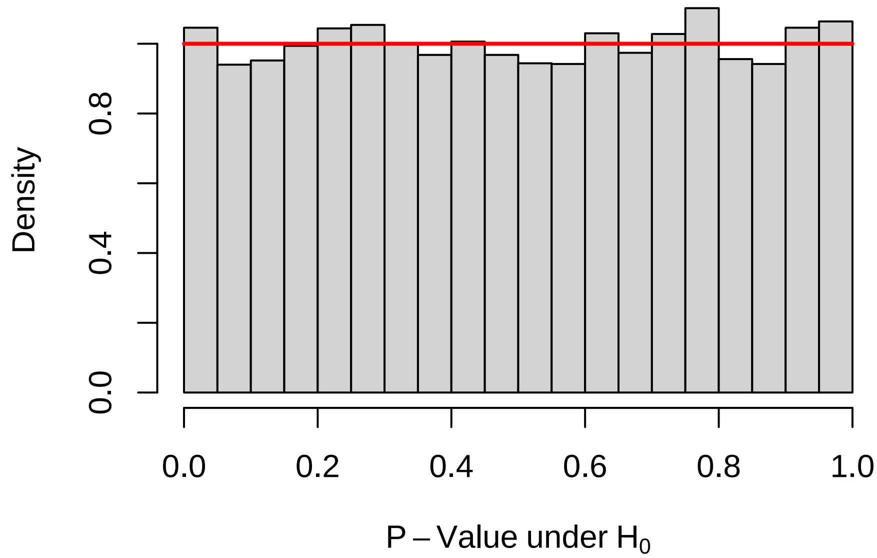
Πόρισμα 2.1. Ισχύει ότι $p(X) \sim \text{Unif}[0, 1]$ κάτω από την υπόθεση H_0.

Απόδειξη. Η στατιστική συνάρτηση $T(X)$ ακολουθεί την κατανομή t_{k-1} κάτω από την υπόθεση H_0. Επομένως, παίρνουμε ότι $p(X) = F_{t_{k-1}}(T(X)) \sim \text{Unif}[0, 1]$. □

```
n = 10000
k = 10
mu = 1
sigma = 2
mu0 = 1
alpha = 0.05
U = matrix(runif(k * n/2), n/2)
D = -2 * log(U)
V = matrix(runif(k * n/2), n/2)
Theta = 2 * pi * V
Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
X = sigma * Z + mu
Xbar = rowMeans(X)
S = apply(X, 1, sd)
t = (Xbar - mu0) * sqrt(k)/S
hist(t, "FD", freq = FALSE, main = NA, xlab = expression(Test ~ Statistic ~
    under ~ H[0]))
curve(dt(x, k - 1), add = TRUE, col = "red", lwd = 2)
```



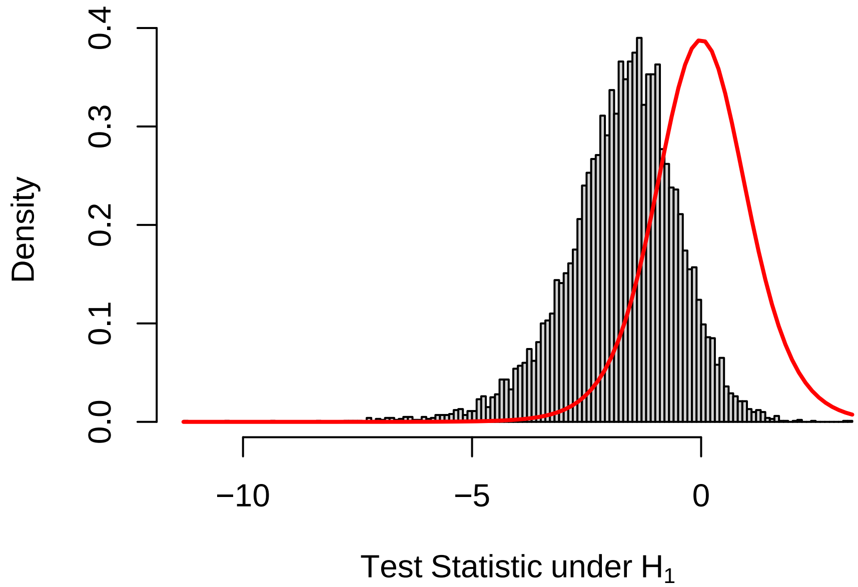
```r
p = pt(t, k - 1)
hist(p, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = expression(P -
    Value ~ under ~ H[0]))
curve(dunif(x), add = TRUE, col = "red", lwd = 2)
```



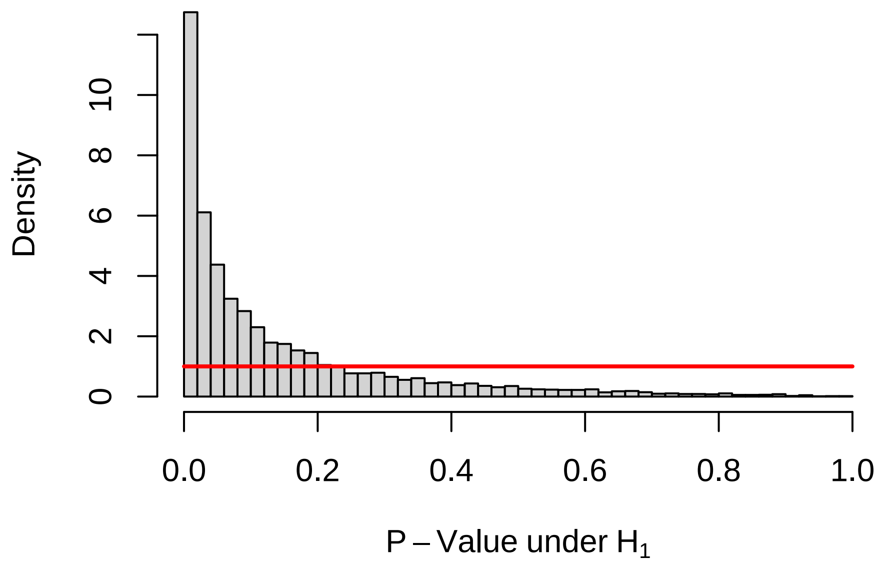

```r
mean(t < qt(alpha, k - 1))
```

```
## [1] 0.0523
```

```r
n = 10000
k = 10
mu = 0
sigma = 2
mu0 = 1
alpha = 0.05
U = matrix(runif(k * n/2), n/2)
D = -2 * log(U)
V = matrix(runif(k * n/2), n/2)
Theta = 2 * pi * V
Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
X = sigma * Z + mu
Xbar = rowMeans(X)
S = apply(X, 1, sd)
t = (Xbar - mu0) * sqrt(k)/S
hist(t, "FD", freq = FALSE, main = NA, xlab = expression(Test ~ Statistic ~
    under ~ H[1]))
curve(dt(x, k - 1), add = TRUE, col = "red", lwd = 2)
```




```
p = pt(t, k - 1)
hist(p, "FD", freq = FALSE, main = NA, xlim = c(0, 1), xlab = expression(P -
    Value ~ under ~ H[1]))
curve(dunif(x), add = TRUE, col = "red", lwd = 2)
```

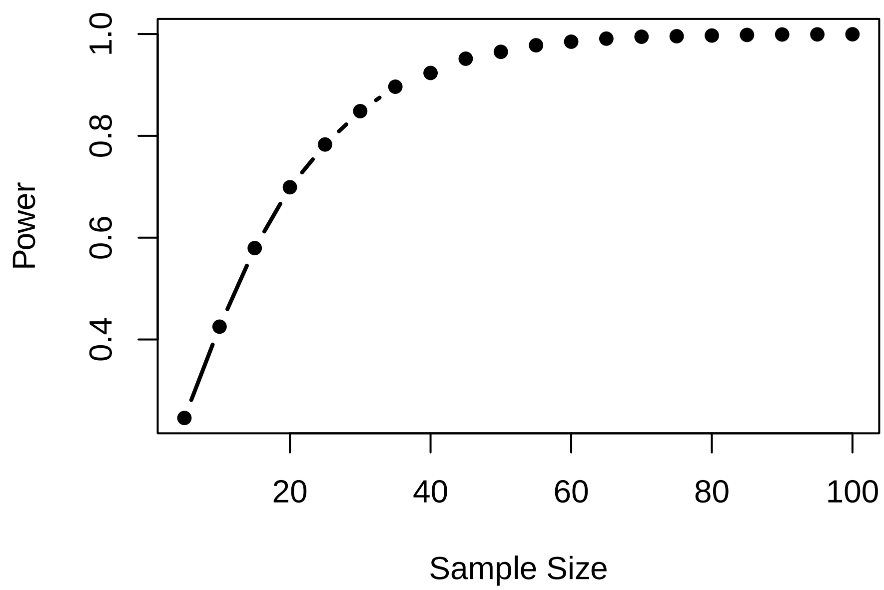



```
beta = mean(t < qt(alpha, k - 1))
print(beta)
```

```
## [1] 0.4231
```

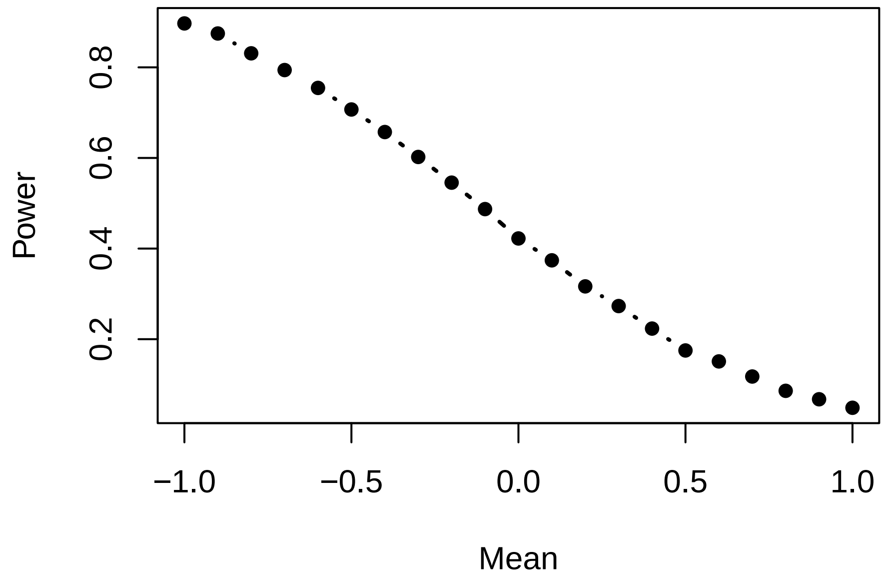
```
n = 10000
k = seq(5, 100, 5)
mu = 0
sigma = 2
mu0 = 1
alpha = 0.05
```

```r
beta = numeric(length(k))
for (j in 1:length(k)) {
    U = matrix(runif(k[j] * n/2), n/2)
    D = -2 * log(U)
    V = matrix(runif(k[j] * n/2), n/2)
    Theta = 2 * pi * V
    Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
    X = sigma * Z + mu
    Xbar = rowMeans(X)
    S = apply(X, 1, sd)
    t = (Xbar - mu0) * sqrt(k[j])/S
    beta[j] = mean(t < qt(alpha, k[j] - 1))
}
plot(k, beta, "b", xlab = "Sample Size", ylab = "Power", pch = 16, lwd = 2)
```

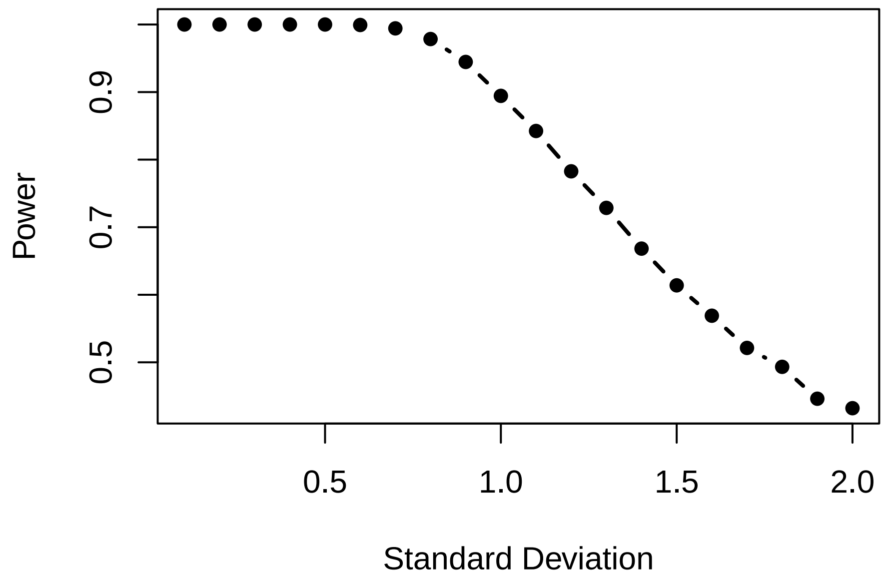



```r
n = 10000
k = 10
mu = seq(-1, 1, 0.1)
sigma = 2
mu0 = 1
alpha = 0.05
beta = numeric(length(mu))
for (j in 1:length(mu)) {
    U = matrix(runif(k * n/2), n/2)
    D = -2 * log(U)
    V = matrix(runif(k * n/2), n/2)
    Theta = 2 * pi * V
    Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
    X = sigma * Z + mu[j]
```

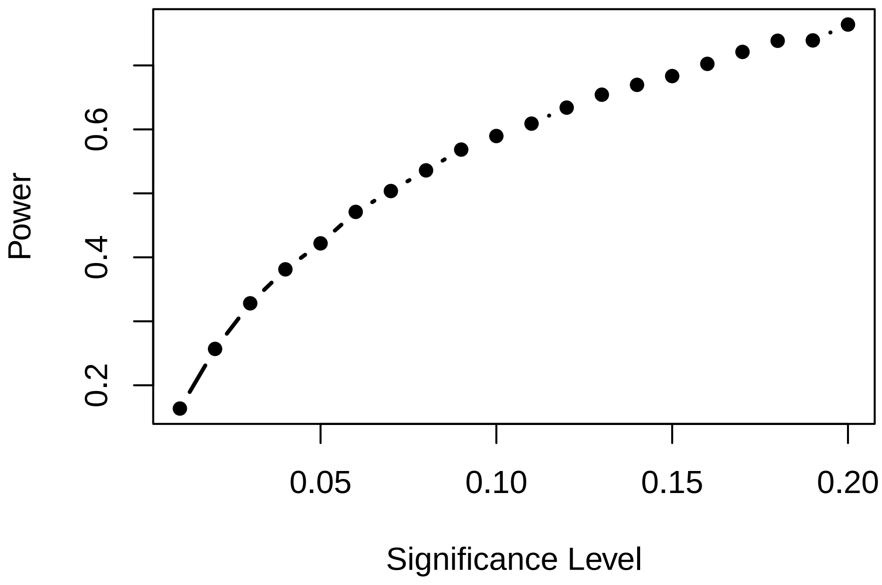
```
    Xbar = rowMeans(X)
    S = apply(X, 1, sd)
    t = (Xbar - mu0) * sqrt(k)/S
    beta[j] = mean(t < qt(alpha, k - 1))
}
plot(mu, beta, "b", xlab = "Mean", ylab = "Power", pch = 16, lwd = 2)
```




```
n = 10000
k = 10
mu = 0
sigma = seq(0.1, 2, 0.1)
mu0 = 1
alpha = 0.05
beta = numeric(length(sigma))
for (j in 1:length(sigma)) {
    U = matrix(runif(k * n/2), n/2)
    D = -2 * log(U)
    V = matrix(runif(k * n/2), n/2)
    Theta = 2 * pi * V
    Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
    X = sigma[j] * Z + mu
    Xbar = rowMeans(X)
    S = apply(X, 1, sd)
    t = (Xbar - mu0) * sqrt(k)/S
    beta[j] = mean(t < qt(alpha, k - 1))
}
plot(sigma, beta, "b", xlab = "Standard Deviation", ylab = "Power", pch = 16,
    lwd = 2)
```




```
n = 10000
k = 10
mu = 0
sigma = 2
mu0 = 1
alpha = seq(0.01, 0.2, 0.01)
beta = numeric(length(alpha))
for (j in 1:length(alpha)) {
    U = matrix(runif(k * n/2), n/2)
    D = -2 * log(U)
    V = matrix(runif(k * n/2), n/2)
    Theta = 2 * pi * V
    Z = rbind(sqrt(D) * cos(Theta), sqrt(D) * sin(Theta))
    X = sigma * Z + mu
    Xbar = rowMeans(X)
    S = apply(X, 1, sd)
    t = (Xbar - mu0) * sqrt(k)/S
    beta[j] = mean(t < qt(alpha[j], k - 1))
}
plot(alpha, beta, "b", xlab = "Significance Level", ylab = "Power", pch = 16,
    lwd = 2)
```



Θέλουμε να προσεγγίσουμε το άθροισμα της σειράς:

$$S = \sum_{j=0}^{\infty} a_j.$$

Θεωρούμε την τυχαία μεταβλητή X με συνάρτηση πιθανότητας $p = \left[p_j\right]$. Αν θέσουμε $b_j = \frac{a_j}{p_j}$, τότε παρατηρούμε ότι:

$$S = \sum_{j=0}^{\infty} p_j \frac{a_j}{p_j} = \mathbb{E}\left(b_X\right).$$

Έστω τυχαίο δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πιθανότητας $p = \left[p_j\right]$. Σύμφωνα με τον ισχυρό νόμο των μεγάλων αριθμών, γνωρίζουμε ότι:

$$\frac{1}{n} \sum_{i=1}^{n} b_{X_i} \overset{\text{a.s.}}{\to} \mathbb{E}\left(b_X\right) = S.$$

Παράδειγμα 2.11. Θέλουμε να προσεγγίσουμε το άθροισμα της σειράς:

$$S = \sum_{j=0}^{\infty} \frac{e^{-j^2}}{j!}.$$

Θεωρούμε μία τυχαία μεταβλητή $X \sim \text{Poisson}(1)$ με συνάρτηση πιθανότητας $p_j = e^{-1}/j!$ για $j = 0, 1, \ldots$. Τότε, παρατηρούμε ότι:

$$S = \sum_{j=0}^{\infty} \frac{e^{-1}}{j!} e^{1-j^2} = \mathbb{E}\left(e^{1-X^2}\right).$$

```
n = 1e+05
X = numeric(n)
for (i in 1:n) {
    U = runif(1)
    pmf = exp(-1)
```

```
    cdf = pmf
    while (U > cdf) {
        X[i] = X[i] + 1
        pmf = pmf/X[i]
        cdf = cdf + pmf
    }
}
I = mean(exp(1 - X^2))
print(I)
```

```
## [1] 1.37738
```

Λήμμα 2.1. Έστω μη-αρνητική και διακριτή τυχαία μεταβλητή X. Τότε, ισχύει ότι:

$$\mathbb{E}(X) = \sum_{k=0}^{\infty} \mathbb{P}(X > k).$$

Απόδειξη. Παρατηρούμε ότι:

$$\mathbb{E}(X) = \sum_{j=0}^{\infty} j\mathbb{P}(X = j) = \sum_{j=0}^{\infty} \sum_{k=0}^{j-1} \mathbb{P}(X = j) = \sum_{k=0}^{\infty} \sum_{j=k+1}^{\infty} \mathbb{P}(X = j) = \sum_{k=0}^{\infty} \mathbb{P}(X > k).$$

□

Σημείωση 2.2. Γνωρίζουμε ότι $\mathrm{Var}(X) = \mathbb{E}\left[(X - \mathbb{E}(X))^2\right]$.

Παράδειγμα 2.12. Έστω ακολουθία μη-αρνητικών και απόλυτα συνεχών τυχαίων μεταβλητών $X_1, X_2, \ldots$. Θέλουμε να εκτιμήσουμε τη μέση τιμή και τη διασπορά της τυχαίας μεταβλητής:

$$N = \sup\left\{k \in \mathbb{N} : X_1 < X_2 < \cdots < X_{k-1}\right\}.$$

Για $k \in \mathbb{N}$, παρατηρούμε ότι:

$$\mathbb{P}(N > k) = \mathbb{P}(X_1 < X_2 < \cdots < X_k) = \frac{1}{k!}.$$

Επομένως, υπολογίζουμε ότι:

$$\mathbb{E}(N) = \sum_{k=0}^{\infty} \mathbb{P}(N > k) = \sum_{k=0}^{\infty} \frac{1}{k!} = e.$$

```
n = 1e+05
N = numeric(n)
for (i in 1:n) {
    Uold = runif(1)
    Unew = runif(1)
    N[i] = 2
    while (Uold < Unew) {
        Uold = Unew
```

```
        Unew = runif(1)
        N[i] = N[i] + 1
    }
}
I = mean(N)
print(I)
```

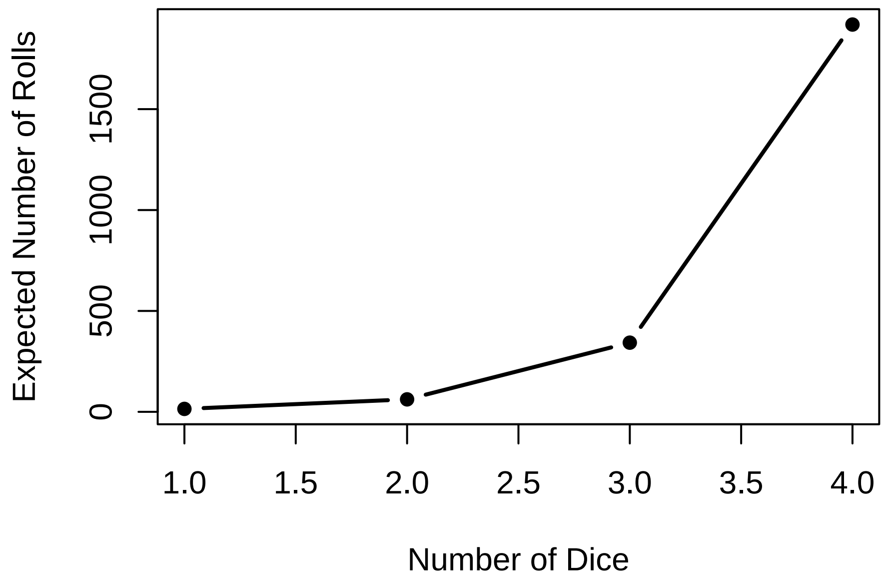
```
## [1] 2.71821
```

```
mean((N - I)^2)
```

```
## [1] 0.7690844
```

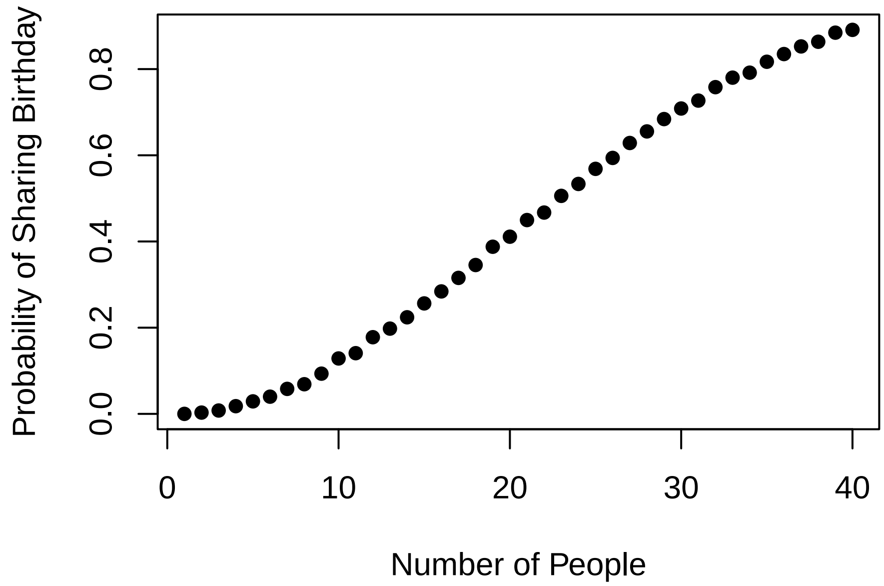
Παράδειγμα 2.13. Ρίχνουμε k δίκαια ζάρια και θέλουμε να εκτιμήσουμε το αναμενόμενο ελάχιστο πλήθος ρίψεων μέχρι να εμφανιστούν όλα τα δυνατά αθροίσματα των εδρών τους ως συνάρτηση του k.

```
n = 1000
k = 1:4
I = numeric(length(k))
for (j in 1:length(k)) {
    N = numeric(n)
    for (i in 1:n) {
        S = numeric(6 * k[j])
        while (sum(S == 0) > k[j] - 1) {
            U = runif(k[j])
            Y = floor(6 * U) + 1
            X = sum(Y)
            S[X] = S[X] + 1
            N[i] = N[i] + 1
        }
    }
    I[j] = mean(N)
}
plot(k, I, "b", xlab = "Number of Dice", ylab = "Expected Number of Rolls",
    pch = 16, lwd = 2)
```



Παράδειγμα 2.14. Θέλουμε να εκτιμήσουμε την πιθανότητα τουλάχιστον 2 από k άτομα να έχουν την ίδια μέρα του χρόνου γενέθλια ως συνάρτηση του k.

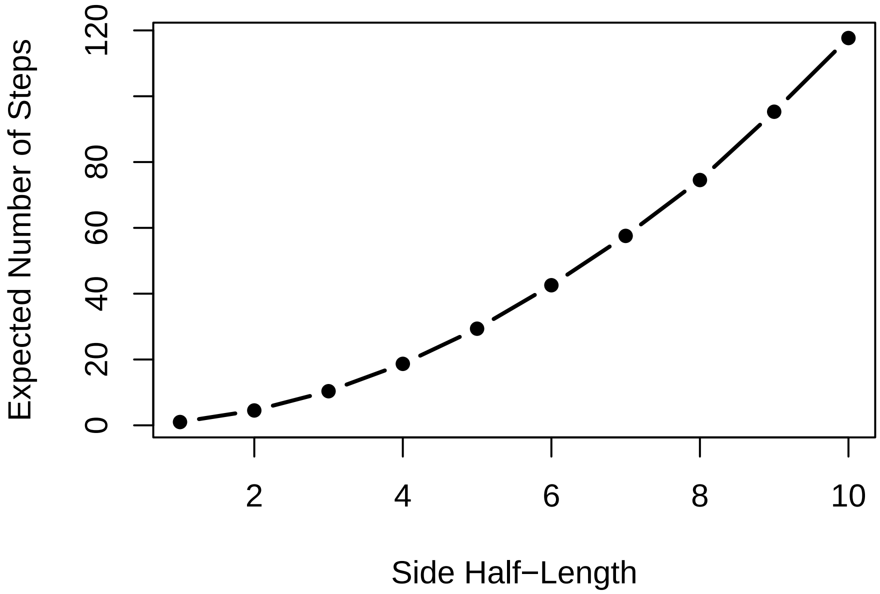
```r
n = 10000
k = 1:40
I = numeric(length(k))
for (j in 1:length(k)) {
    found = logical(n)
    for (i in 1:n) {
        D = numeric(365)
        l = 0
        while (!found[i] && l < k[j]) {
            U = runif(1)
            X = floor(365 * U) + 1
            if (D[X] == 1) {
                found[i] = TRUE
            } else {
                D[X] = D[X] + 1
                l = l + 1
            }
        }
    }
    I[j] = mean(found)
}
plot(k, I, "b", xlab = "Number of People", ylab = "Probability of Sharing Birthday",
    pch = 16, lwd = 2)
```



Παράδειγμα 2.15. Έστω ένα τετράγωνο πλευράς $2k$ για $k \in \mathbb{N}$ με κέντρο την αρχή των αξόνων. Ένα σώμα εκτελεί τυχαίο περίπατο πάνω στα ζεύγη ακεραίων με αφετηρία την αρχή των αξόνων μέχρι να φτάσει στο σύνορο του τετραγώνου. Θέλουμε να εκτιμήσουμε το μέσο πλήθος βημάτων ως συνάρτηση του k.

```
n = 10000
k = 1:10
I = numeric(length(k))
for (j in 1:length(k)) {
    N = numeric(n)
    for (i in 1:n) {
        X = 0
        Y = 0
        while (abs(X) < k[j] && abs(Y) < k[j]) {
            U = runif(1)
            if (U <= 0.25) {
                X = X + 1
            } else if (U <= 0.5) {
                Y = Y + 1
            } else if (U <= 0.75) {
                X = X - 1
            } else {
                Y = Y - 1
            }
            N[i] = N[i] + 1
        }
    }
    I[j] = mean(N)
}
plot(k, I, "b", xlab = "Side Half-Length", ylab = "Expected Number of Steps",
```

```
     pch = 16, lwd = 2)
```



Expected Number of Steps
0
20
40
60
80
120
2
4
6
8
10
Side Half−Length

3 Προσομοίωση Συστημάτων Διακριτών Γεγονότων

Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\text{Exp}(\mu)$. Θέλουμε να προσομοιώσουμε την κατάσταση του συστήματος στον χρόνο.

Είσοδος: Ρυθμός αφίξεων λ και ρυθμός εξυπηρετήσεων μ.

Θέτουμε $Q \leftarrow 0$, $D \leftarrow \infty$, προσομοιώνουμε $A \sim \text{Exp}(\lambda)$ και επαναλαμβάνουμε τα παρακάτω βήματα:

1: Θέτουμε $t \leftarrow \min\{A, D\}$.

2: Αν $t = A$, τότε:

i: Θέτουμε $Q \leftarrow Q + 1$, προσομοιώνουμε $R \sim \text{Exp}(\lambda)$ και θέτουμε $A \leftarrow t + R$.

ii: Αν $Q = 1$, τότε προσομοιώνουμε $X \sim \text{Exp}(\mu)$ και θέτουμε $D \leftarrow t + X$.

Αν $t = D$, τότε:

i: Θέτουμε $Q \leftarrow Q - 1$.

ii: Αν $Q > 0$, τότε προσομοιώνουμε $X \sim \text{Exp}(\mu)$ και θέτουμε $D \leftarrow t + X$. Διαφορετικά, θέτουμε $D \leftarrow \infty$.

Παράδειγμα 3.1. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\text{Exp}(\mu)$. Η περίοδος συνεχούς λειτουργίας του συστήματος ορίζεται ως το χρονικό διάστημα από τη στιγμή που ένας πελάτης αφικνείται σε άδειο σύστημα μέχρι τη στιγμή που ένας αναχωρούμενος πελάτης αφήνει το σύστημα άδειο. Θέλουμε να εκτιμήσουμε τη μέση διάρκεια μίας περιόδου συνεχούς λειτουργίας και τον αναμενόμενο μέγιστο αριθμό πελατών που είναι παρόντες στο σύστημα κατά τη διάρκεια μίας περιόδου συνεχούς λειτουργίας.

```
n = 10000
lambda = 4
mu = 6
Y = numeric(n)
M = numeric(n)
for (i in 1:n) {
    U = runif(1)
    A = -log(U)/lambda
    t = A
    Q = 1
    Y[i] = t
    M[i] = 1
    U = runif(1)
    A = t - log(U)/lambda
    V = runif(1)
    D = t - log(V)/mu
    while (Q > 0) {
        t = min(A, D)
        if (t == A) {
```

```
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            }
            M[i] = max(M[i], Q)
        } else {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                D = Inf
            }
        }
    }
    Y[i] = t - Y[i]
}
mean(Y)
```

```
## [1] 0.4986261
```

```
mean(M)
```

```
## [1] 1.9508
```

Παράδειγμα 3.2. Έστω ένα σύστημα εξυπηρέτησης $M/E_s/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή Gamma(s, μ). Θεωρούμε ότι υπάρχει μία χρονική στιγμή T^* μετά από την οποία δεν επιτρέπονται νέες αφίξεις στο σύστημα, αλλά ο υπηρέτης συνεχίζει την εξυπηρέτηση όλων των πελατών που ήταν ήδη παρόντες στο σύστημα τη στιγμή T^*. Θέλουμε να εκτιμήσουμε την αναμενόμενη υπερωρία που θα χρειαστεί να εργαστεί ο υπηρέτης, τον μέσο χρόνο παραμονής ενός πελάτη στο σύστημα και την αναμενόμενη συνολική περίοδο αργίας του υπηρέτη.

```
n = 1000
lambda = 10
mu = 40
s = 3
Tstar = 100
S = numeric(n)
I = numeric(n)
O = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
```

```r
    A = -log(U)/lambda
    t = A
    N = 0
    temp = 0
    arrivals = numeric(0)
    while (t < Tstar || Q > 0) {
        if (t == A) {
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (A > Tstar) {
                A = Inf
            }
            if (Q == 1) {
                V = runif(s)
                D = t - log(prod(V))/mu
                I[i] = I[i] + t - temp
            }
            N = N + 1
            arrivals = c(arrivals, t)
        } else {
            Q = Q - 1
            if (Q > 0) {
                V = runif(s)
                D = t - log(prod(V))/mu
            } else {
                D = Inf
                temp = t
            }
            S[i] = S[i] + t - arrivals[1]
            arrivals = arrivals[-1]
        }
        t = min(A, D)
    }
    S[i] = S[i]/N
    O[i] = max(temp - Tstar, 0)
}
mean(S)
```

```
## [1] 0.2221672
```

```r
mean(O)
```

```
## [1] 0.1529004
```

```r
mean(I)
```

```
## [1] 25.13479
```

Παράδειγμα 3.3. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\text{Exp}(\mu)$. Όταν αδειάσει το σύστημα, ο υπηρέτης σταματάει να εργάζεται και ξεκινάει την εξυπηρέτηση μόνο όταν συγκεντρωθούν s πελάτες στο σύστημα. Θέλουμε να εκτιμήσουμε το αναμενόμενο ποσοστό του χρόνου μέχρι τη στιγμή T^* που βρίσκονται τουλάχιστον m πελάτες στο σύστημα.

```r
n = 1000
lambda = 4
mu = 6
s = 10
m = 12
Tstar = 100
P = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    D = Inf
    t = A
    vacation = TRUE
    while (t < Tstar) {
        if (t == A) {
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (Q == s && vacation) {
                V = runif(1)
                D = t - log(V)/mu
                vacation = FALSE
            }
            if (Q == m) {
                temp = t
            }
        } else {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                D = Inf
```

```
                vacation = TRUE
            }
            if (Q == m - 1) {
                P[i] = P[i] + t - temp
            }
        }
        t = min(A, D)
    }
    if (Q >= m) {
        P[i] = P[i] + Tstar - temp
    }
    P[i] = 100 * P[i]/Tstar
}
mean(P)
```

```
## [1] 8.75385
```

Παράδειγμα 3.4. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\mathrm{Exp}(\mu)$. Θεωρούμε ότι κάθε πελάτης αναμένει στην ουρά για ένα χρονικό διάστημα που ακολουθεί την κατανομή $\mathrm{Unif}[0, \nu]$ προτού αναχωρήσει από το σύστημα χωρίς να εξυπηρετηθεί. Θέλουμε να εκτιμήσουμε το μέσο πλήθος των χαμένων πελατών μέχρι τη χρονική στιγμή T^*.

```
n = 1000
lambda = 5
mu = 4
nu = 5
Tstar = 100
L = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    t = A
    R = numeric(0)
    while (t < Tstar) {
        if (t == A) {
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                W = runif(1)
```

```
                R = c(R, t + nu * W)
            }
        } else if (t == D) {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
                R = R[-1]
            } else {
                D = Inf
            }
        } else {
            Q = Q - 1
            R = R[-which.min(R)]
            L[i] = L[i] + 1
        }
        t = min(A, D, R)
    }
}
mean(L)
```

```
## [1] 118.119
```

Παράδειγμα 3.5. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\text{Exp}(\mu)$. Θεωρούμε ότι κάθε πελάτης αναμένει στην ουρά για ένα χρονικό διάστημα που ακολουθεί την κατανομή $\text{Unif}[0, \nu]$ προτού αναχωρήσει από το σύστημα χωρίς να εξυπηρετηθεί. Υποθέτουμε ότι κάθε φορά που ο υπηρέτης ολοκληρώνει μία εξυπηρέτηση επιλέγει στη συνέχεια να εξυπηρετήσει τον πελάτη με τον συντομότερο χρόνο αναχώρησης από το σύστημα. Θέλουμε να συγκρίνουμε το μέσο πλήθος των χαμένων πελατών μέχρι τη χρονική στιγμή T^* με αυτό του προηγούμενο παραδείγματος.

```
n = 1000
lambda = 5
mu = 4
nu = 5
Tstar = 100
L = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    t = A
    R = numeric(0)
    while (t < Tstar) {
```

```
        if (t == A) {
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                W = runif(1)
                R = c(R, t + nu * W)
            }
        } else if (t == D) {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
                R = R[-which.min(R)]
            } else {
                D = Inf
            }
        } else {
            Q = Q - 1
            R = R[-which.min(R)]
            L[i] = L[i] + 1
        }
        t = min(A, D, R)
    }
}
mean(L)
```

```
## [1] 100.265
```

Παράδειγμα 3.6. Έστω ένα δίκτυο αποτελούμενο από δύο συστήματα εξυπηρέτησης $M/M/1$ συνδεδεμένα στη σειρά, όπου η διαδικασία αφίξεων στο πρώτο σύστημα είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης στο σύστημα j ακολουθούν την κατανομή $\text{Exp}(\mu_j)$. Θεωρούμε ότι υπάρχει μία χρονική στιγμή T^* μετά από την οποία δεν επιτρέπονται νέες αφίξεις στο δίκτυο, αλλά οι υπηρέτες συνεχίζουν την εξυπηρέτηση όλων των πελατών που ήταν ήδη παρόντες στο δίκτυο τη στιγμή T^*. Θέλουμε να εκτιμήσουμε τον μέσο χρόνο παραμονής ενός πελάτη σε καθένα από τα δύο συστήματα.

```
n = 1000
lambda = 4
mu1 = 5
mu2 = 6
Tstar = 100
```

```
S1 = numeric(n)
S2 = numeric(n)
for (i in 1:n) {
    Q1 = 0
    Q2 = 0
    U = runif(1)
    A = -log(U)/lambda
    D2 = Inf
    t = A
    N = 0
    A1 = numeric(0)
    A2 = numeric(0)
    while (t < Tstar || Q1 > 0 || Q2 > 0) {
        if (t == A) {
            Q1 = Q1 + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (A > Tstar) {
                A = Inf
            }
            if (Q1 == 1) {
                V = runif(1)
                D1 = t - log(V)/mu1
            }
            N = N + 1
            A1 = c(A1, t)
        } else if (t == D1) {
            Q1 = Q1 - 1
            Q2 = Q2 + 1
            if (Q1 > 0) {
                V = runif(1)
                D1 = t - log(V)/mu1
            } else {
                D1 = Inf
            }
            if (Q2 == 1) {
                W = runif(1)
                D2 = t - log(W)/mu2
            }
            A2 = c(A2, t)
            S1[i] = S1[i] + t - A1[1]
            A1 = A1[-1]
        } else {
```

```
            Q2 = Q2 - 1
            if (Q2 > 0) {
                W = runif(1)
                D2 = t - log(W)/mu2
            } else {
                D2 = Inf
            }
            S2[i] = S2[i] + t - A2[1]
            A2 = A2[-1]
        }
        t = min(A, D1, D2)
    }
    S1[i] = S1[i]/N
    S2[i] = S2[i]/N
}
mean(S1)
```

```
## [1] 0.9374927
```

```
mean(S2)
```

```
## [1] 0.4806224
```

Παράδειγμα 3.7. Έστω ένα σύστημα εξυπηρέτησης $M/M/c$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\text{Exp}(\mu)$. Θέλουμε να εκτιμήσουμε τον μέσο χρόνο παραμονής ενός πελάτη στο σύστημα και το αναμενόμενο ποσοστό των πρώτων N^* εξυπηρετήσεων που πραγματοποιούνται από κάθε υπηρέτη.

```
n = 1000
c = 2
lambda = 6
mu = c(4, 3)
Nstar = 1000
S = numeric(n)
P = matrix(0, n, c)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    D = rep(Inf, c)
    N = 0
    arrivals = numeric(0)
    while (N < Nstar || Q > 0) {
        t = min(A, D)
        if (t == A) {
            Q = Q + 1
```

```
            N = N + 1
            if (N < Nstar) {
                U = runif(1)
                A = t - log(U)/lambda
            } else {
                A = Inf
            }
            if (Q <= c) {
                I = match(Inf, D)
                V = runif(1)
                D[I] = t - log(V)/mu[I]
                S[i] = S[i] + D[I] - t
            } else {
                arrivals = c(arrivals, t)
            }
        } else {
            Q = Q - 1
            I = which.min(D)
            if (Q >= c) {
                V = runif(1)
                D[I] = t - log(V)/mu[I]
                S[i] = S[i] + D[I] - arrivals[1]
                arrivals = arrivals[-1]
            } else {
                D[I] = Inf
            }
            P[i, I] = P[i, I] + 1
        }
    }
    S[i] = S[i]/Nstar
    P[i, ] = P[i, ]/Nstar
}
mean(S)
```

```
## [1] 1.021141
```

```
colMeans(P)
```

```
## [1] 0.58169 0.41831
```

Παράδειγμα 3.8. Έστω ένα σύστημα εξυπηρέτησης $M/M/c$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\mathrm{Exp}(\mu)$. Θεωρούμε ότι υπάρχει μία χρονική στιγμή T^* κατά την οποία το σύστημα σταματάει να λειτουργεί και οι πελάτες που δεν έχουν ολοκληρώσει ακόμα την εξυπηρέτησή τους χάνονται. Θέλουμε να εκτιμήσουμε τον μέσο χρόνο παραμονής στο σύστημα ενός πελάτη που έχει ολοκληρώσει την εξυπηρέτησή του πριν τη χρονική στιγμή T^*, το μέσο πλήθος των χαμένων πελατών και την πιθανότητα να χαθούν περισσότεροι από 2 πελάτες.

```
n = 1000
c = 2
lambda = 6
mu = c(4, 3)
Tstar = 100
S = numeric(n)
Q = numeric(n)
for (i in 1:n) {
    U = runif(1)
    A = -log(U)/lambda
    D = rep(Inf, c)
    t = A
    N = 0
    arrivals = numeric(0)
    while (t < Tstar) {
        if (t == A) {
            Q[i] = Q[i] + 1
            U = runif(1)
            A = t - log(U)/lambda
            if (Q[i] <= c) {
                I = match(Inf, D)
                V = runif(1)
                D[I] = t - log(V)/mu[I]
                if (D[I] < Tstar) {
                  S[i] = S[i] + D[I] - t
                }
            } else {
                arrivals = c(arrivals, t)
            }
        } else {
            Q[i] = Q[i] - 1
            I = which.min(D)
            if (Q[i] >= c) {
                V = runif(1)
                D[I] = t - log(V)/mu[I]
                if (D[I] < Tstar) {
                  S[i] = S[i] + D[I] - arrivals[1]
                }
                arrivals = arrivals[-1]
            } else {
                D[I] = Inf
            }
```

```
            N = N + 1
        }
        t = min(A, D)
    }
    S[i] = S[i]/N
}
mean(S)
```

```
## [1] 0.9902442
```

```
mean(Q)
```

```
## [1] 6.151
```

```
mean(Q > 2)
```

```
## [1] 0.655
```

Παράδειγμα 3.9. Έστω ένα δίκτυο αποτελούμενο από δύο παράλληλα συστήματα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων στο δίκτυο είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης στο σύστημα j ακολουθούν την κατανομή $\mathrm{Exp}(\mu_j)$. Θεωρούμε ότι ένας αφικνούμενος πελάτης εισέρχεται στο σύστημα με το μικρότερο πλήθος πελατών ή στο πρώτο σύστημα αν και τα δύο έχουν το ίδιο πλήθος πελατών. Θέλουμε να εκτιμήσουμε τον μέσο χρόνο παραμονής ενός πελάτη στο σύστημα και το αναμενόμενο ποσοστό των πρώτων N^* πελατών που εισέρχονται στο πρώτο σύστημα.

```
n = 1000
lambda = 6
mu1 = 4
mu2 = 3
Nstar = 1000
S = numeric(n)
P = numeric(n)
for (i in 1:n) {
    Q1 = 0
    Q2 = 0
    U = runif(1)
    A = -log(U)/lambda
    D1 = Inf
    D2 = Inf
    N = 0
    A1 = numeric(0)
    A2 = numeric(0)
    while (N < Nstar || Q1 > 0 || Q2 > 0) {
        t = min(A, D1, D2)
        if (t == A) {
            if (Q1 <= Q2) {
                Q1 = Q1 + 1
```

```
            if (Q1 == 1) {
              V = runif(1)
              D1 = t - log(V)/mu1
            }
            A1 = c(A1, t)
        } else {
            Q2 = Q2 + 1
            if (Q2 == 1) {
              W = runif(1)
              D2 = t - log(W)/mu2
            }
            A2 = c(A2, t)
        }
        N = N + 1
        if (N < Nstar) {
            U = runif(1)
            A = t - log(U)/lambda
        } else {
            A = Inf
        }
    } else if (t == D1) {
        Q1 = Q1 - 1
        if (Q1 > 0) {
            V = runif(1)
            D1 = t - log(V)/mu1
        } else {
            D1 = Inf
        }
        P[i] = P[i] + 1
        S[i] = S[i] + t - A1[1]
        A1 = A1[-1]
    } else {
        Q2 = Q2 - 1
        if (Q2 > 0) {
            W = runif(1)
            D2 = t - log(W)/mu2
        } else {
            D2 = Inf
        }
        S[i] = S[i] + t - A2[1]
        A2 = A2[-1]
    }
}
```

```
    S[i] = S[i]/Nstar
    P[i] = P[i]/Nstar
}
mean(S)
```

```
## [1] 1.072401
```

```
mean(P)
```

```
## [1] 0.583024
```

Παράδειγμα 3.10. Έστω ένα δίκτυο αποτελούμενο από δύο παράλληλα συστήματα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων στο δίκτυο είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης στο σύστημα j ακολουθούν την κατανομή $\mathrm{Exp}(\mu_j)$. Θεωρούμε ότι ένας αφικνούμενος πελάτης εισέρχεται στο πρώτο σύστημα με πιθανότητα p, όπου p είναι το εκτιμημένο αναμενόμενο ποσοστό των πρώτων N^* πελατών που εισέρχονται στο πρώτο σύστημα του προηγούμενου παραδείγματος. Θέλουμε να συγκρίνουμε τον μέσο χρόνο παραμονής ενός πελάτη στο σύστημα με αυτόν του προηγούμενο παραδείγματος.

```
n = 1000
lambda = 6
mu1 = 4
mu2 = 3
Nstar = 1000
p = mean(P)
S = numeric(n)
for (i in 1:n) {
    Q1 = 0
    Q2 = 0
    U = runif(1)
    A = -log(U)/lambda
    D1 = Inf
    D2 = Inf
    N = 0
    A1 = numeric(0)
    A2 = numeric(0)
    while (N < Nstar || Q1 > 0 || Q2 > 0) {
        t = min(A, D1, D2)
        if (t == A) {
            U = runif(1)
            if (U < p) {
                Q1 = Q1 + 1
                if (Q1 == 1) {
                  V = runif(1)
                  D1 = t - log(V)/mu1
                }
                A1 = c(A1, t)
```

```
            } else {
                Q2 = Q2 + 1
                if (Q2 == 1) {
                  W = runif(1)
                  D2 = t - log(W)/mu2
                }
                A2 = c(A2, t)
            }
            N = N + 1
            if (N < Nstar) {
                U = runif(1)
                A = t - log(U)/lambda
            } else {
                A = Inf
            }
        } else if (t == D1) {
            Q1 = Q1 - 1
            if (Q1 > 0) {
                V = runif(1)
                D1 = t - log(V)/mu1
            } else {
                D1 = Inf
            }
            S[i] = S[i] + t - A1[1]
            A1 = A1[-1]
        } else {
            Q2 = Q2 - 1
            if (Q2 > 0) {
                W = runif(1)
                D2 = t - log(W)/mu2
            } else {
                D2 = Inf
            }
            S[i] = S[i] + t - A2[1]
            A2 = A2[-1]
        }
    }
    S[i] = S[i]/Nstar
}
mean(S)
```

```
## [1] 1.827888
```

Παράδειγμα 3.11. Έστω ένα σύστημα που χρειάζεται N μηχανήματα για να λειτουργήσει. Θεωρούμε ότι υπάρχουν s εφεδρικά μηχανήματα στο σύστημα. Κάθε μηχάνημα λειτουργεί για ένα χρονικό διάστημα που ακολουθεί την κατανομή $\text{Exp}(\lambda)$ προτού χαλάσει. Όποτε χαλάει ένα μηχάνημα, αντικαθίσταται άμεσα από ένα εφεδρικό και στέλνεται στο συνεργείο για επισκευή. Ένας μηχανικός επισκευάζει τα μηχανήματα στο συνεργείο σε χρόνο που ακολουθεί την κατανομή $\text{Exp}(\mu)$. Μόλις επισκευαστεί ένα μηχάνημα, γίνεται διαθέσιμο ως εφεδρικό για όποτε χρειαστεί. Το σύστημα σταματάει να λειτουργεί όταν χαλάσει ένα μηχάνημα και δεν υπάρχει κανένα εφεδρικό για να το αντικαταστήσει. Θέλουμε να εκτιμήσουμε τον αναμενόμενο χρόνο μέχρι να σταματήσει να λειτουργεί το σύστημα.

```
n = 1e+05
lambda = 1
mu = 2
s = 3
N = 4
C = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(N)
    A = -log(U)/lambda
    D = Inf
    while (Q <= s) {
        t = min(A, D)
        if (t == D) {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                D = Inf
            }
        } else {
            Q = Q + 1
            U = runif(1)
            A[which.min(A)] = t - log(U)/lambda
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            }
        }
    }
    C[i] = t
}
mean(C)
```

```
## [1] 1.536251
```

Λήμμα 3.1. Έστω ανεξάρτητες τυχαίες μεταβλητές $X \sim \text{Exp}(\lambda)$ και $Y \sim \text{Exp}(\mu)$. Τότε, παίρνουμε ότι $W = \min\{X, Y\} \sim \text{Exp}(\lambda + \mu)$.

Απόδειξη. Για $w > 0$, υπολογίζουμε ότι:

$$
\begin{aligned}
F_W(w) &= \mathbb{P}\left(\min\{X,Y\} \leqslant w\right) = 1 - \mathbb{P}\left(\min\{X,Y\} > w\right) = 1 - \mathbb{P}(X > w, Y > w) \\
&= 1 - \mathbb{P}(X > w)\mathbb{P}(Y > w) = 1 - [1 - F_X(w)]\,[1 - F_Y(w)] = 1 - e^{-\lambda w}e^{-\mu w} = 1 - e^{-(\lambda+\mu)w}.
\end{aligned}
$$

□

Πόρισμα 3.1. Ο χρόνος μέχρι να χαλάσει ένα από τα N μηχανήματα που βρίσκονται σε λειτουργία ακολουθεί την κατανομή $\text{Exp}(N\lambda)$.

```
C = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/(N * lambda)
    D = Inf
    while (Q <= s) {
        t = min(A, D)
        if (t == A) {
            Q = Q + 1
            U = runif(1)
            A = t - log(U)/(N * lambda)
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            }
        } else {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                D = Inf
            }
        }
    }
    C[i] = t
}
mean(C)
```

```
## [1] 1.531271
```

Παράδειγμα 3.12. Έστω ότι μηνύματα φθάνουν σε μία μονάδα επικοινωνίας σύμφωνα με μία διαδικασία Poisson ρυθμού λ. Η μονάδα διαθέτει c κανάλια επικοινωνίας. Αν όλα τα κανάλια είναι απασχολημένα κατά την άφιξη ενός μηνύματος, τότε το μήνυμα χάνεται. Ο καιρός είναι αρχικά καλός και εναλλάσσεται μεταξύ καλών και κακών περιόδων με διάρκεια s_1 και s_2 ωρών αντίστοιχα. Αν ο καιρός είναι καλός τη στιγμή που φτάσει ένα μήνυμα, τότε ο χρόνος που απαιτείται για την επεξεργασία του ακολουθεί την κατανομή $\text{Beta}(\mu_1, 1)$, διαφορετικά ακολουθεί την κατανομή $\text{Beta}(\mu_2, 1)$. Θέλουμε να εκτιμήσουμε το μέσο πλήθος χαμένων μηνυμάτων μέχρι τη χρονική στιγμή T^*.

```r
n = 1000
c = 3
lambda = 2
mu = c(1, 3)
s = c(2, 1)
Tstar = 100
L = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    D = rep(Inf, c)
    t = A
    while (t < Tstar) {
        if (t == A) {
            U = runif(1)
            A = t - log(U)/lambda
            if (Q < c) {
                Q = Q + 1
                V = runif(1)
                if (t%%sum(s) <= s[1]) {
                  D[match(Inf, D)] = t + V^(1/mu[1])
                } else {
                  D[match(Inf, D)] = t + V^(1/mu[2])
                }
            } else {
                L[i] = L[i] + 1
            }
        } else {
            Q = Q - 1
            D[which.min(D)] = Inf
        }
        t = min(A, D)
    }
}
mean(L)
```

```
## [1] 17.229
```

4 Τεχνικές Μείωσης Διασποράς

Αντιθετικές Μεταβλητές

Έστω τυχαίες μεταβλητές X_1, X_2, X, Y με μέση τιμή μ και διασπορά σ^2. Θεωρούμε ότι οι τυχαίες μεταβλητές X_1, X_2 είναι ανεξάρτητες, ενώ οι τυχαίες μεταβλητές X, Y έχουν συνδιακύμανση $\mathrm{Cov}(X, Y) = \sigma_{XY}$ και συντελεστή συσχέτισης Pearson $\mathrm{Corr}(X, Y) = \rho_{XY}$. Τότε, παρατηρούμε ότι:

$$\mathbb{E}\left(\frac{X_1 + X_2}{2}\right) = \mathbb{E}\left(\frac{X + Y}{2}\right) = \mu,$$

$$\mathrm{Var}\left(\frac{X_1 + X_2}{2}\right) = \frac{\mathrm{Var}(X_1) + \mathrm{Var}(X_2)}{4} = \frac{\sigma^2}{2},$$

$$\mathrm{Var}\left(\frac{X + Y}{2}\right) = \frac{\mathrm{Var}(X) + \mathrm{Var}(Y) + 2\mathrm{Cov}(X, Y)}{4} = \frac{\sigma^2 + \sigma_{XY}}{2},$$

$$\mathrm{Var}\left(\frac{X + Y}{2}\right) < \mathrm{Var}\left(\frac{X_1 + X_2}{2}\right) \quad \Leftrightarrow \quad \sigma_{XY} < 0.$$

Το ποσοστό μείωσης της διασποράς μίας εκτιμήτριας του μ με χρήση της μεθόδου αντιθετικών μεταβλητών ισούται με:

$$100 \cdot \frac{\sigma^2 - (\sigma^2 + \sigma_{XY})}{\sigma^2} = 100 \cdot \frac{|\sigma_{XY}|}{\sigma^2} = 100 \cdot |\rho_{XY}| \,.$$

Πρόταση 4.1. Έστω ανεξάρτητες τυχαίες μεταβλητές $U_1, U_2, \dots, U_k \sim \mathrm{Unif}[0, 1]$. Θεωρούμε μία μονότονη κατά συντεταγμένη συνάρτηση $h : [0, 1]^k \to \mathbb{R}$. Τότε, ισχύει ότι:

$$\mathrm{Cov}\left[h\left(U_1, \dots, U_k\right), h\left(1 - U_1, \dots, 1 - U_k\right)\right] \leqslant 0.$$

Πόρισμα 4.1. Έστω ανεξάρτητες τυχαίες μεταβλητές $U, U_1, U_2 \sim \mathrm{Unif}[0, 1]$. Θεωρούμε μία μονότονη συνάρτηση $h : [0, 1] \to \mathbb{R}$. Τότε, ισχύει ότι:

$$\mathbb{E}\left[\frac{h(U) + h(1 - U)}{2}\right] = \mathbb{E}\left[\frac{h(U_1) + h(U_2)}{2}\right],$$

$$\mathrm{Var}\left[\frac{h(U) + h(1 - U)}{2}\right] \leqslant \mathrm{Var}\left[\frac{h(U_1) + h(U_2)}{2}\right].$$

Παράδειγμα 4.1. Έστω τυχαία μεταβλητή $U \sim \mathrm{Unif}[0, 1]$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(e^U\right)$. Παρατηρούμε ότι η συνάρτηση $h(x) = e^x$ είναι αύξουσα για $x \in [0, 1]$.

```r
n = 1e+05
U = runif(n)
X = exp(U)
I = mean(X)
print(I)
```

```
## [1] 1.718251
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.2427706
```

```
U = runif(n/2)
X = exp(U)
Y = exp(1 - U)
W = (X + Y)/2
I = mean(W)
print(I)
```

```
## [1] 1.718183
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.003916482
```

```
rho = (2 * VarW - VarX)/VarX
print(rho)
```

```
## [1] -0.9677351
```

```
100 * abs(rho)
```

```
## [1] 96.77351
```

Παράδειγμα 4.2. Έστω ανεξάρτητες τυχαίες μεταβλητές $U, V \sim \mathrm{Unif}[0,1]$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left[e^{(U+V)^2}\right]$. Παρατηρούμε ότι η συνάρτηση $h(x,y) = e^{(x+y)^2}$ είναι αύξουσα κατά συντεταγμένη για $x, y \in [0,1]$.

```
n = 1e+05
U = runif(n)
V = runif(n)
X = exp((U + V)^2)
I = mean(X)
print(I)
```

```
## [1] 4.886297
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 35.24747
```

```
U = runif(n/2)
V = runif(n/2)
X = exp((U + V)^2)
Y = exp((2 - U - V)^2)
W = (X + Y)/2
```

```
I = mean(W)
print(I)
```

```
## [1] 4.897734
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 11.51644
```

```
rho = (2 * VarW - VarX)/VarX
print(rho)
```

```
## [1] -0.3465383
```

```
100 * abs(rho)
```

```
## [1] 34.65383
```

Παράδειγμα 4.3. Έστω ακολουθία ανεξάρτητων τυχαίων μεταβλητών $U_1, U_2, \cdots \sim \text{Unif}[0,1]$. Θέλουμε να εκτιμήσουμε τη μέση τιμή της τυχαίας μεταβλητής:

$$X = \sup\left\{k \in \mathbb{N} : U_1 < U_2 < \cdots < U_{k-1}\right\}.$$

Ορίζουμε την τυχαία μεταβλητή:

$$Y = \sup\left\{k \in \mathbb{N} : 1 - U_1 < 1 - U_2 < \cdots < 1 - U_{k-1}\right\} = \sup\left\{k \in \mathbb{N} : U_1 > U_2 > \cdots > U_{k-1}\right\}.$$

```
n = 1e+06
X = numeric(n)
for (i in 1:n) {
    Uold = runif(1)
    Unew = runif(1)
    X[i] = 2
    while (Uold < Unew) {
        Uold = Unew
        Unew = runif(1)
        X[i] = X[i] + 1
    }
}
I = mean(X)
print(I)
```

```
## [1] 2.717766
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.765044
```

```r
X = numeric(n/2)
Y = numeric(n/2)
for (i in 1:(n/2)) {
    Uold = runif(1)
    Unew = runif(1)
    X[i] = 2
    Y[i] = 2
    if (Uold < Unew) {
        while (Uold < Unew) {
            Uold = Unew
            Unew = runif(1)
            X[i] = X[i] + 1
        }
    } else {
        while (Uold > Unew) {
            Uold = Unew
            Unew = runif(1)
            Y[i] = Y[i] + 1
        }
    }
}
W = (X + Y)/2
I = mean(W)
print(I)
```

```
## [1] 2.718524
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.1254453
```

```r
rho = (2 * VarW - VarX)/VarX
print(rho)
```

```
## [1] -0.6720574
```

```r
100 * abs(rho)
```

```
## [1] 67.20574
```

Σημείωση 4.1. Έστω τυχαία μεταβλητή $X \sim \mathcal{N}\left(\mu, \sigma^2\right)$. Τότε, η τυχαία μεταβλητή $Y = 2\mu - X$ είναι ισόνομη και αρνητικά συσχετισμένη με την X.

Παράδειγμα 4.4. Έστω τυχαία μεταβλητή $Z \sim \mathcal{N}(0, 1)$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(e^Z\right)$.

```r
n = 1e+05
U = runif(n/2)
```

```r
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = exp(Z)
I = mean(X)
print(I)
```

```
## [1] 1.651902
```

```r
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 4.685197
```

```r
U = runif(n/4)
D = -2 * log(U)
V = runif(n/4)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = exp(Z)
Y = exp(-Z)
W = (X + Y)/2
I = mean(W)
print(I)
```

```
## [1] 1.654055
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 1.504112
```

```r
rho = (2 * VarW - VarX)/VarX
print(rho)
```

```
## [1] -0.3579303
```

```r
100 * abs(rho)
```

```
## [1] 35.79303
```

Μεταβλητές Ελέγχου

Θεωρούμε δύο τυχαίες μεταβλητές X και Y με $\mathbb{E}(X) = \mu$, $\mathbb{E}(Y) = \mu_Y$, $\mathrm{Var}(X) = \sigma_X^2$, $\mathrm{Var}(Y) = \sigma_Y^2$, $\mathrm{Cov}(X, Y) = \sigma_{XY}$ και $\mathrm{Corr}(X, Y) = \rho_{XY}$. Παρατηρούμε ότι η τυχαία μεταβλητή $W_c = X + c\,(Y - \mu_Y)$ έχει κι αυτή μέση τιμή μ για κάθε $c \in \mathbb{R}$. Υπολογίζουμε ότι:

$$\mathrm{Var}\,(W_c) = \mathrm{Var}(X) + c^2\mathrm{Var}\,(Y - \mu_Y) + 2c\mathrm{Cov}\,(X, Y - \mu_Y) = \sigma_Y^2 c^2 + 2\sigma_{XY} c + \sigma_X^2.$$

Εφόσον $\sigma_Y^2 > 0$, γνωρίζουμε ότι η συνάρτηση $\mathrm{Var}\left(W_c\right)$ έχει μοναδικό ολικό ελάχιστο στο σημείο:

$$c^* = -\frac{\sigma_{XY}}{\sigma_Y^2}.$$

Παρατηρούμε ότι:

$$\mathrm{Var}\left(W_{c^*}\right) = \frac{\sigma_{XY}^2}{\sigma_Y^2} - 2\frac{\sigma_{XY}^2}{\sigma_Y^2} + \sigma_X^2 = \sigma_X^2 - \frac{\sigma_{XY}^2}{\sigma_Y^2} = \sigma_X^2\left(1 - \frac{\sigma_{XY}^2}{\sigma_X^2\sigma_Y^2}\right) = \sigma_X^2\left(1 - \rho_{XY}^2\right) \leqslant \sigma_X^2.$$

Επομένως, το ποσοστό μείωσης της διασποράς μίας εκτιμήτριας του μ με χρήση της μεθόδου μεταβλητής ελέγχου ισούται με:

$$100 \cdot \frac{\sigma_X^2 - \sigma_X^2\left(1-\rho_{XY}^2\right)}{\sigma_X^2} = 100 \cdot \rho_{XY}^2.$$

Αν X και Y ασυσχέτιστες, τότε $\mathrm{Var}\left(W_{c^*}\right) = \sigma_X^2$, δηλαδή δεν μπορεί να επιτευχθεί μείωση διασποράς για τη συγκεκριμένη επιλογή μεταβλητής ελέγχου Y. Οι ποσότητες σ_{XY} και σ_Y^2 δεν είναι συνήθως υπολογίσιμες, οπότε εκτιμούνται από τα προσομοιωμένα δεδομένα $(X_1, Y_1), \ldots, (X_n, Y_n)$ ως:

$$\hat{\sigma}_{XY} = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right), \quad \hat{\sigma}_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2.$$

Παράδειγμα 4.5. Έστω τυχαία μεταβλητή $U \sim \mathrm{Unif}[0,1]$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(\sqrt{1-U^2}\right)$. Θα χρησιμοποιήσουμε αρχικά ως μεταβλητή ελέγχου την $Y = U$ με $\mathbb{E}(Y) = 0.5$.

```
n = 1e+05
U = runif(n)
X = sqrt(1 - U^2)
I = mean(X)
print(I)
```

```
## [1] 0.78518
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.05000468
```

```
Y = U
muY = 0.5
VarY = var(Y)
Cov = cov(X, Y)
c = -Cov/VarY
W = X + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.7850612
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.007591752
```

```
rho = Cov/sqrt(VarX * VarY)
print(rho)
```

```
## [1] -0.920971
```

```
100 * rho^2
```

```
## [1] 84.81877
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = U^2$ με:

$$\mathbb{E}(Y) = \mathrm{Var}(U) + [\mathbb{E}(U)]^2 = \frac{1}{12} + \frac{1}{4} = \frac{1}{3}.$$

```
Y = U^2
muY = 1/3
VarY = var(Y)
Cov = cov(X, Y)
c = -Cov/VarY
W = X + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.7852937
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.001647797
```

```
rho = Cov/sqrt(VarX * VarY)
print(rho)
```

```
## [1] -0.9833905
```

```
100 * rho^2
```

```
## [1] 96.70568
```

Παράδειγμα 4.6. Έστω τυχαία μεταβλητή $S \sim \mathrm{Gamma}(2, 1)$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(S^2 \leqslant 4)$. Θα χρησιμοποιήσουμε αρχικά ως μεταβλητή ελέγχου την $Y = S$ με $\mathbb{E}(Y) = 2$.

```
n = 1e+05
U = matrix(runif(2 * n), n)
R = -log(U)
S = rowSums(R)
X = S^2 <= 4
```

```r
I = mean(X)
print(I)
```

```
## [1] 0.59344
```

```r
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.241269
```

```r
Y = S
muY = 2
VarY = var(Y)
Cov = cov(X, Y)
c = -Cov/VarY
W = X + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.5937954
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.09494908
```

```r
rho = Cov/sqrt(VarX * VarY)
print(rho)
```

```
## [1] -0.7787591
```

```r
100 * rho^2
```

```
## [1] 60.64657
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = S^2$ με:

$$\mathbb{E}(Y) = \mathrm{Var}(S) + [\mathbb{E}(S)]^2 = 2 + 4 = 6.$$

```r
Y = S^2
muY = 6
VarY = var(Y)
Cov = cov(X, Y)
c = -Cov/VarY
W = X + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.5936546
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.1553603
```

```r
rho = Cov/sqrt(VarX * VarY)
print(rho)
```

```
## [1] -0.5967191
```

```r
100 * rho^2
```

```
## [1] 35.60737
```

Μέθοδος Δέσμευσης

Έστω δύο τυχαίες μεταβλητές X, Y με $\mathbb{E}(X) = \mu$ και $\text{Var}(X) = \sigma^2$. Σύμφωνα με το θεώρημα διπλής μέσης τιμής, γνωρίζουμε ότι $\mathbb{E}(X) = \mathbb{E}\left[\mathbb{E}\left(X \mid Y\right)\right]$, δηλαδή η τυχαία μεταβλητή $W = \mathbb{E}\left(X \mid Y\right)$ έχει κι αυτή μέση τιμή μ. Σύμφωνα με το θεώρημα ολικής διασποράς, γνωρίζουμε ότι:

$$\text{Var}(X) = \text{Var}\left[\mathbb{E}\left(X \mid Y\right)\right] + \mathbb{E}\left[\text{Var}\left(X \mid Y\right)\right] \quad \Rightarrow \quad \text{Var}\left[\mathbb{E}\left(X \mid Y\right)\right] \leqslant \sigma^2.$$

Το ποσοστό μείωσης της διασποράς μίας εκτιμήτριας του μ με χρήση της μεθόδου δέσμευσης ισούται με:

$$100 \cdot \frac{\text{Var}(X) - \text{Var}\left[\mathbb{E}\left(X \mid Y\right)\right]}{\text{Var}(X)} = 100 \cdot \frac{\mathbb{E}\left[\text{Var}\left(X \mid Y\right)\right]}{\sigma^2}.$$

Σημείωση 4.2. Γνωρίζουμε ότι $\text{Var}\left(X \mid Y\right) \equiv 0$ αν και μόνο αν $X = g(Y)$ για κάποια μετρήσιμη συνάρτηση g. Σε αυτήν την περίπτωση, παρατηρούμε ότι $\text{Var}\left[\mathbb{E}\left(X \mid Y\right)\right] = \sigma^2$, δηλαδή δεν μπορεί να επιτευχθεί μείωση διασποράς με τη μέθοδο της δέσμευσης για τη συγκεκριμένη επιλογή τυχαίας μεταβλητής Y.

Παράδειγμα 4.7. Έστω δύο τυχαίες μεταβλητές $Y \sim \text{Exp}(1)$ και $(S \mid Y) \sim \mathcal{N}(Y, 4)$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(S > 1)$. Για $y > 0$, υπολογίζουμε ότι:

$$\mathbb{P}(S > 1 \mid Y = y) = \mathbb{P}\left(\frac{S-y}{2} > \frac{1-y}{2} \middle| Y = y\right) = 1 - \Phi\left(\frac{1-y}{2}\right).$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{P}(S > 1) = \mathbb{E}\left(\mathbb{1}_{\{S>1\}}\right) = \mathbb{E}\left[\mathbb{E}\left(\mathbb{1}_{\{S>1\}} \middle| Y\right)\right] = \mathbb{E}\left[\mathbb{P}\left(S > 1 \mid Y\right)\right] = \mathbb{E}\left[1 - \Phi\left(\frac{1-Y}{2}\right)\right].$$

```r
n = 1e+05
U = runif(n)
Y = -log(U)
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
```

```
Z = sqrt(D) * c(cos(Theta), sin(Theta))
S = 2 * Z + Y
X = S > 1
I = mean(X)
print(I)
```

```
## [1] 0.49067
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.249913
```

```
W = 1 - pnorm((1 - Y)/2)
I = mean(W)
print(I)
```

```
## [1] 0.4905546
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.02691248
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 89.23126
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την Y με $\mathbb{E}(Y) = 1$.

```
muY = 1
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.4900535
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.001347862
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 99.46067
```

Παράδειγμα 4.8. Θέλουμε να προσεγγίσουμε την τιμή της σταθεράς π. Θεωρούμε δύο ανεξάρτητες τυχαίες μεταβλητές $U, V \sim \text{Unif}[0,1]$. Τότε, γνωρίζουμε ότι:

$$\pi = 4\mathbb{P}\left(U^2 + V^2 \leqslant 1\right) = \mathbb{E}\left(4 \cdot \mathbb{1}_{\{U^2+V^2 \leqslant 1\}}\right).$$

Για $u \in [0,1]$, υπολογίζουμε ότι:

$$\mathbb{P}\left(U^2 + V^2 \leqslant 1 \middle| U = u\right) = \mathbb{P}\left(V \leqslant \sqrt{1-u^2} \middle| U = u\right) = \mathbb{P}\left(V \leqslant \sqrt{1-u^2}\right) = \sqrt{1-u^2}.$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\pi = \mathbb{E}\left[\mathbb{E}\left(4 \cdot \mathbb{1}_{\{U^2+V^2 \leqslant 1\}} \middle| U\right)\right] = \mathbb{E}\left[4\mathbb{P}\left(U^2 + V^2 \leqslant 1 \middle| U\right)\right] = \mathbb{E}\left(4\sqrt{1-U^2}\right).$$

```
n = 1e+06
U = runif(n)
V = runif(n)
X = 4 * (U^2 + V^2 <= 1)
I = mean(X)
print(I)
```

```
## [1] 3.141728
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 2.696457
```

```
W = 4 * sqrt(1 - U^2)
I = mean(W)
print(I)
```

```
## [1] 3.141884
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.7960046
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 70.47961
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = U$ με $\mathbb{E}(Y) = 0.5$.

```
Y = U
muY = 0.5
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
```

```r
print(I)
```

```
## [1] 3.141869
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.1205411
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 95.52965
```

Παράδειγμα 4.9. Έστω δύο ανεξάρτητες τυχαίες μεταβλητές $R \sim \text{Exp}(1)$ και $S \sim \text{Exp}(0.5)$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(R+S>4)$. Για $s>0$, υπολογίζουμε ότι:

$$\mathbb{P}(R+S>4 \mid S=s) = \mathbb{P}\left(R>4-s \mid S=s\right) = \mathbb{P}(R>4-s) = \begin{cases} e^{-(4-s)}, & s \leqslant 4 \\ 1, & s>4 \end{cases}.$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{P}(R+S>4) = \mathbb{E}\left(\mathbb{1}_{\{R+S>4\}}\right) = \mathbb{E}\left[\mathbb{E}\left(\mathbb{1}_{\{R+S>4\}}\middle| S\right)\right] = \mathbb{E}\left[\mathbb{P}\left(R+S>4 \mid S\right)\right] = \mathbb{E}\left[\min\left\{e^{-(4-S)},1\right\}\right].$$

```r
n = 1e+05
U = runif(n)
R = -log(U)
V = runif(n)
S = -2 * log(V)
X = R + S > 4
I = mean(X)
print(I)
```

```
## [1] 0.253
```

```r
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.188991
```

```r
W = pmin(exp(-(4 - S)), 1)
I = mean(W)
print(I)
```

```
## [1] 0.2518087
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.11645
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 38.38331
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = S$ με $\mathbb{E}(Y) = 2$.

```r
Y = S
muY = 2
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.2523751
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.02201785
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 88.34979
```

Εναλλακτικά, υπολογίζουμε ότι $\mathbb{P}(R + S > 4) = \mathbb{E}\left[\mathbb{P}\left(R + S > 4 \mid R\right)\right] = \mathbb{E}\left[\min\left\{e^{-(4-R)/2}, 1\right\}\right]$.

```r
W = pmin(exp(-(4 - R)/2), 1)
I = mean(W)
print(I)
```

```
## [1] 0.2530393
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.02831303
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 85.01885
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = R$ με $\mathbb{E}(Y) = 1$.

```r
Y = R
muY = 1
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
```

```
print(I)
```

```
## [1] 0.2525319
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.002096755
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 98.89055
```

Παράδειγμα 4.10. Έστω δύο ανεξάρτητες τυχαίες μεταβλητές $R, S \sim \text{Bin}(k,p)$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(e^{RS}\right)$. Για $r \in \{0, 1, \dots, k\}$, υπολογίζουμε ότι:

$$\mathbb{E}\left(e^{RS} \middle| R = r\right) = \mathbb{E}\left(e^{rS} \middle| R = r\right) = \mathbb{E}\left(e^{rS}\right) = \sum_{s=0}^{k} \binom{k}{s} p^s (1-p)^{k-s} e^{rs}$$
$$= \sum_{s=0}^{k} \binom{k}{s} \left(pe^r\right)^s (1-p)^{k-s} = \left(pe^r + 1 - p\right)^k .$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{E}\left(e^{RS}\right) = \mathbb{E}\left[\mathbb{E}\left(e^{RS} \middle| R\right)\right] = \mathbb{E}\left[\left(pe^R + 1 - p\right)^k\right].$$

```
n = 1e+05
k = 2
p = 0.1
U = matrix(runif(n * k), n)
R = rowSums(U < p)
V = matrix(runif(n * k), n)
S = rowSums(V < p)
X = exp(R * S)
I = mean(X)
print(I)
```

```
## [1] 1.081712
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.5129571
```

```
W = (p * exp(R) + 1 - p)^k
I = mean(W)
print(I)
```

```
## [1] 1.084086
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.04611129
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 91.01069
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = R$ με $\mathbb{E}(Y) = kp$.

```r
Y = R
muY = k * p
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 1.083844
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.007070317
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 98.62166
```

Λήμμα 4.1. Έστω τυχαία μεταβλητή $S \sim \text{Gamma}(k, \lambda)$ με $k \in \mathbb{N}$. Για $s > 0$, γνωρίζουμε ότι:

$$F_S(s) = 1 - \sum_{j=0}^{k-1} e^{-\lambda s} \frac{(\lambda s)^j}{j!}.$$

Απόδειξη. Έστω διαδικασία Poisson $\{N(t) : t \geqslant 0\}$ με ρυθμό λ και χρόνους αφίξεων $S_1, S_2, \ldots$. Τότε, γνωρίζουμε ότι $S_k \sim \text{Gamma}(k, \lambda)$. Παρατηρούμε ότι:

$$F_{S_k}(s) = \mathbb{P}\left(S_k \leqslant s\right) = \mathbb{P}\left[N(s) \geqslant k\right] = 1 - \mathbb{P}\left[N(s) \leqslant k-1\right] = 1 - \sum_{j=0}^{k-1} \mathbb{P}\left[N(s) = j\right] = 1 - \sum_{j=0}^{k-1} e^{-\lambda s} \frac{(\lambda s)^j}{j!}.$$

Παράδειγμα 4.11. Έστω τυχαία μεταβλητή $K \sim \text{Poisson}(\lambda)$. Θεωρούμε μία ακολουθία ανεξάρτητων τυχαίων μεταβλητών $R_1, R_2, \cdots \sim \text{Exp}(\mu)$ ανεξάρτητη από την K. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}\left(S_K > s\right)$, όπου:

$$S_K = \sum_{\ell=1}^{K} R_\ell.$$

Για $k \in \mathbb{N}$, παρατηρούμε ότι $S_k \sim \text{Gamma}(k, \mu)$. Επομένως, υπολογίζουμε ότι:

$$\mathbb{P}\left(S_K > s \mid K = k\right) = \mathbb{P}\left(S_k > s \mid K = k\right) = \mathbb{P}\left(S_k > s\right) = \sum_{j=0}^{k-1} e^{-\mu s} \frac{(\mu s)^j}{j!}.$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{P}\left(S_K > s\right) = \mathbb{E}\left(\mathbb{1}_{\{S_K > s\}}\right) = \mathbb{E}\left[\mathbb{E}\left(\mathbb{1}_{\{S_K > s\}} \middle| K\right)\right] = \mathbb{E}\left[\mathbb{P}\left(S_K > s \mid K\right)\right] = \mathbb{E}\left[\sum_{j=0}^{K-1} e^{-\mu s} \frac{(\mu s)^j}{j!}\right].$$

```
n = 1e+05
lambda = 4
mu = 6
s = 1
K = numeric(n)
S = numeric(n)
for (i in 1:n) {
    U = runif(1)
    pmf = exp(-lambda)
    cdf = pmf
    while (U > cdf) {
        K[i] = K[i] + 1
        pmf = pmf * lambda/K[i]
        cdf = cdf + pmf
    }
    V = runif(K[i])
    R = -log(V)/mu
    S[i] = sum(R)
}
X = S > s
I = mean(X)
print(I)
```

```
## [1] 0.21204
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.167079
```

```
W = numeric(n)
for (i in 1:n) {
    if (K[i] > 0) {
        pmf = exp(-mu * s)
        for (j in 0:(K[i] - 1)) {
            W[i] = W[i] + pmf
```

```
            pmf = pmf * mu * s/(j + 1)
        }
    }
}
I = mean(W)
print(I)
```

```
## [1] 0.2116497
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.04840946
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 71.02601
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = K$ με $\mathbb{E}(Y) = \lambda$.

```
Y = K
muY = lambda
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 0.2125292
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.003967733
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 97.62524
```

Εναλλακτικά, ορίζουμε $M = \min\{m \in \mathbb{N} : S_m > s\}$. Για $m \in \mathbb{N}$, υπολογίζουμε ότι:

$$\begin{aligned}\mathbb{P}(S_K > s \mid M = m) &= \mathbb{P}(K \geqslant M \mid M = m) = \mathbb{P}(K \geqslant m \mid M = m)\\ &= 1 - \mathbb{P}(K \leqslant m-1) = 1 - \sum_{j=0}^{m-1} e^{-\lambda}\frac{\lambda^j}{j!}.\end{aligned}$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{P}(S_K > s) = \mathbb{E}\left(\mathbb{1}_{\{S_K > s\}}\right) = \mathbb{E}\left[\mathbb{E}\left(\mathbb{1}_{\{S_K > s\}} \middle| M\right)\right] = \mathbb{E}\left[\mathbb{P}(S_K > s \mid M)\right] = \mathbb{E}\left(1 - \sum_{j=0}^{M-1} e^{-\lambda}\frac{\lambda^j}{j!}\right).$$

```r
n = 1e+05
lambda = 4
mu = 6
s = 1
S = numeric(n)
M = numeric(n)
W = numeric(n)
for (i in 1:n) {
    while (S[i] <= s) {
        U = runif(1)
        R = -log(U)/mu
        S[i] = S[i] + R
        M[i] = M[i] + 1
    }
    W[i] = 1
    pmf = exp(-lambda)
    for (j in 0:(M[i] - 1)) {
        W[i] = W[i] - pmf
        pmf = pmf * lambda/(j + 1)
    }
}
I = mean(W)
print(I)
```

```
## [1] 0.2129754
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.05250765
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 68.57317
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή ελέγχου την $Y = S_M - M/\mu$. Για $m \in \mathbb{N}$, υπολογίζουμε ότι:

$$\mathbb{E}\left(Y \mid M = m\right) = \mathbb{E}\left(S_m - \frac{m}{\mu}\middle| M = m\right) = \mathbb{E}\left(S_m\right) - \frac{m}{\mu} = 0.$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι $\mathbb{E}(Y) = \mathbb{E}\left[\mathbb{E}\left(Y \mid M\right)\right] = 0$.

```r
Y = S - M/mu
muY = 0
VarY = var(Y)
Cov = cov(W, Y)
c = -Cov/VarY
W = W + c * (Y - muY)
```

```r
I = mean(W)
print(I)
```

```
## [1] 0.2126705
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.01807493
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 89.18181
```

Παράδειγμα 4.12. Έστω ένα σύστημα εξυπηρέτησης $M/M/1/k$, όπου η διαδικασία αφίξεων $\{N(t) : t \geqslant 0\}$ είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\mathrm{Exp}(\mu)$. Θέλουμε να εκτιμήσουμε το μέσο πλήθος X των χαμένων πελατών μέχρι τη χρονική στιγμή T^*. Ορίζουμε S τον συνολικό χρόνο μέχρι τη στιγμή T^* που το σύστημα είναι πλήρες. Τότε, παρατηρούμε ότι $X \overset{d}{=} N(S)$. Σύμφωνα με το θεώρημα διπλής μέσης τιμής, παίρνουμε ότι:

$$\mathbb{E}(X) = \mathbb{E}\left[N(S)\right] = \mathbb{E}\left[\mathbb{E}\left(N(S) \mid S\right)\right] = \mathbb{E}\left(\lambda S\right).$$

```r
n = 1000
lambda = 4
mu = 6
k = 10
Tstar = 100
X = numeric(n)
S = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    t = A
    while (t < Tstar) {
        if (t == A) {
            if (Q < k) {
                Q = Q + 1
                if (Q == k) {
                  S[i] = S[i] - t
                }
            } else {
                X[i] = X[i] + 1
            }
            U = runif(1)
            A = t - log(U)/lambda
```

```
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
            }
        } else {
            Q = Q - 1
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
            } else {
                D = Inf
            }
            if (Q == k - 1) {
                S[i] = S[i] + t
            }
        }
        t = min(A, D)
    }
    if (Q == k) {
        S[i] = S[i] + Tstar
    }
}
I = mean(X)
print(I)
```

```
## [1] 2.221
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 9.140159
```

```
W = lambda * S
I = mean(W)
print(I)
```

```
## [1] 2.231415
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 7.261257
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 20.55656
```

Παράδειγμα 4.13. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson με ρυθμό λ και οι χρόνοι εξυπηρέτησης ακολουθούν την κατανομή $\mathrm{Exp}(\mu)$. Θέλουμε να εκτιμήσουμε τον αναμενόμενο συνολικό χρόνο παραμονής των πρώτων N^* πελατών στο σύστημα. Έστω S_j ο χρόνος παραμονής j-οστού πελάτη στο σύστημα, R_j ο χρόνος εξυπηρέτησης του j-οστού πελάτη και M_j το πλήθος των πελατών που είναι παρόντες στο σύστημα τη χρονική στιγμή άφιξης του j-οστού πελάτη. Τότε, ορίζουμε:

$$X = \sum_{j=1}^{N^*} S_j, \quad Y = \sum_{j=1}^{N^*} R_j, \quad K = \sum_{j=1}^{N^*} M_j.$$

Θα χρησιμοποιήσουμε αρχικά ως μεταβλητή ελέγχου την Y με $\mathbb{E}(Y) = N^*/\mu$.

```
n = 10000
lambda = 4
mu = 6
Nstar = 10
X = numeric(n)
Y = numeric(n)
K = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    D = Inf
    N = 0
    arrivals = numeric(0)
    while (N < Nstar || Q > 0) {
        t = min(A, D)
        if (t == A) {
            K[i] = K[i] + Q
            Q = Q + 1
            N = N + 1
            if (N < Nstar) {
                U = runif(1)
                A = t - log(U)/lambda
            } else {
                A = Inf
            }
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/mu
                Y[i] = Y[i] + D - t
            }
            arrivals = c(arrivals, t)
        } else {
            Q = Q - 1
```

```
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
                Y[i] = Y[i] + D - t
            } else {
                D = Inf
            }
            X[i] = X[i] + t - arrivals[1]
            arrivals = arrivals[-1]
        }
    }
}
I = mean(X)
print(I)
```

```
## [1] 3.204967
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 4.15866
```

```
muY = Nstar/mu
VarY = var(Y)
Cov = cov(X, Y)
c = -Cov/VarY
W = X + c * (Y - muY)
I = mean(W)
print(I)
```

```
## [1] 3.212926
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 1.587515
```

```
rho = Cov/sqrt(VarX * VarY)
print(rho)
```

```
## [1] 0.7863362
```

```
100 * rho^2
```

```
## [1] 61.83246
```

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, υπολογίζουμε εναλλακτικά ότι:

$$\mathbb{E}(X) = \sum_{j=1}^{N^*} \mathbb{E}\left(S_j\right) = \sum_{j=1}^{N^*} \mathbb{E}\left[\mathbb{E}\left(S_j \mid M_j\right)\right] = \sum_{j=1}^{N^*} \mathbb{E}\left(\frac{M_j+1}{\mu}\right) = \mathbb{E}\left(\frac{1}{\mu}\sum_{j=1}^{N^*} M_j + \frac{N^*}{\mu}\right) = \mathbb{E}\left(\frac{K+N^*}{\mu}\right).$$

```
W = (K + Nstar)/mu
I = mean(W)
print(I)
```

```
## [1] 3.2149
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 1.452951
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 65.06203
```

Δειγματοληψία Σπουδαιότητας

Έστω τυχαίες μεταβλητές R, Y με συναρτήσεις πυκνότητας πιθανότητας $f(x)$, $g(x)$ και στηρίγματα S_f, S_g αντίστοιχα. Θεωρούμε μία συνάρτηση $h : S_f \to \mathbb{R}$. Υποθέτουμε ότι είτε $S_g \subseteq S_f$ και $h(x) = 0$ για $x \in S_f \setminus S_g$ είτε $S_f \subseteq S_g$. Ενδιαφερόμαστε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}[h(R)]$. Αν θέσουμε $\phi(x) = \frac{h(x)f(x)}{g(x)}$, τότε παρατηρούμε ότι:

$$\mathbb{E}[h(R)] = \int_{S_f} h(x)f(x)dx = \int_{S_g} \frac{h(x)f(x)}{g(x)} g(x)dx = \int_{S_g} \phi(x)g(x)dx = \mathbb{E}[\phi(Y)].$$

Ο στόχος της μεθόδου δειγματοληψίας σπουδαιότητας είναι να επιλέξουμε την τυχαία μεταβλητή Y με τέτοιον τρόπο ώστε $f(x) \gg g(x)$ αν και μόνο αν $|h(x)| \approx 0$ και $f(x) \ll g(x)$ αν και μόνο αν $|h(x)| \gg 0$. Με αυτόν τον τρόπο επιτυγχάνουμε να ελαχιστοποιήσουμε τη διασπορά της τυχαίας μεταβλητής $\phi(Y)$.

Παράδειγμα 4.14. Έστω τυχαία μεταβλητή $Z \sim \mathcal{N}(0,1)$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(Z > 3)$. Θα χρησιμοποιήσουμε ως συνάρτηση σπουδαιότητας την $g(x) = e^{-(x-3)}$ για $x > 3$. Τότε, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = \frac{1}{e^{-(x-3)}} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} = \frac{e^{-x^2/2+x-3}}{\sqrt{2\pi}}.$$

```
n = 1e+05
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = Z > 3
I = mean(X)
```

```
print(I)
```

```
## [1] 0.0014
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.00139804
```

```
U = runif(n)
Y = 3 - log(U)
W = exp(-Y^2/2 + Y - 3)/sqrt(2 * pi)
I = mean(W)
print(I)
```

```
## [1] 0.001350963
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 1.850184e-06
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 99.86766
```

Παράδειγμα 4.15. Έστω τυχαία μεταβλητή $S \sim \text{Gamma}(3, 1)$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(\max\{X-8, 0\}\right)$. Θα χρησιμοποιήσουμε ως συνάρτηση σπουδαιότητας την $g(x) = e^{-(x-8)}$ για $x > 8$. Τότε, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = \frac{x-8}{e^{-(x-8)}}\frac{1}{\Gamma(3)}x^2e^{-x} = \frac{x^2(x-8)}{2e^8}.$$

```
n = 1e+05
U = matrix(runif(3 * n), n)
R = -log(U)
S = rowSums(R)
X = pmax(S - 8, 0)
I = mean(X)
print(I)
```

```
## [1] 0.01686379
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.04255399
```

```
V = runif(n)
Y = 8 - log(V)
W = Y^2 * (Y - 8)/(2 * exp(8))
```

```
I = mean(W)
print(I)
```

```
## [1] 0.01721288
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.0006926254
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 98.37236
```

Παράδειγμα 4.16. Έστω τυχαία μεταβλητή $U \sim \text{Unif}[0,1]$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left[\left(1-U^2\right)e^U\right]$. Θα χρησιμοποιήσουμε αρχικά ως μεταβλητή σπουδαιότητας την $Y \sim \text{Beta}(2,1)$. Για $x \in [0,1]$, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = \frac{\left(1-x^2\right)e^x}{2x}.$$

```
n = 1e+05
U = runif(n)
X = (1 - U^2) * exp(U)
I = mean(X)
print(I)
```

```
## [1] 0.9993458
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.09743515
```

```
V = runif(n)
Y = V^(1/2)
W = (1 - Y^2) * exp(Y)/(2 * Y)
I = mean(W)
print(I)
```

```
## [1] 1.007208
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 3.706268
```

```
100 * (VarW - VarX)/VarX
```

```
## [1] 3703.831
```

Παρατηρούμε ότι η διασπορά της εκτιμήτριας της μέσης τιμής $\mathbb{E}\left[\left(1-U^2\right)e^U\right]$ αυξήθηκε δραματικά για τη συγκεκριμένη επιλογή μεταβλητής σπουδαιότητας. Για $x \in [0,1]$, υπολογίζουμε ότι $h'(x) = \left(1-2x-x^2\right)e^x$

και $h''(x) = -\left(x^2 + 4x + 1\right) e^x < 0$. Δηλαδή, η συνάρτηση h μεγιστοποιείται στο σημείο $x^* = \sqrt{2} - 1$. Γνωρίζουμε ότι η συνάρτηση πυκνότητας πιθανότητας της κατανομής $\text{Beta}(a, b)$ για $a, b > 1$ μεγιστοποιείται στο σημείο:

$$x^* = \frac{a - 1}{a + b - 2}.$$

Αν επιλέξουμε $a = \sqrt{2}$ και $b = 3 - \sqrt{2}$, τότε οι συναρτήσεις $h(x)$ και $g(x)$ θα μεγιστοποιούνται στο ίδιο σημείο. Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή σπουδαιότητας την $Y \sim \text{Beta}\left(\sqrt{2}, 3 - \sqrt{2}\right)$. Για $x \in [0, 1]$, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x) f(x)}{g(x)} = \left(1 - x^2\right) e^x \frac{\Gamma\left(\sqrt{2}\right) \Gamma\left(3 - \sqrt{2}\right)}{2x^{\sqrt{2}-1}(1 - x)^{2-\sqrt{2}}}.$$

```
M = dbeta(sqrt(2) - 1, sqrt(2), 3 - sqrt(2))
Y = numeric(n)
for (i in 1:n) {
    Y[i] = runif(1)
    U = runif(1)
    V = M * U
    while (dbeta(Y[i], sqrt(2), 3 - sqrt(2)) < V) {
        Y[i] = runif(1)
        U = runif(1)
        V = M * U
    }
}
W = (1 - Y^2) * exp(Y)/dbeta(Y, sqrt(2), 3 - sqrt(2))
I = mean(W)
print(I)
```

```
## [1] 1.000158
```

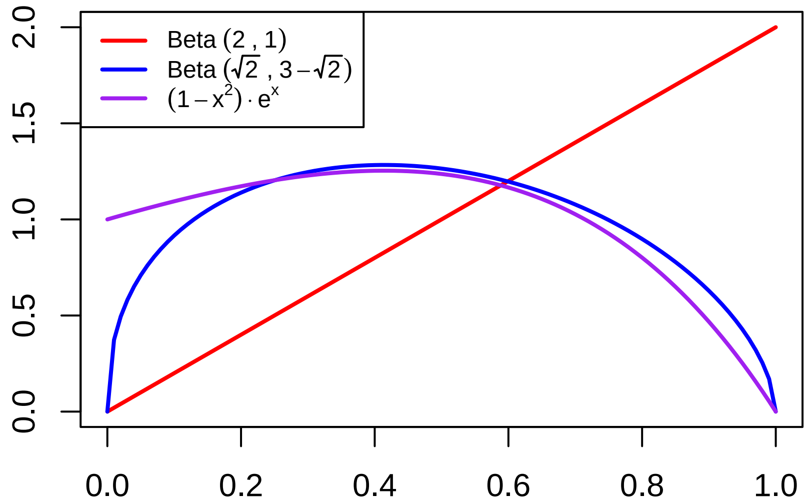
```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.05403956
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 44.53792
```

```
curve(dbeta(x, 2, 1), col = "red", lwd = 2, xlab = NA, ylab = NA, xlim = c(0,
    1))
curve(dbeta(x, sqrt(2), 3 - sqrt(2)), add = TRUE, col = "blue", lwd = 2)
curve((1 - x^2) * exp(x), add = TRUE, col = "purple", lwd = 2)
legend("topleft", c(expression("Beta" ~ (2 ~ "," ~ 1)), expression("Beta" ~
    (sqrt(2) ~ "," ~ 3 - sqrt(2))), expression((1 - x^2) %.% e^x)), col = c("red",
    "blue", "purple"), lty = c(1, 1, 1), lwd = c(2, 2, 2), cex = 0.75)
```



Σημείωση 4.3. Έστω τυχαία μεταβλητή $X \sim t_\nu$. Για $x \in \mathbb{R}$, γνωρίζουμε ότι:

$$f_X(x) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}.$$

Για $k \in \mathbb{N}$, γνωρίζουμε ότι:

$$\Gamma\left(k+\frac{1}{2}\right) = \frac{(2k)!}{4^k k!}\sqrt{\pi}.$$

Για $\nu = 1$, γνωρίζουμε ότι $X \sim t_1 \equiv \text{Cauchy}(0,1)$. Τότε, παίρνουμε ότι:

$$f_X(x) = \frac{1}{\pi\left(1+x^2\right)}, \quad F_X(x) = \frac{1}{\pi}\arctan x + \frac{1}{2}, \quad F_X^{-1}(u) = \tan\left[\pi\left(u-\frac{1}{2}\right)\right].$$

Παράδειγμα 4.17. Έστω τυχαία μεταβλητή $S \sim t_3$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}\left(|S|\right)$. Θα χρησιμοποιήσουμε αρχικά ως μεταβλητή σπουδαιότητας την $Y \sim \text{Cauchy}(0,1)$. Για $x \in \mathbb{R}$, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = |x|\pi\left(1+x^2\right)\frac{1}{\sqrt{3\pi}\sqrt{\pi}/2}\left(1+\frac{x^2}{3}\right)^{-2} = \frac{2|x|\left(1+x^2\right)}{\sqrt{3}}\left(1+\frac{x^2}{3}\right)^{-2}.$$

```
n = 1e+05
M = (3/2)^(3/2) * 2/sqrt(pi) * exp(-0.5)
Y = numeric(n)
for (i in 1:n) {
    W = runif(1)
    Y[i] = -3 * log(W)
    U = runif(1)
    V = M * dexp(Y[i], 1/3) * U
    while (dchisq(Y[i], 3) < V) {
        W = runif(1)
        Y[i] = -3 * log(W)
        U = runif(1)
        V = M * dexp(Y[i], 1/3) * U
```

```
    }
}
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
S = sqrt(3/Y) * Z
X = abs(S)
I = mean(X)
print(I)
```

```
## [1] 1.099443
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 1.727892
```

```
U = runif(n)
Y = tan(pi * (U - 0.5))
W = 2 * abs(Y) * (1 + Y^2)/sqrt(3) * (1 + Y^2/3)^(-2)
I = mean(W)
print(I)
```

```
## [1] 1.102958
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.5165322
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 70.10622
```

Θα χρησιμοποιήσουμε στη συνέχεια ως μεταβλητή σπουδαιότητας την $Y \sim \mathcal{N}(0,1)$. Για $x \in \mathbb{R}$, υπολογίζουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = |x|\sqrt{2\pi}e^{x^2/2}\frac{1}{\sqrt{3\pi}\sqrt{\pi}/2}\left(1+\frac{x^2}{3}\right)^{-2} = \frac{4|x|e^{x^2/2}}{\sqrt{6\pi}}\left(1+\frac{x^2}{3}\right)^{-2}.$$

```
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Y = sqrt(D) * c(cos(Theta), sin(Theta))
W = 4 * abs(Y) * exp(Y^2/2)/sqrt(6 * pi) * (1 + Y^2/3)^(-2)
I = mean(W)
print(I)
```

```
## [1] 1.010371
```

```
VarW = mean((W - I)^2)
print(VarW)
```

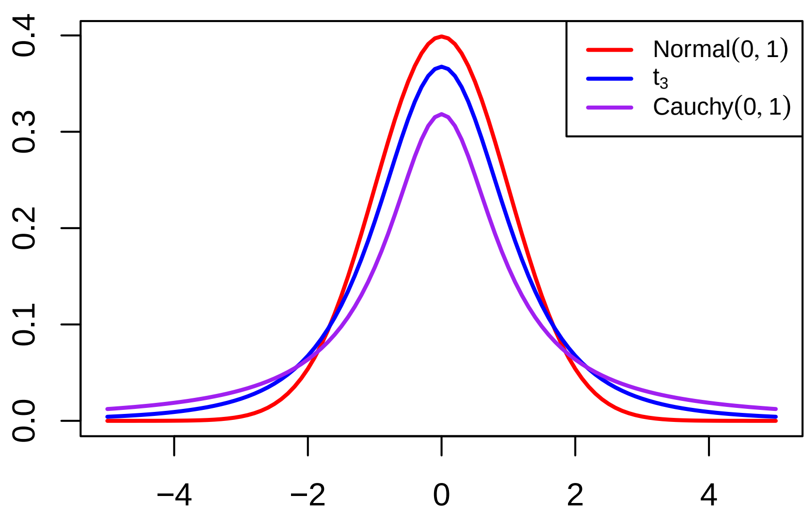
```
## [1] 304.6457
```

```
100 * (VarW - VarX)/VarX
```

```
## [1] 17531.06
```

Παρατηρούμε ότι η διασπορά της εκτιμήτριας της $\mathbb{E}(|S|)$ αυξήθηκε δραματικά για τη συγκεκριμένη επιλογή μεταβλητής σπουδαιότητας. Σχετικά με τη συνάρτηση $h(x) = |x|$ παρατηρούμε ότι $h(x) \approx 0$ για $x \approx 0$ και $h(x) \gg 0$ για $|x| \gg 0$. Αν $Y \sim \text{Cauchy}(0, 1)$, τότε $f(x) \gg g(x)$ για $x \approx 0$ και $f(x) \ll g(x)$ για $|x| \gg 0$, δηλαδή η επιλογή της μεταβλητής σπουδαιότητας είναι κατάλληλη. Αν $Y \sim \mathcal{N}(0, 1)$, τότε $f(x) \ll g(x)$ για $x \approx 0$ και $f(x) \gg g(x)$ για $|x| \gg 0$, δηλαδή η μέθοδος της δειγματοληψίας σπουδαιότητας οδηγεί σε εκτιμήτρια με πολύ μεγαλύτερη διασπορά από την αρχική.

```
curve(dnorm(x), col = "red", lwd = 2, xlab = NA, ylab = NA, xlim = c(-5, 5))
curve(dt(x, 3), add = TRUE, col = "blue", lwd = 2)
curve(dcauchy(x), add = TRUE, col = "purple", lwd = 2)
legend("topright", c(expression(Normal(0, 1)), expression(t[3]), expression(Cauchy(0,
    1))), col = c("red", "blue", "purple"), lty = c(1, 1, 1), lwd = c(2, 2,
    2), cex = 0.75)
```



Παράδειγμα 4.18. Έστω ανεξάρτητες τυχαίες μεταβλητές $S \sim \text{Exp}(2)$ και $R \sim \text{Exp}(1)$. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}(\max\{S + R - 7, 0\})$. Θα χρησιμοποιήσουμε ως πυκνότητα σπουδαιότητας την $g(x, y) = 2e^{-2x}e^{-(y-\max\{7-x,0\})}$ για $x > 0$ και $y > \max\{7 - x, 0\}$. Τότε, υπολογίζουμε ότι:

$$\phi(x, y) = \frac{h(x, y)f(x, y)}{g(x, y)} = (x + y - 7)\frac{2e^{-2x}e^{-y}}{2e^{-2x}e^{-(y-\max\{7-x,0\})}} = \frac{x + y - 7}{e^{\max\{7-x,0\}}}.$$

```
n = 1e+05
U = runif(n)
S = -log(U)/2
V = runif(n)
R = -log(V)
X = pmax(S + R - 7, 0)
I = mean(X)
print(I)
```

```
## [1] 0.001809774
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.003704638
```

```
U = runif(n)
S = -log(U)/2
V = runif(n)
Y = pmax(7 - S, 0) - log(V)
W = (S + Y - 7)/exp(pmax(7 - S, 0))
I = mean(W)
print(I)
```

```
## [1] 0.001837356
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 2.769405e-05
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 99.25245
```

Παράδειγμα 4.19. Έστω ακολουθία ανεξάρτητων τυχαίων μεταβλητών $R_1, R_2, \cdots \sim \mathcal{N}\left(\mu, \sigma^2\right)$ με $\mu < 0$ και $A, B > 0$. Ορίζουμε τις τυχαίες μεταβλητές:

$$S_m = \sum_{j=1}^{m} R_j, \quad M = \min\left\{m \in \mathbb{N} : S_m < -A \text{ or } S_m > B\right\}.$$

Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}\left(S_M > B\right)$. Θα χρησιμοποιήσουμε τις μεταβλητές σπουδαιότητας $Y_1, Y_2, \cdots \sim \mathcal{N}\left(-\mu, \sigma^2\right)$. Τότε, παίρνουμε ότι:

$$\phi(x) = \frac{h(x)f(x)}{g(x)} = \mathbb{1}_{\{S_M > B\}} \prod_{j=1}^{M} \frac{f_{R_j}(x_j)}{f_{Y_j}(x_j)} = \mathbb{1}_{\{S_M > B\}} \prod_{j=1}^{M} \exp\left\{-\frac{1}{2\sigma^2}\left(x_j - \mu\right)^2\right\} \exp\left\{\frac{1}{2\sigma^2}\left(x_j + \mu\right)^2\right\}$$
$$= \mathbb{1}_{\{S_M > B\}} \exp\left\{\sum_{j=1}^{M} 2\mu x_j\right\} = \mathbb{1}_{\{S_M > B\}} \exp\left\{\frac{2\mu S_M}{\sigma^2}\right\}.$$

```r
n = 10000
mu = -3
sigma = 2
A = 6
B = 3
lambda = 1/sigma
M = sqrt(2 * exp(1)/pi)
S = numeric(n)
for (i in 1:n) {
    while (S[i] >= -A && S[i] <= B) {
        W = runif(1)
        Y = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y - mu), lambda)/2 * U
        while (dnorm(Y, mu, sigma) < V) {
            W = runif(1)
            Y = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
            U = runif(1)
            V = M * dexp(abs(Y - mu), lambda)/2 * U
        }
        S[i] = S[i] + Y
    }
}
X = S > B
I = mean(X)
print(I)
```

```
## [1] 0.0022
```

```r
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.00219516
```

```r
S = numeric(n)
for (i in 1:n) {
    while (S[i] >= -A && S[i] <= B) {
        W = runif(1)
        Y = ifelse(W <= 0.5, -mu + log(2 * W)/lambda, -mu - log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y + mu), lambda)/2 * U
        while (dnorm(Y, -mu, sigma) < V) {
            W = runif(1)
            Y = ifelse(W <= 0.5, -mu + log(2 * W)/lambda, -mu - log(2 * (1 -
                W))/lambda)
```

```
            U = runif(1)
            V = M * dexp(abs(Y + mu), lambda)/2 * U
        }
        S[i] = S[i] + Y
    }
}
W = (S > B) * exp(2 * mu * S/sigma^2)
I = mean(W)
print(I)
```

```
## [1] 0.002104925
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 7.516815e-06
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 99.65757
```

Ορισμός 4.1. Έστω τυχαία μεταβλητή X με στήριγμα S, συνάρτηση πυκνότητας πιθανότητας f και ροπογεννήτρια $M(t) = \mathbb{E}\left[e^{tX}\right]$ για $t \in \mathbb{R}$. Για $x \in S$, ορίζουμε την κεκλιμένη συνάρτηση πυκνότητας πιθανότητας της f ως:

$$f_t(x) = \frac{e^{tx} f(x)}{M(t)}.$$

Σημείωση 4.4. Έστω ανεξάρτητες τυχαίες μεταβλητές $X \sim f$ και $Y \sim f_t$. Αν $t > 0$, τότε $\mathbb{E}(Y) > \mathbb{E}(X)$. Διαφορετικά, $\mathbb{E}(Y) < \mathbb{E}(X)$.

Παράδειγμα 4.20. Έστω ανεξάρτητες τυχαίες μεταβλητές $R_1, \dots, R_k \sim \text{Exp}(1)$. Ορίζουμε την τυχαία μεταβλητή $S = R_1 + \cdots + R_k$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(S > a)$ με $a > k$. Για $t < 1$, γνωρίζουμε ότι $M_j(t) = \mathbb{E}\left(e^{tR_j}\right) = \frac{1}{1-t}$. Για $x > 0$, ορίζουμε τις κεκλιμένες συναρτήσεις σπουδαιότητας:

$$f_t^{(j)}(x) = \frac{e^{tx} f_{R_j}(x)}{M_j(t)} = (1-t)e^{tx}e^{-x} = (1-t)e^{-(1-t)x}.$$

Δηλαδή, οδηγούμαστε στις κεκλιμένες μεταβλητές σπουδαιότητας $Y_1, \dots, Y_k \sim \text{Exp}(1-t)$. Τότε, παίρνουμε ότι:

$$\begin{aligned}
\phi(x) = \frac{h(x)f(x)}{g(x)} &= \mathbb{1}_{\{S>a\}} \prod_{j=1}^{k} \frac{f_{R_j}(x_j)}{f_t^{(j)}(x_j)} = \mathbb{1}_{\{S>a\}} \prod_{j=1}^{k} \frac{e^{-x_j}}{(1-t)e^{-(1-t)x_j}} \\
&= \frac{\mathbb{1}_{\{S>a\}}}{(1-t)^k} \exp\left\{-\sum_{j=1}^{k} t x_j\right\} = \frac{\mathbb{1}_{\{S>a\}} e^{-tS}}{(1-t)^k}.
\end{aligned}$$

Μία κατάλληλη τιμή για την παράμετρο t προκύπτει αν θέσουμε $\mathbb{E}_t(S) = a$. Τότε, παίρνουμε ότι:

$$\sum_{j=1}^{k} \frac{1}{1-t} = a \quad \Rightarrow \quad t = 1 - \frac{k}{a}.$$

```r
n = 1e+05
k = 4
a = 10
U = matrix(runif(k * n), n)
R = -log(U)
S = rowSums(R)
X = S > a
I = mean(X)
print(I)
```

```
## [1] 0.00991
```

```r
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.009811792
```

```r
t = 1 - k/a
print(t)
```

```
## [1] 0.6
```

```r
V = matrix(runif(k * n), n)
Y = -log(V)/(1 - t)
S = rowSums(Y)
W = (S > a) * exp(-t * S)/(1 - t)^k
I = mean(W)
print(I)
```

```
## [1] 0.0103581
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.0004474145
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 95.44003
```

Παράδειγμα 4.21. Έστω ανεξάρτητες τυχαίες μεταβλητές $R_1, \dots, R_k \sim \text{Bernoulli}(p)$. Ορίζουμε την τυχαία μεταβλητή $S = R_1 + \cdots + R_k$. Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}(S \geqslant a)$ με $a < k$. Για $t \in \mathbb{R}$, γνωρίζουμε ότι $M_j(t) = \mathbb{E}\left(e^{tR_j}\right) = 1 - p + pe^t$. Για $x \in \{0, 1\}$, ορίζουμε τις κεκλιμένες συναρτήσεις σπουδαιότητας:

$$f_t^{(j)}(x) = \frac{e^{tx} f_{R_j}(x)}{M_j(t)} = \frac{e^{tx} p^x (1-p)^{1-x}}{1 - p + pe^t} = \left(\frac{pe^t}{1 - p + pe^t}\right)^x \left(\frac{1-p}{1 - p + pe^t}\right)^{1-x}.$$

Δηλαδή, οδηγούμαστε στις κεκλιμένες μεταβλητές σπουδαιότητας $Y_1, \dots, Y_k \sim \text{Bernoulli}\left(\frac{pe^t}{1-p+pe^t}\right)$. Τότε, παίρνουμε ότι:

$$\begin{aligned}\phi(x) = \frac{h(x)f(x)}{g(x)} &= \mathbb{1}_{\{S\geqslant a\}}\prod_{j=1}^{k}\frac{f_{R_j}(x_j)}{f_t^{(j)}(x_j)} = \mathbb{1}_{\{S\geqslant a\}}\prod_{j=1}^{k}p^{x_j}(1-p)^{1-x_j}\frac{1-p+pe^t}{e^{tx_j}p^{x_j}(1-p)^{1-x_j}} \\ &= \mathbb{1}_{\{S\geqslant a\}}\left(1-p+pe^t\right)^k \exp\left\{-\sum_{j=1}^{k}tx_j\right\} = \mathbb{1}_{\{S\geqslant a\}}\left(1-p+pe^t\right)^k e^{-tS}\end{aligned}.$$

Μία κατάλληλη τιμή για την παράμετρο t προκύπτει αν θέσουμε $\mathbb{E}_t(S) = a$. Τότε, παίρνουμε ότι:

$$\sum_{j=1}^{k}\frac{pe^t}{1-p+pe^t} = a \quad\Rightarrow\quad t = \log\frac{a(1-p)}{p(k-a)}.$$

```
n = 1e+05
p = 0.4
k = 20
a = 16
U = matrix(runif(k * n), n)
S = rowSums(U < p)
X = S >= a
I = mean(X)
print(I)
```

```
## [1] 0.00032
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.0003198976
```

```
t = log(a * (1 - p)/(p * (k - a)))
print(t)
```

```
## [1] 1.791759
```

```
V = matrix(runif(k * n), n)
S = rowSums(V < p * exp(t)/(1 - p + p * exp(t)))
W = (S >= a) * (1 - p + p * exp(t))^k * exp(-t * S)
I = mean(W)
print(I)
```

```
## [1] 0.0003173146
```

```
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 2.419507e-07
```

```
100 * (VarX - VarW)/VarX
```

```
## [1] 99.92437
```

Παράδειγμα 4.22. Έστω ένα σύστημα εξυπηρέτησης $M/M/1$, όπου η διαδικασία αφίξεων είναι Poisson ρυθμού λ με ενδιάμεσους χρόνους $P_1, P_2, \ldots$ και οι χρόνοι εξυπηρέτησης $R_1, R_2, \ldots$ ακολουθούν την κατανομή $\text{Exp}(\mu)$ με $\mu > \lambda$. Ορίζουμε W_j τον χρόνο αναμονής του j-οστού πελάτη στην ουρά. Επιπλέον, ορίζουμε:

$$S_W = \sum_{j=1}^{N^*} W_j, \quad S_P = \sum_{j=1}^{N^*} P_j, \quad S_R = \sum_{j=1}^{N*} R_j.$$

Θέλουμε να εκτιμήσουμε την πιθανότητα $\mathbb{P}\left(S_W > a\right)$. Θα χρησιμοποιήσουμε τις μεταβλητές σπουδαιότητας $\widetilde{P}_1, \ldots, \widetilde{P}_{N^*} \sim \text{Exp}(\mu)$ και $\widetilde{R}_1, \ldots, \widetilde{R}_{N^*} \sim \text{Exp}(\lambda)$. Τότε, παίρνουμε ότι:

$$\begin{aligned}\phi(x) = \frac{h(x)f(x)}{g(x)} &= \mathbb{1}_{\{S_W > a\}} \prod_{j=1}^{N^*} \frac{f_{P_j}(x_j) f_{R_j}(y_j)}{f_{\widetilde{P}_j}(x_j) f_{\widetilde{R}_j}(y_j)} = \mathbb{1}_{\{S_W > a\}} \prod_{j=1}^{N^*} \frac{\lambda e^{-\lambda x_j} \mu e^{-\mu y_j}}{\mu e^{-\mu x_j} \lambda e^{-\lambda y_j}} \\ &= \mathbb{1}_{\{S_W > a\}} \exp\left\{(\mu - \lambda)\sum_{j=1}^{N^*} x_j + (\lambda - \mu)\sum_{j=1}^{N^*} y_j\right\} = \mathbb{1}_{\{S_W > a\}} e^{(\mu-\lambda)(S_P - S_R)}.\end{aligned}$$

```
n = 10000
lambda = 4
mu = 6
Nstar = 5
a = 4
SW = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/lambda
    D = Inf
    N = 0
    arrivals = numeric(0)
    while (N < Nstar || Q > 0) {
        t = min(A, D)
        if (t == A) {
            Q = Q + 1
            N = N + 1
            if (N < Nstar) {
                U = runif(1)
                A = t - log(U)/lambda
            } else {
                A = Inf
            }
            if (Q == 1) {
```

```
                V = runif(1)
                D = t - log(V)/mu
            }
            arrivals = c(arrivals, t)
        } else {
            Q = Q - 1
            arrivals = arrivals[-1]
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/mu
                SW[i] = SW[i] + t - arrivals[1]
            } else {
                D = Inf
            }
        }
    }
}
X = SW > a
I = mean(X)
print(I)
```

```
## [1] 0.0019
```

```
VarX = mean((X - I)^2)
print(VarX)
```

```
## [1] 0.00189639
```

```
SW = numeric(n)
SP = numeric(n)
SR = numeric(n)
for (i in 1:n) {
    Q = 0
    U = runif(1)
    A = -log(U)/mu
    SP[i] = SP[i] - log(U)/mu
    D = Inf
    N = 0
    arrivals = numeric(0)
    while (N < Nstar || Q > 0) {
        t = min(A, D)
        if (t == A) {
            Q = Q + 1
            N = N + 1
            if (N < Nstar) {
```

```r
                U = runif(1)
                A = t - log(U)/mu
                SP[i] = SP[i] - log(U)/mu
            } else {
                A = Inf
            }
            if (Q == 1) {
                V = runif(1)
                D = t - log(V)/lambda
                SR[i] = SR[i] - log(V)/lambda
            }
            arrivals = c(arrivals, t)
        } else {
            Q = Q - 1
            arrivals = arrivals[-1]
            if (Q > 0) {
                V = runif(1)
                D = t - log(V)/lambda
                SR[i] = SR[i] - log(V)/lambda
                SW[i] = SW[i] + t - arrivals[1]
            } else {
                D = Inf
            }
        }
    }
}
W = (SW > a) * exp((mu - lambda) * (SP - SR))
I = mean(W)
print(I)
```

```
## [1] 0.002036983
```

```r
VarW = mean((W - I)^2)
print(VarW)
```

```
## [1] 0.0001958506
```

```r
100 * (VarX - VarW)/VarX
```

```
## [1] 89.67245
```

5 Αλγόριθμοι Markov Chain Monte Carlo

Δειγματολήπτης Gibbs

Θέλουμε να παράγουμε δείγμα $X^{(1)}, X^{(2)}, \dots, X^{(n)}$ από την από κοινού συνάρτηση πυκνότητας πιθανότητας $f_{X_1,X_2,\dots,X_k}$. Υποθέτουμε ότι είτε οι περιθώριες συναρτήσεις πυκνότητας πιθανότητας f_{X_j} δεν είναι εύκολα υπολογίσιμες είτε δεν είναι εύκολο να προσομοιώσουμε από αυτές, αλλά είναι εύκολο να προσομοιώσουμε από τις δεσμευμένες συναρτήσεις πυκνότητας πιθανότητας $f_{X_j|X_{-j}} \equiv f_{X_j|X_1,\dots,X_{j-1},X_{j+1},\dots,X_k}$.

Algorithm 5.1 Δειγματολήπτης Gibbs

Είσοδος: Δεσμευμένες συναρτήσεις κατανομής, ζητούμενο burn-in b και μέγεθος δείγματος n.

1: Θεωρούμε αρχικές τιμές $X_1^{(1)}, X_2^{(1)}, \dots, X_k^{(1)}$.

2: Για $i = 2, 3, \dots, b+n$, επαναλαμβάνουμε το παρακάτω βήμα:

i: Για $j = 1, 2, \dots, k$, προσομοιώνουμε τη $X_j^{(i)}$ από τη δεσμευμένη συνάρτηση κατανομής:

$$F_{X_j|X_1,\dots,X_{j-1},X_{j+1},\dots,X_k}\left(x \,\middle|\, X_1^{(i)}, \dots, X_{j-1}^{(i)}, X_{j+1}^{(i-1)}, \dots, X_k^{(i-1)}\right).$$

Έξοδος: Τυχαίο δείγμα $X^{(b+1)}, X^{(b+2)}, \dots, X^{(b+n)}$ από την από κοινού συνάρτηση κατανομής.

Η ακολουθία τυχαίων μεταβλητών $\left\{X^{(i)}\right\}$ αποτελεί μία Μαρκοβιανή διαδικασία διακριτού χρόνου με πυρήνα μεταβάσεων:

$$K\left(x^{(i)} \middle| x^{(i-1)}\right) = \prod_{j=1}^{k} f_{X_j|X_{-j}}\left(x_j^{(i)} \,\middle|\, x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right).$$

Θεώρημα 5.1. Αν η Μαρκοβιανή διαδικασία $\left\{X^{(i)}\right\}$ με πυρήνα μεταβάσεων $K\left(x^{(i)} \middle| x^{(i-1)}\right)$ και χώρο καταστάσεων S είναι αδιαχώριστη, τότε έχει ως μοναδική στάσιμη κατανομή την $f_{X_1,X_2,\dots,X_k}$.

Απόδειξη. Ορίζουμε τον αντίστροφο πυρήνα μετάβασης:

$$L\left(x^{(i-1)} \middle| x^{(i)}\right) = \prod_{j=1}^{k} f_{X_j|X_{-j}}\left(x_j^{(i-1)} \,\middle|\, x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right).$$

Τότε, παρατηρούμε ότι:

$$\begin{aligned}
f_{X_1,\dots,X_k}\left(x^{(i-1)}\right) K\left(x^{(i)} \middle| x^{(i-1)}\right) &= f_{X_1,\dots,X_k}\left(x^{(i-1)}\right) \prod_{j=1}^{k} f_{X_j|X_{-j}}\left(x_j^{(i)} \,\middle|\, x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right) \\
&= f_{X_1,\dots,X_k}\left(x^{(i-1)}\right) \prod_{j=1}^{k} \frac{f_{X_1,\dots,X_k}\left(x_1^{(i)}, \dots, x_j^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right)}{f_{X_{-j}}\left(x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right)} \\
&= f_{X_1,\dots,X_k}\left(x^{(i)}\right) \prod_{j=1}^{k} \frac{f_{X_1,\dots,X_k}\left(x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_j^{(i-1)}, \dots, x_k^{(i-1)}\right)}{f_{X_{-j}}\left(x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right)} \\
&= f_{X_1,\dots,X_k}\left(x^{(i)}\right) \prod_{j=1}^{k} f_{X_j|X_{-j}}\left(x_j^{(i-1)} \,\middle|\, x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_k^{(i-1)}\right) \\
&= f_{X_1,\dots,X_k}\left(x^{(i)}\right) L\left(x^{(i-1)} \middle| x^{(i)}\right).
\end{aligned}$$

Επομένως, παίρνουμε ότι:

$$\begin{aligned}\int_S f_{X_1,\dots,X_k}\left(x^{(i-1)}\right) K\left(x^{(i)} \middle| x^{(i-1)}\right) dx^{(i-1)} &= \int_S f_{X_1,\dots,X_k}\left(x^{(i)}\right) L\left(x^{(i-1)} \middle| x^{(i)}\right) dx^{(i-1)} \\ &= f_{X_1,\dots,X_k}\left(x^{(i)}\right) \int_S L\left(x^{(i-1)} \middle| x^{(i)}\right) dx^{(i-1)} = f_{X_1,\dots,X_k}\left(x^{(i)}\right).\end{aligned}$$

Εφόσον η συνάρτηση $f_{X_1,X_2,\dots,X_k}$ ικανοποιεί τις εξισώσεις ισορροπίας για τη Μαρκοβιανή διαδικασία $\left\{X^{(i)}\right\}$, συμπεραίνουμε ότι είναι η μοναδική στάσιμη κατανομή της. □

Σημείωση 5.1. Έστω S_j το στήριγμα της περιθώριας συνάρτησης κατανομής F_{X_j} για $j = 1, 2, \dots, k$. Αν $S = S_1 \times S_2 \times \cdots \times S_k$, τότε η Μαρκοβιανή διαδικασία $\left\{X^{(i)}\right\}$ με χώρο καταστάσεων S και πυρήνα μεταβάσεων $K\left(x^{(i)} \middle| x^{(i-1)}\right)$ είναι αδιαχώριστη.

Σημείωση 5.2.

i. Οι αλγόριθμοι Markov Chain Monte Carlo απαιτούν ένα αρχικό πλήθος επαναλήψεων μέχρι η Μαρκοβιανή διαδικασία που παράγουν να συγκλίνει στη στάσιμη κατανομή της, δηλαδή τη ζητούμενη από κοινού συνάρτηση πυκνότητας πιθανότητας. Αυτό το αρχικό πλήθος επαναλήψεων αποκαλείται περίοδος burn-in του αλγορίθμου και το δείγμα που έχει παραχθεί σε αυτήν την περίοδο απορρίπτεται εφόσον δεν προέρχεται από τη ζητούμενη κατανομή.

ii. Οι παρατηρήσεις που παράγονται από έναν αλγόριθμο Markov Chain Monte Carlo δεν είναι ανεξάρτητες. Αντιθέτως, έχουν μία μορφή εξάρτησης που καθορίζεται από τις ιδιότητες της Μαρκοβιανής διαδικασίας που παράγει ο αλγόριθμος. Η έλλειψη ανεξαρτησίας των παρατηρήσεων δεν επηρεάζει την προσέγγιση μέσων τιμών μέσω της μεθόδου Monte Carlo. Σε περιπτώσεις όπου απαιτείται η χρήση ανεξάρτητων παρατηρήσεων μπορούμε να πραγματοποιήσουμε thinning του δείγματος που παράγεται από τον αλγόριθμο. Αν υπολογίσουμε ότι μέχρι T^* διαδοχικές παρατηρήσεις που παράγονται από τον αλγόριθμο επιδεικνύουν στατιστικά σημαντική αυτοσυσχέτιση, τότε δεχόμαστε μόνο μία παρατήρηση κάθε T^* από αυτές που παράγει ο αλγόριθμος και απορρίπτουμε όλες τις άλλες.

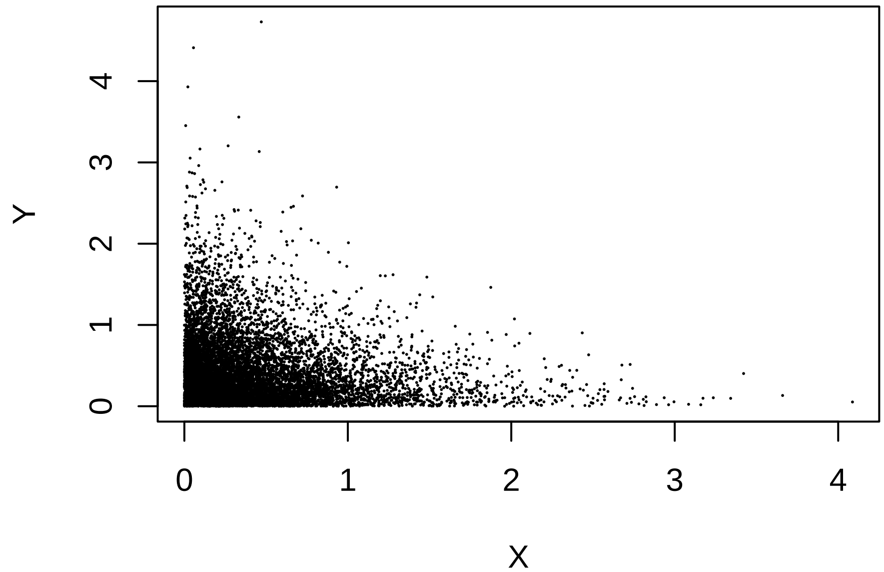
Παράδειγμα 5.1. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1), \dots, (X_n, Y_n)$ με συνάρτηση πυκνότητας πιθανότητας $f(x, y) \propto e^{-x-y-axy}$ για $x, y > 0$ και $a > 0$. Παρατηρούμε ότι:

$$f_{X|Y}(x \mid y) \propto e^{-x-axy} = e^{-(y+a)x}, \quad f_{Y|X}(y \mid x) \propto e^{-y-axy} = e^{-(x+a)y}.$$

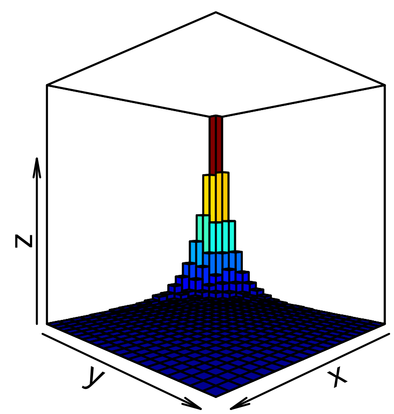
Δηλαδή, $(X \mid Y = y) \sim \mathrm{Exp}(y + a)$ και $(Y \mid X = x) \sim \mathrm{Exp}(x + a)$.

```
library(plot3D)
b = 10000
n = 10000
a = 2
X = numeric(b + n)
Y = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    X[i] = -log(U)/(Y[i - 1] + a)
    V = runif(1)
    Y[i] = -log(V)/(X[i] + a)
```

```
}
X = X[-(1:b)]
Y = Y[-(1:b)]
plot(X, Y, pch = 16, cex = 0.2)
```




```
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 135,
    border = 1)
```



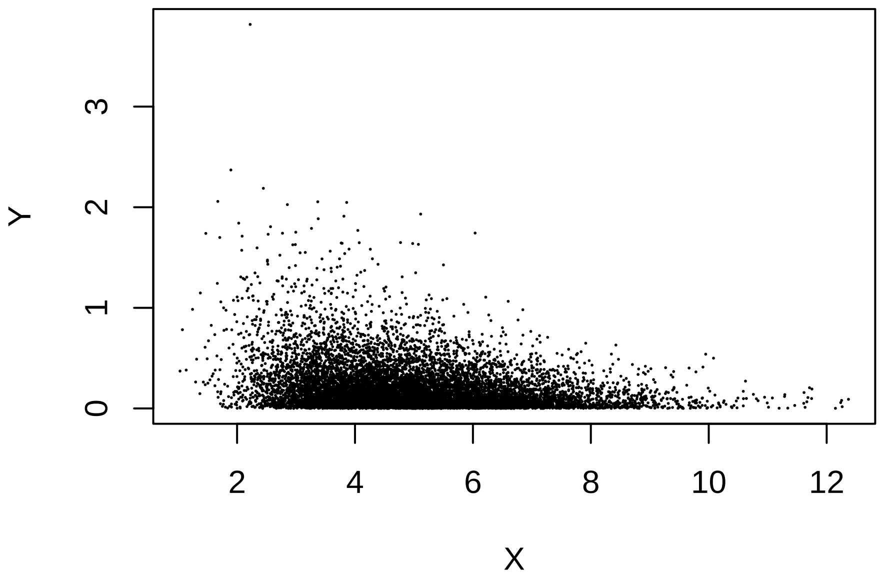
Παράδειγμα 5.2. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1), \ldots, (X_n, Y_n)$ με συνάρτηση πυκνότητας πιθανότητας $f(x, y) \propto x^k e^{-(\lambda+y)x}$ για $x, y, k, \lambda > 0$. Παρατηρούμε ότι:

$$f_{X|Y}(x \mid y) \propto x^k e^{-(\lambda+y)x}, \quad f_{Y|X}(y \mid x) \propto e^{-xy}.$$

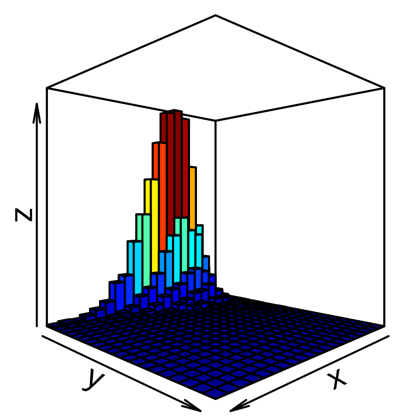
Δηλαδή, $(X \mid Y = y) \sim \mathrm{Gamma}(k+1, \lambda+y)$ και $(Y \mid X = x) \sim \mathrm{Exp}(x)$.

```
library(plot3D)
b = 10000
n = 10000
k = 10
lambda = 2
```

```
X = numeric(b + n)
Y = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(k + 1)
    R = -log(U)/(lambda + Y[i - 1])
    X[i] = sum(R)
    V = runif(1)
    Y[i] = -log(V)/X[i]
}
X = X[-(1:b)]
Y = Y[-(1:b)]
plot(X, Y, pch = 16, cex = 0.2)
```

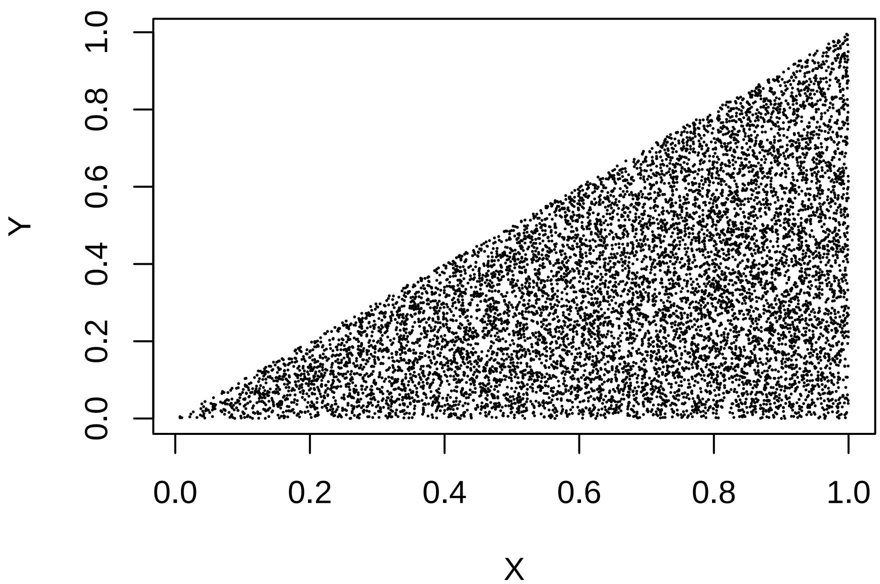



```
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 135,
    border = 1)
```

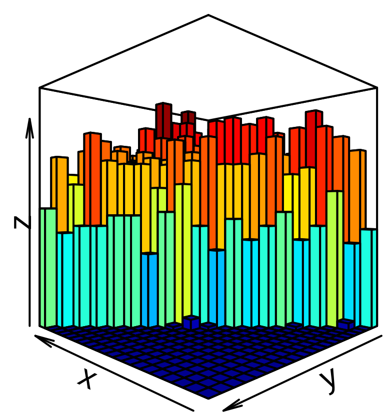


Παράδειγμα 5.3. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1), \dots, (X_n, Y_n)$ με συνάρτηση πυκνότητας πιθανότητας $f(x, y) \propto 1$ για $0 \leqslant y \leqslant x \leqslant 1$. Παρατηρούμε ότι $(X \mid Y = y) \sim \text{Unif}[y, 1]$ για $y \in [0, 1]$ και $(Y \mid X = x) \sim \text{Unif}[0, x]$ για $x \in [0, 1]$.

```r
library(plot3D)
b = 10000
n = 10000
X = numeric(b + n)
Y = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    X[i] = (1 - Y[i - 1]) * U + Y[i - 1]
    V = runif(1)
    Y[i] = X[i] * V
}
X = X[-(1:b)]
Y = Y[-(1:b)]
plot(X, Y, pch = 16, cex = 0.2)
```




```r
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 225,
    border = 1)
```

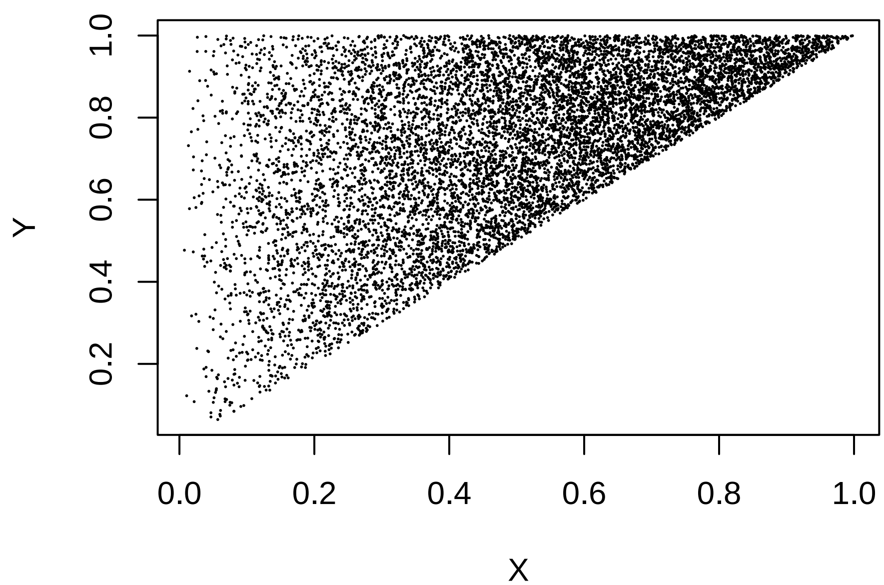


Παράδειγμα 5.4. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1), \ldots, (X_n, Y_n)$ με συνάρτηση πυκνότητας πιθα-

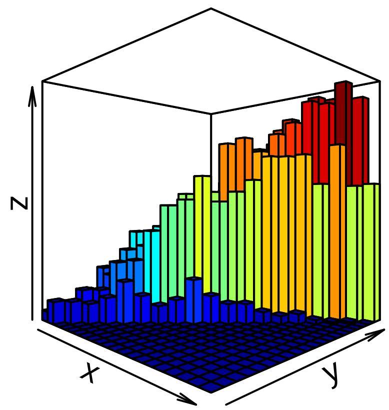
νότητας $f(x,y) \propto x$ για $0 \leqslant x \leqslant y \leqslant 1$. Για $x \in [0,1]$, παρατηρούμε ότι $f_{Y|X}(y \mid x) \propto 1$, δηλαδή $(Y \mid X = x) \sim \text{Unif}[x,1]$. Για $y \in [0,1]$, παρατηρούμε ότι $f_{X|Y}(x \mid y) \propto x$. Για $x \in [0,y]$, υπολογίζουμε ότι:

$$f_{X|Y}(x \mid y) = \frac{2x}{y^2}, \quad F_{X|Y}(x \mid y) = \frac{x^2}{y^2}, \quad F^{-1}_{X|Y}(u \mid y) = y\sqrt{u}.$$

```r
library(plot3D)
b = 10000
n = 10000
X = numeric(b + n)
Y = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    X[i] = Y[i - 1] * sqrt(U)
    V = runif(1)
    Y[i] = (1 - X[i]) * V + X[i]
}
X = X[-(1:b)]
Y = Y[-(1:b)]
plot(X, Y, pch = 16, cex = 0.2)
```




```r
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 45,
    border = 1)
```



Παράδειγμα 5.5. Για $\rho \in (-1, 1)$, θέλουμε να παράγουμε δείγμα:

$$(X_1, Y_1), \ldots, (X_n, Y_n) \sim \mathcal{N}_2 \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right).$$

Για $x, y \in \mathbb{R}$, υπολογίζουμε ότι:

$$f_{X,Y}(x, y) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{ -\frac{x^2 - 2\rho xy + y^2}{2(1-\rho^2)} \right\},$$

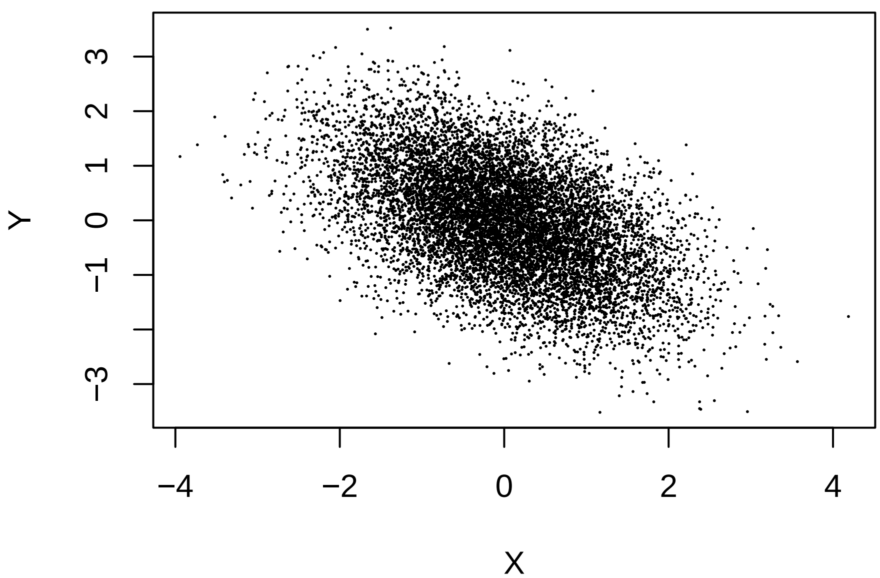
$$f_{X|Y}(x \mid y) \propto \exp\left\{ -\frac{x^2 - 2\rho yx}{2(1-\rho^2)} \right\} \propto \exp\left\{ -\frac{(x-\rho y)^2}{2(1-\rho^2)} \right\},$$

$$f_{Y|X}(y \mid x) \propto \exp\left\{ -\frac{y^2 - 2\rho xy}{2(1-\rho^2)} \right\} \propto \exp\left\{ -\frac{(y-\rho x)^2}{2(1-\rho^2)} \right\}.$$

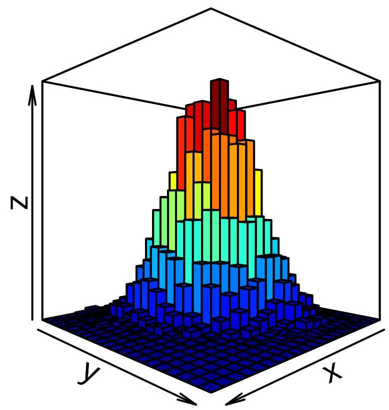
Δηλαδή, $(X \mid Y = y) \sim \mathcal{N}(\rho y, 1-\rho^2)$ και $(Y \mid X = x) \sim \mathcal{N}(\rho x, 1-\rho^2)$.

```
library(plot3D)
b = 10000
n = 10000
rho = -0.5
lambda = 1/sqrt(1 - rho^2)
M = sqrt(2 * exp(1)/pi)
X = numeric(b + n)
Y = numeric(b + n)
for (i in 2:(b + n)) {
    W = runif(1)
    X[i] = ifelse(W <= 0.5, rho * Y[i - 1] + log(2 * W)/lambda, rho * Y[i -
        1] - log(2 * (1 - W))/lambda)
    U = runif(1)
    V = M * dexp(abs(X[i] - rho * Y[i - 1]), lambda)/2 * U
    while (dnorm(X[i], rho * Y[i - 1], sqrt(1 - rho^2)) < V) {
        W = runif(1)
        X[i] = ifelse(W <= 0.5, rho * Y[i - 1] + log(2 * W)/lambda, rho * Y[i -
```

```
            1] - log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(X[i] - rho * Y[i - 1]), lambda)/2 * U
    }
    W = runif(1)
    Y[i] = ifelse(W <= 0.5, rho * X[i] + log(2 * W)/lambda, rho * X[i] - log(2 *
        (1 - W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y[i] - rho * X[i]), lambda)/2 * U
    while (dnorm(Y[i], rho * X[i], sqrt(1 - rho^2)) < V) {
        W = runif(1)
        Y[i] = ifelse(W <= 0.5, rho * X[i] + log(2 * W)/lambda, rho * X[i] -
            log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y[i] - rho * X[i]), lambda)/2 * U
    }
}
X = X[-(1:b)]
Y = Y[-(1:b)]
plot(X, Y, pch = 16, cex = 0.2)
```




```
hist3D(z = table(cut(X, 20), cut(Y, 20)), colkey = FALSE, phi = 0, theta = 135,
    border = 1)
```



Παράδειγμα 5.6. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1, Z_1), \ldots, (X_n, Y_n, Z_n)$ με συνάρτηση πυκνότητας πιθανότητας $f(x, y, z) \propto \binom{z}{x} y^{x+a-1}(1-y)^{zx+b-1}\frac{\lambda^z}{z!}$ για $x \in \{0, 1, \ldots, z\}$, $y \in [0, 1]$ και $z \in \mathbb{N}$. Παρατηρούμε ότι:

$$f_{X|Y,Z}(x \mid y, z) \propto \binom{z}{x} y^x (1-y)^{zx} = \binom{z}{x} [y(1-y)^z]^x \propto \binom{z}{x} \left[\frac{y(1-y)^z}{y(1-y)^z + 1}\right]^x \left[\frac{1}{y(1-y)^z + 1}\right]^{z-x},$$

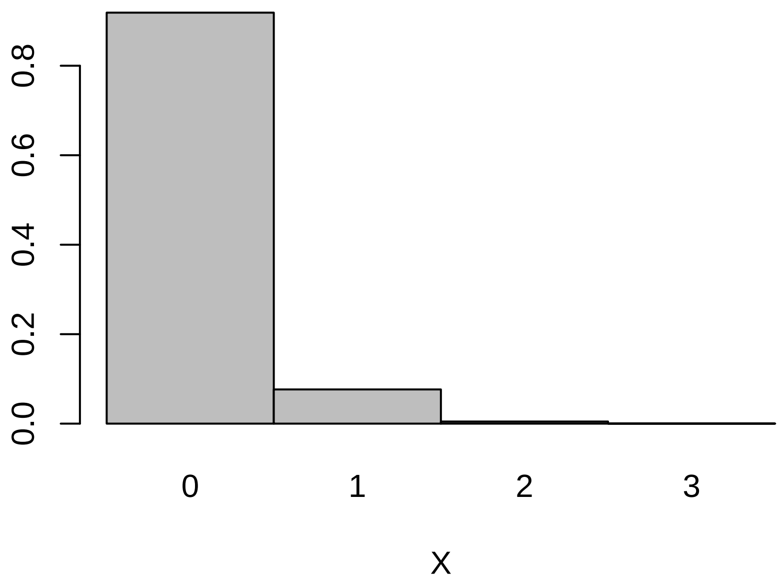
$$f_{Y|X,Z}(y \mid x, z) \propto y^{x+a-1}(1-y)^{zx+b-1},$$

$$f_{Z|X,Y}(z \mid x, y) \propto \binom{z}{x} (1-y)^{zx} \frac{\lambda^z}{z!} \propto \frac{[\lambda(1-y)^x]^z}{(z-x)!} \propto \frac{[\lambda(1-y)^x]^{z-x}}{(z-x)!}.$$

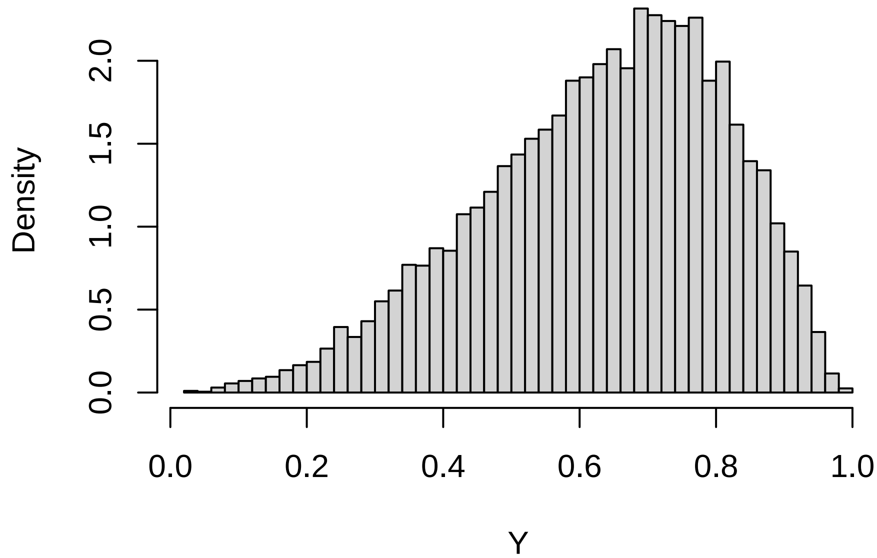
Δηλαδή, $(X \mid Y = y, Z = z) \sim \text{Binom}\left(z, \frac{y(1-y)^z}{y(1-y)^z+1}\right)$, $(Y \mid X = x, Z = z) \sim \text{Beta}(z + a, zx + b)$ και $(Z - X \mid X = x, Y = y) \sim \text{Poisson}\left(\lambda(1-y)^x\right)$.

```
b = 10000
n = 10000
a = 2
b = 3
lambda = 4
X = numeric(b + n)
Y = numeric(b + n)
Z = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(Z[i - 1])
    X[i] = sum(U < Y[i - 1] * (1 - Y[i - 1])^Z[i - 1]/(Y[i - 1] * (1 - Y[i -
        1])^Z[i - 1] + 1))
    M = dbeta((Z[i - 1] + a - 1)/(Z[i - 1] * (1 + X[i]) + a + b - 2), Z[i -
        1] + a, Z[i - 1] * X[i] + b)
    Y[i] = runif(1)
    U = runif(1)
    V = M * U
    while (dbeta(Y[i], Z[i - 1] + a, Z[i - 1] * X[i] + b) < V) {
        Y[i] = runif(1)
        U = runif(1)
```

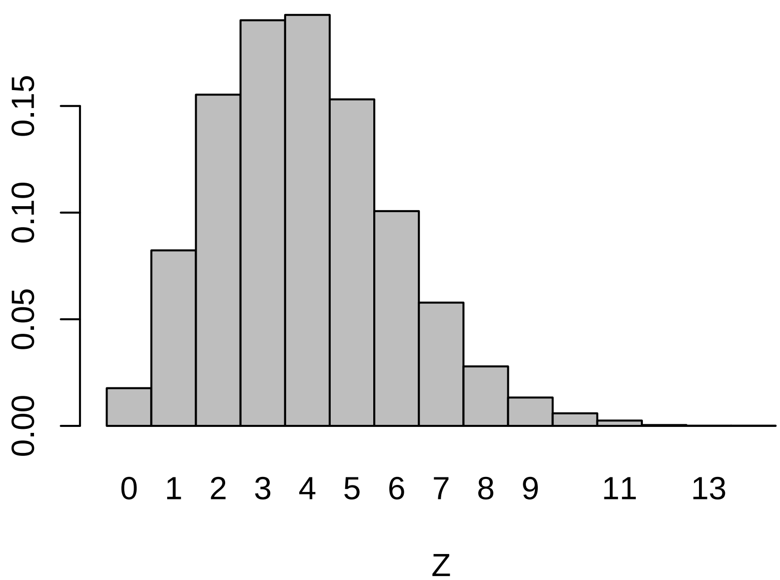
```
        V = M * U
    }
    Z[i] = X[i]
    U = runif(1)
    pmf = exp(-(lambda * (1 - Y[i])^X[i]))
    cdf = pmf
    while (U > cdf) {
        Z[i] = Z[i] + 1
        pmf = pmf * lambda * (1 - Y[i])^X[i]/(Z[i] - X[i])
        cdf = cdf + pmf
    }
}
X = X[-(1:b)]
Y = Y[-(1:b)]
Z = Z[-(1:b)]
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0, xlab = "X")
```




```
hist(Y, "FD", freq = FALSE, main = NA)
```




```
barplot(table(factor(Z, levels = 0:max(Z)))/n, space = 0, xlab = "Z")
```



Παράδειγμα 5.7. Έστω ανεξάρτητες τυχαίες μεταβλητές $X \sim \text{Poisson}(4)$ και $Y \sim \text{Poisson}(7)$. Θέλουμε να παράγουμε δείγμα $(X_1, Y_1), \dots, (X_n, Y_n)$ από τη δεσμευμένη κατανομή του (X, Y) δεδομένου ότι $X+Y \leqslant 10$. Για $x, y \in \mathbb{N}$ με $x + y \leqslant 10$, υπολογίζουμε ότι:

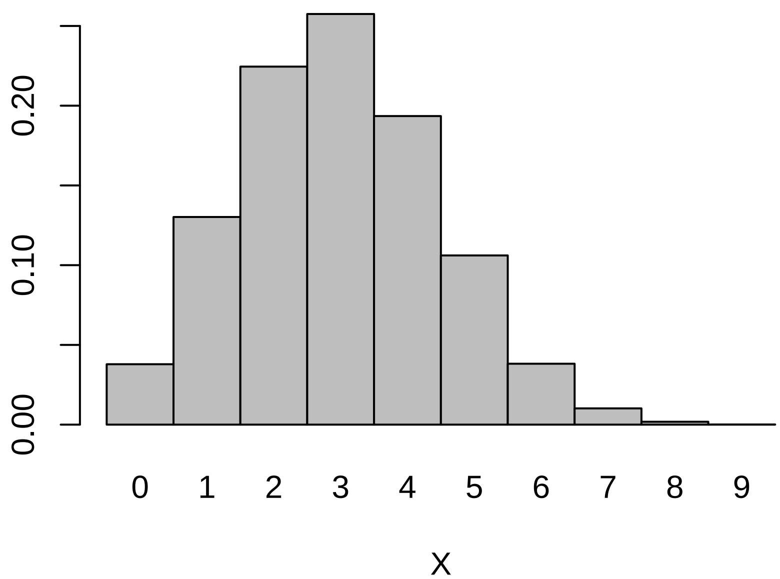
$$f_{X,Y|X+Y\leqslant 10}(x, y) = \frac{f_{X,Y}(x, y)}{\mathbb{P}(X + Y \leqslant 10)} \propto \frac{4^x 7^y}{x!y!},$$

$$f_{X|Y,X+Y\leqslant 10}(x \mid y) \propto \frac{4^x}{x!}, \quad f_{Y|X,X+Y\leqslant 10}(y \mid x) \propto \frac{7^y}{y!}.$$

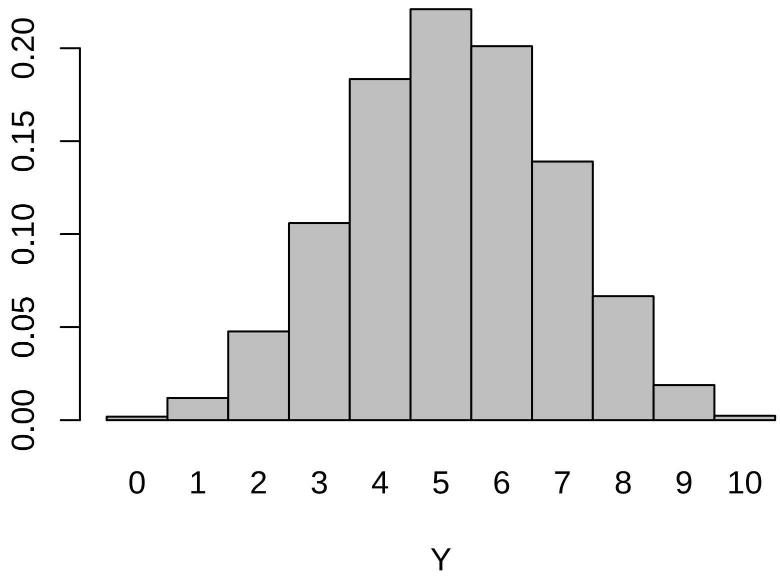
Δηλαδή, $(X \mid Y = y, X + Y \leqslant 10) \stackrel{d}{=} (X \mid X \leqslant 10 - y)$ και $(Y \mid X = x, X + Y \leqslant 10) \stackrel{d}{=} (Y \mid Y \leqslant 10 - x)$.

```
b = 10000
n = 10000
X = numeric(b + n)
```

```
Y = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    pmf = exp(-4)/ppois(10 - Y[i - 1], 4)
    cdf = pmf
    while (U > cdf) {
        X[i] = X[i] + 1
        pmf = pmf * 4/X[i]
        cdf = cdf + pmf
    }
    V = runif(1)
    pmf = exp(-7)/ppois(10 - X[i], 7)
    cdf = pmf
    while (V > cdf) {
        Y[i] = Y[i] + 1
        pmf = pmf * 7/Y[i]
        cdf = cdf + pmf
    }
}
X = X[-(1:b)]
Y = Y[-(1:b)]
barplot(table(factor(X, levels = 0:max(X)))/n, space = 0, xlab = "X")
```




```
barplot(table(factor(Y, levels = 0:max(Y)))/n, space = 0, xlab = "Y")
```



Παράδειγμα 5.8. Έστω ανεξάρτητες τυχαίες μεταβλητές $X_j \sim \text{Exp}(\lambda_j)$ για $j = 1, 2, \dots, k$. Ορίζουμε την τυχαία μεταβλητή $S = X_1 + \dots + X_k$. Θέλουμε να παράγουμε δείγμα $X^{(1)}, X^{(2)}, \dots, X^{(n)}$ από τη δεσμευμένη κατανομή του $(X_1, X_2, \dots, X_k)$ δεδομένου ότι $S > c$. Για $x_1, \dots, x_k > 0$ με $x_1 + \dots + x_k > c$, υπολογίζουμε ότι:

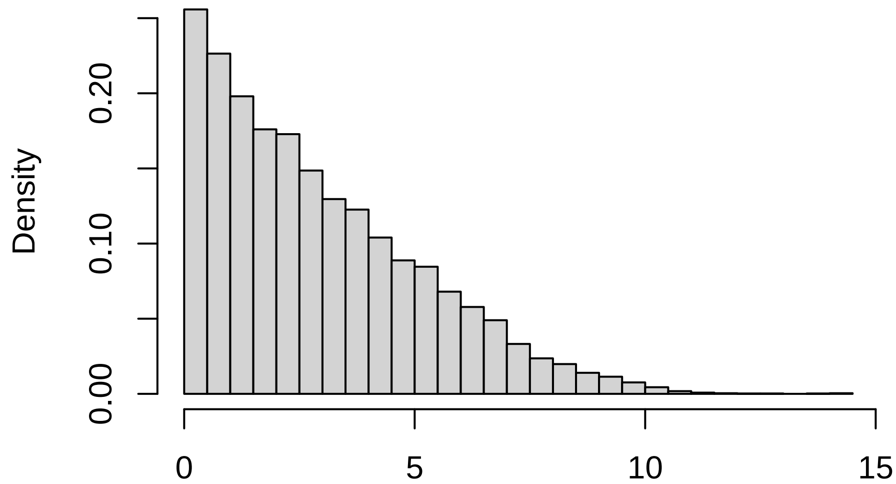
$$f_{X_1,\dots,X_k \mid S>c}(x_1, \dots, x_k) = \frac{f_{X_1,\dots,X_k}(x_1, \dots, x_k)}{\mathbb{P}(X_1 + \dots + X_k > c)} \propto \prod_{j=1}^{k} e^{-\lambda_j x_j} = \exp\left\{-\sum_{j=1}^{k} \lambda_j x_j\right\},$$

$$f_{X_j \mid X_{-j}, S>c}(x_j \mid x_{-j}) \propto e^{-\lambda_j x_j}.$$

Ορίζουμε $s_{-j} = x_1 + \dots + x_{j-1} + x_{j+1} + \dots + x_k$. Τότε, $\left(X_j \mid X_{-j} = x_{-j}, S > c\right) \stackrel{d}{=} \left(X_j \mid X_j > c - s_{-j}\right)$. Δηλαδή, παίρνουμε ότι $\left(X_j \mid X_{-j} = x_{-j}, S > c\right) \stackrel{d}{=} X_j + \max\left\{c - s_{-j}, 0\right\}$.

```
b = 10000
n = 10000
c = 10
k = 4
lambda = rep(1, k)
X = matrix(0, b + n, k)
for (i in 2:(b + n)) {
    for (j in 1:k) {
        if (j == 1) {
            s = sum(X[i - 1, -1])
        } else if (j == k) {
            s = sum(X[i, -k])
        } else {
            s = sum(c(X[i, 1:(j - 1)], X[i - 1, (j + 1):k]))
        }
        U = runif(1)
        X[i, j] = max(c - s, 0) - log(U)/lambda[j]
    }
```

```
}
X = X[-(1:b), ]
hist(X[, 1], "FD", freq = FALSE, main = NA, xlab = NA)
```



Παράδειγμα 5.9. Έστω τυχαία μεταβλητή $Y \sim \mathrm{Unif}[0.02, 0.1]$. Θεωρούμε τις δεσμευμένα ανεξάρτητες τυχαίες μεταβλητές $(W_1 \mid Y = y), (W_2 \mid Y = y) \sim \mathrm{Exp}(y)$. Δεδομένου ότι $W_1 = w_1$ και $W_2 = w_2$, ορίζουμε τις δεσμευμένα ανεξάρτητες διαδικασίες Poisson $\{N_1(t) : t \geqslant 0\}$ με ρυθμό w_1 και $\{N_2(t) : t \geqslant 0\}$ με ρυθμό w_2. Θέλουμε να εκτιμήσουμε τις δεσμευμένες μέσες τιμές:

$$\mathbb{E}\left[N_1(1) \mid N_1(0.5) = 25, N_2(0.5) = 18\right], \quad \mathbb{E}\left[N_2(1) \mid N_1(0.5) = 25, N_2(0.5) = 18\right].$$

Σύμφωνα με το θεώρημα διπλής μέσης τιμής, υπολογίζουμε ότι:

$$\begin{aligned}\mathbb{E}\left[N_1(1) \mid N_1(0.5) = 25, N_2(0.5) = 18\right] &= \mathbb{E}\left[\mathbb{E}\left(N_1(1) \mid W_1, N_1(0.5) = 25, N_2(0.5) = 18\right)\right] \\ &= \mathbb{E}\left[25 + 0.5W_1 \mid N_1(0.5) = 25, N_2(0.5) = 18\right],\end{aligned}$$

$$\begin{aligned}\mathbb{E}\left[N_2(1) \mid N_1(0.5) = 25, N_2(0.5)\right] &= \mathbb{E}\left[\mathbb{E}\left(N_2(1) \mid W_2, N_1(0.5) = 25, N_2(0.5) = 18\right)\right] \\ &= \mathbb{E}\left[18 + 0.5W_2 \mid N_1(0.5) = 25, N_2(0.5) = 18\right].\end{aligned}$$

Ορίζουμε τα ενδεχόμενα $A_1 = [N_1(0.5) = 25]$, $A_2 = [N_2(0.5) = 18]$. Για $y \in [0.02, 0.1]$ και $w_1, w_2 > 0$, υπολογίζουμε ότι:

$$\begin{aligned}f_{Y,W_1,W_2|A_1,A_2}(y, w_1, w_2) &= \frac{f_{Y,W_1,W_2}(y, w_1, w_2)\mathbb{P}\left[A_1, A_2 \mid Y = y, W_1 = w_1, W_2 = w_2\right]}{\mathbb{P}\left(A_1, A_2\right)} \\ &\propto f_Y(y) f_{W_1,W_2|Y}(w_1, w_2 \mid y)\mathbb{P}\left[A_1 \mid W_1 = w_1\right]\mathbb{P}\left[A_2 \mid W_2 = w_2\right] \\ &\propto f_{W_1|Y}(w_1 \mid y) f_{W_2|Y}(w_2 \mid y) e^{-0.5w_1} w_1^{25} e^{-0.5w_2} w_2^{18} \\ &= y e^{-yw_1} y e^{-yw_2} w_1^{25} w_2^{18} e^{-0.5(w_1+w_2)} = y^2 w_1^{25} w_2^{18} e^{-(y+0.5)(w_1+w_2)},\end{aligned}$$

$$\begin{aligned}&f_{Y|W_1,W_2,A_1,A_2}(y \mid w_1, w_2) \propto y^2 e^{-(w_1+w_2)y}, \\ &f_{W_1|Y,W_2,A_1,A_2}(w_1 \mid y, w_2) \propto w_1^{25} e^{-(y+0.5)w_1}, \\ &f_{W_2|Y,W_1,A_1,A_2}(w_2 \mid y, w_1) \propto w_2^{18} e^{-(y+0.5)w_2}.\end{aligned}$$

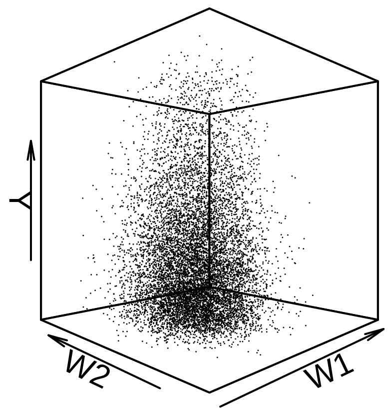
Αν $(X \mid W_1 = w_1, W_2 = w_2) \sim \text{Gamma}(3, w_1 + w_2)$, τότε:

$$[Y \mid W_1 = w_2, W_2 = w_2, A_1, A_2] \stackrel{d}{=} (X \mid W_1 = w_1, W_2 = w_2, 0.02 \leqslant X \leqslant 0.1)\,,$$

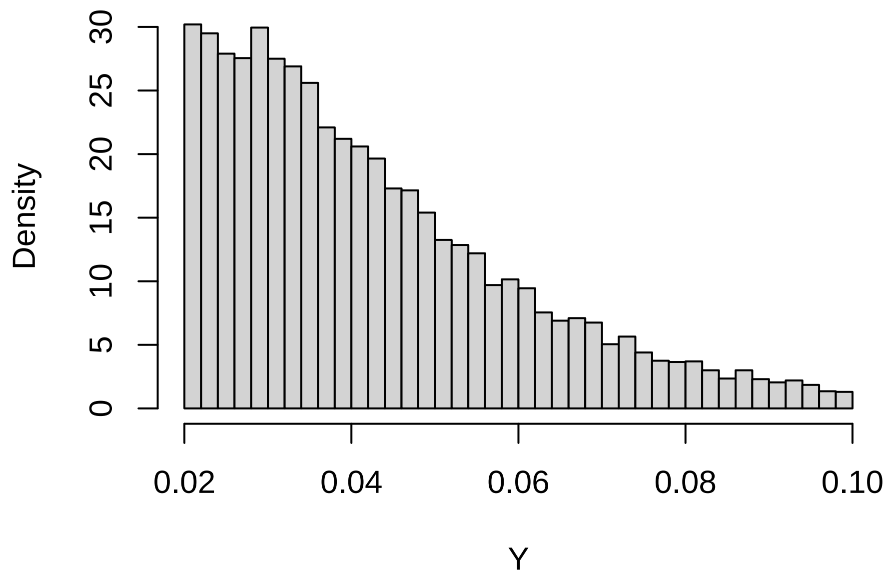
$$[W_1 \mid Y = y, W_2 = w_2, A_1, A_2] \sim \text{Gamma}(26, y + 0.5),$$

$$[W_2 \mid Y = y, W_1 = w_1, A_1, A_2] \sim \text{Gamma}(19, y + 0.5).$$

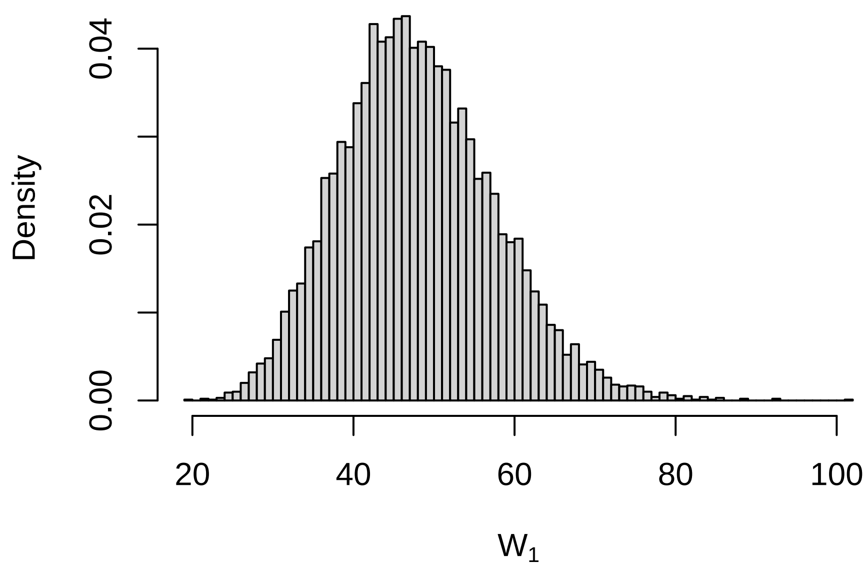
```
library(plot3D)
b = 10000
n = 10000
Y = numeric(b + n)
W1 = numeric(b + n)
W2 = numeric(b + n)
W1[1] = 1
W2[1] = 1
for (i in 2:(b + n)) {
    U = runif(3)
    R = -log(U)/(W1[i - 1] + W2[i - 1])
    Y[i] = sum(R)
    while (Y[i] < 0.02 || Y[i] > 0.1) {
        U = runif(3)
        R = -log(U)/(W1[i - 1] + W2[i - 1])
        Y[i] = sum(R)
    }
    U = runif(26)
    R = -log(U)/(Y[i] + 0.5)
    W1[i] = sum(R)
    U = runif(19)
    R = -log(U)/(Y[i] + 0.5)
    W2[i] = sum(R)
}
Y = Y[-(1:b)]
W1 = W1[-(1:b)]
W2 = W2[-(1:b)]
scatter3D(W1, W2, Y, colvar = NA, phi = 0, theta = 315, xlab = "W1", ylab = "W2",
    zlab = "Y", pch = 16, cex = 0.1)
```



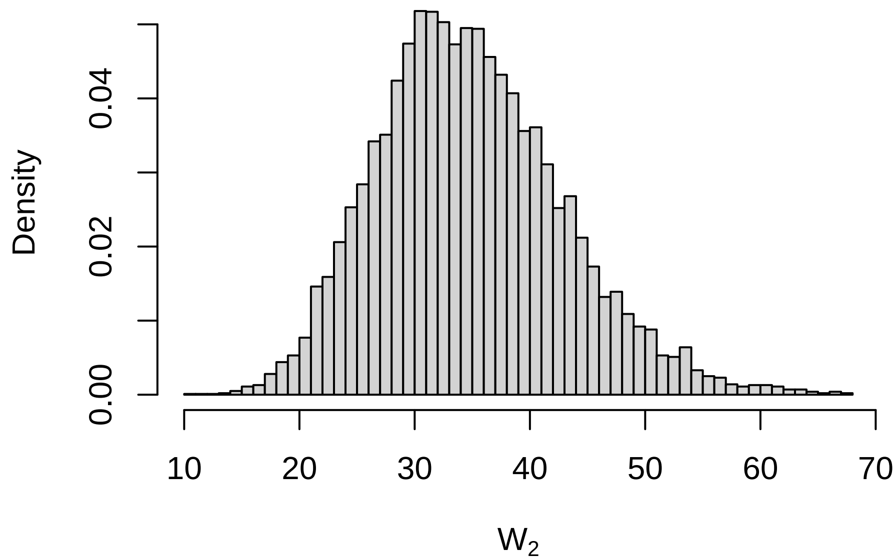

```
hist(Y, "FD", freq = FALSE, main = NA)
```




```
hist(W1, "FD", freq = FALSE, main = NA, xlab = expression(W[1]))
```



```r
hist(W2, "FD", freq = FALSE, main = NA, xlab = expression(W[2]))
```




```r
mean(25 + 0.5 * W1)
```

```
## [1] 48.99764
```

```r
mean(18 + 0.5 * W2)
```

```
## [1] 35.43046
```

Slice Sampling

Θέλουμε να παράγουμε δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας f με στήριγμα S και $M = \max_{x \in S} f(x)$. Υποθέτουμε ότι η εφαρμογή της μεθόδου αντίστροφου μετασχηματισμού δεν είναι εφικτή. Θεωρούμε τις τυχαίες μεταβλητές X, Y με από κοινού συνάρτηση πυκνότητας πιθανότητας $f_{X,Y}(x,y) = \mathbb{1}_{\{y \leqslant f(x)\}}$ για $x \in S$ και $y \in [0, M]$. Τότε, παρατηρούμε ότι:

$$f_X(x) = \int_0^M f_{X,Y}(x,y)dy = \int_0^{f(x)} 1dy = f(x), \quad f_{X|Y}(x \mid y) \propto \mathbb{1}_{\{x \in f^{-1}[y,\infty)\}},$$

όπου $f^{-1}[y, \infty) = \{x \in S : f(x) \geqslant y\}$. Δηλαδή, $X \sim f$, $(Y \mid X = x) \sim \text{Unif}\,[0, f(x)]$ και η δεσμευμένη κατανομή της X δεδομένου ότι $Y = y$ είναι η ομοιόμορφη στο σύνολο $f^{-1}[y, \infty)$.

Algorithm 5.2 Slice Sampling

Είσοδος: Συνάρτηση πυκνότητας πιθανότητας f, ζητούμενο burn-in b και μέγεθος δείγματος n.

1: Θεωρούμε μία αρχική τιμή X_1.

2: Για $i = 2, 3, \ldots, b + n$, επαναλαμβάνουμε τα παρακάτω βήματα:

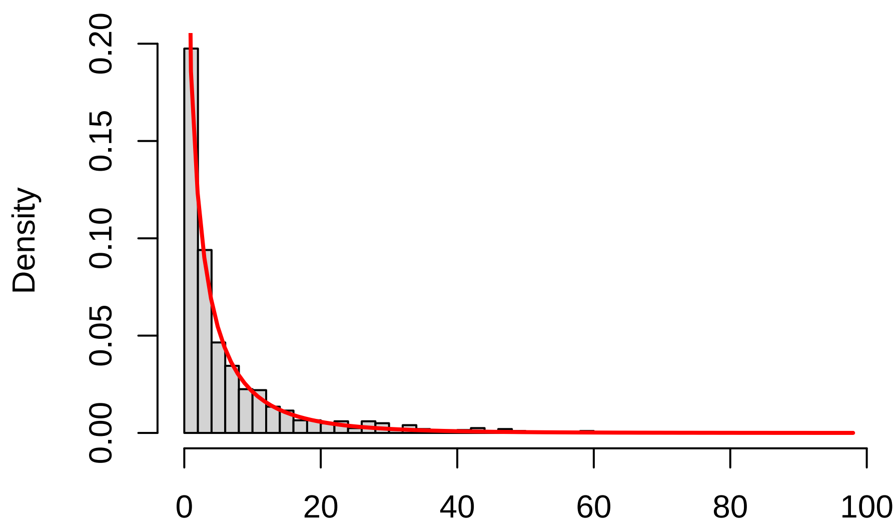
i: Προσομοιώνουμε $U \sim \text{Unif}[0, 1]$ και θέτουμε $Y = f(X_{i-1})\, U \sim \text{Unif}\,[0, f(X_{i-1})]$.

ii: Προσομοιώνουμε τη X_i από την ομοιόμορφη κατανομή στο σύνολο $f^{-1}[Y, \infty)$.

Έξοδος: Τυχαίο δείγμα $X_{b+1}, X_{b+2}, \ldots, X_{b+n}$ από την f.

Παράδειγμα 5.10. Θέλουμε να παράγουμε δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας $f(x) = e^{-\sqrt{x}}/2$ για $x > 0$. Για $y \in [0, 0.5]$, παρατηρούμε ότι $f(x) \geqslant y \Leftrightarrow x \leqslant \log^2(2y)$. Δηλαδή, $(X \mid Y = y) \sim \text{Unif}\left[0, \log^2(2y)\right]$.

```
b = 1000
n = 1000
X = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    Y = exp(-sqrt(X[i - 1]))/2 * U
    V = runif(1)
    X[i] = log(2 * Y)^2 * V
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(exp(-sqrt(x))/2, add = TRUE, col = "red", lwd = 2)
```



Παράδειγμα 5.11. Θέλουμε να παράγουμε δείγμα $X_1, \ldots, X_n \sim \mathcal{N}(\mu, \sigma^2)$. Για $y \in \left[0, \frac{1}{\sqrt{2\pi}\sigma}\right]$, παρατηρούμε ότι:

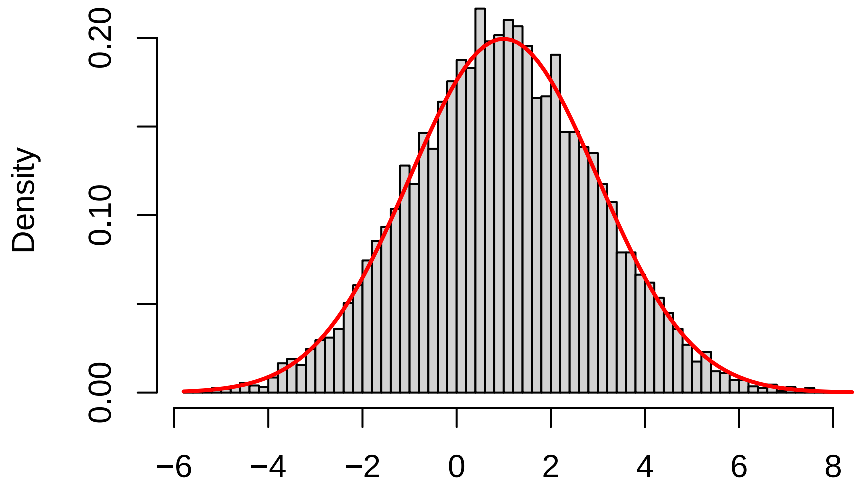
$$f(x) \geqslant y \quad \Leftrightarrow \quad \mu - \sigma\sqrt{\log\frac{1}{2\pi\sigma^2 y^2}} \leqslant x \leqslant \mu + \sigma\sqrt{\log\frac{1}{2\pi\sigma^2 y^2}}.$$

Με άλλα λόγια,

$$(X \mid Y = y) \sim \text{Unif}\left[\mu - \sigma\sqrt{\log\frac{1}{2\pi\sigma^2 y^2}}, \mu + \sigma\sqrt{\log\frac{1}{2\pi\sigma^2 y^2}}\right].$$

```
b = 10000
n = 10000
mu = 1
sigma = 2
X = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
```

```
    Y = exp(-(X[i - 1] - mu)^2/(2 * sigma^2))/(sqrt(2 * pi) * sigma) * U
    V = runif(1)
    X[i] = 2 * sigma * sqrt(log(1/(2 * pi * sigma^2 * Y^2))) * V + mu - sigma *
        sqrt(log(1/(2 * pi * sigma^2 * Y^2)))
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(dnorm(x, mu, sigma), add = TRUE, col = "red", lwd = 2)
```



Αλγόριθμος Metropolis-Hastings

Θέλουμε να παράγουμε δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας f. Υποθέτουμε ότι η εφαρμογή της μεθόδου αντίστροφου μετασχηματισμού ή της μεθόδου slice sampling δεν είναι εφικτή.

Algorithm 5.3 Metropolis-Hastings

Είσοδος: Συνάρτηση πυκνότητας πιθανότητας f, προτείνουσα δεσμευμένη συνάρτηση πυκνότητας πιθανότητας g, ζητούμενο burn-in b και μέγεθος δείγματος n.

1: Θεωρούμε μία αρχική τιμή X_1.

2: Για $i = 2, 3, \ldots, b + n$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε την Y από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $g\left(y \mid X_{i-1}\right)$.

ii: Προσομοιώνουμε $U \sim \mathrm{Unif}[0, 1]$ και υπολογίζουμε την πιθανότητα αποδοχής:

$$A\left(X_{i-1}, Y\right) = \min\left\{\frac{f(Y) g\left(X_{i-1} \mid Y\right)}{f\left(X_{i-1}\right) g\left(Y \mid X_{i-1}\right)}, 1\right\}.$$

iii: Αν $U < A\left(X_{i-1}, Y\right)$, τότε θέτουμε $X_i = Y$. Διαφορετικά, θέτουμε $X_i = X_{i-1}$.

Έξοδος: Τυχαίο δείγμα $X_{b+1}, X_{b+2}, \ldots, X_{b+n}$ από την f.

Η ακολουθία τυχαίων μεταβλητών $\{X_i\}$ αποτελεί μία Μαρκοβιανή διαδικασία διακριτού χρόνου με πυρήνα μεταβάσεων $K\left(x_i \mid x_{i-1}\right) = g\left(x_i \mid x_{i-1}\right) A\left(x_{i-1}, x_i\right)$.

Θεώρημα 5.2. Η Μαρκοβιανή διαδικασία $\{X_i\}$ με πυρήνα μεταβάσεων $K(x_i \mid x_{i-1})$ είναι αντιστρέψιμη και έχει ως μοναδική στάσιμη κατανομή την f.

Απόδειξη. Αν $\frac{f(x_i)}{f(x_{i-1})} < \frac{g(x_i|x_{i-1})}{g(x_{i-1}|x_i)}$, τότε ισχύει ότι:

$$A(x_{i-1}, x_i) = \frac{f(x_i) g(x_{i-1} \mid x_i)}{f(x_{i-1}) g(x_i \mid x_{i-1})}, \quad A(x_i, x_{i-1}) = 1.$$

Αν $\frac{f(x_i)}{f(x_{i-1})} > \frac{g(x_i|x_{i-1})}{g(x_{i-1}|x_i)}$, τότε ισχύει:

$$A(x_{i-1}, x_i) = 1, \quad A(x_i, x_{i-1}) = \frac{f(x_{i-1}) g(x_i \mid x_{i-1})}{f(x_i) g(x_{i-1} \mid x_i)}.$$

Επομένως, παίρνουμε ότι:

$$f(x_{i-1}) g(x_i \mid x_{i-1}) \min\left\{\frac{f(x_i) g(x_{i-1} \mid x_i)}{f(x_{i-1}) g(x_i \mid x_{i-1})}, 1\right\} = f(x_i) g(x_{i-1} \mid x_i) \min\left\{\frac{f(x_{i-1}) g(x_i \mid x_{i-1})}{f(x_i) g(x_{i-1} \mid x_i)}, 1\right\},$$

$$\begin{aligned} f(x_{i-1}) K(x_i \mid x_{i-1}) &= f(x_{i-1}) g(x_i \mid x_{i-1}) A(x_{i-1}, x_i) \\ &= f(x_i) g(x_{i-1} \mid x_i) A(x_i, x_{i-1}) = f(x_i) K(x_{i-1} \mid x_i). \end{aligned}$$

Εφόσον η συνάρτηση f ικανοποιεί τις εξισώσεις λεπτομερούς ισορροπίας για τη Μαρκοβιανή διαδικασία $\{X_i\}$, συμπεραίνουμε ότι η $\{X_i\}$ είναι αντιστρέψιμη με μοναδική στάσιμη κατανομή την f. □

Παράδειγμα 5.12. Θέλουμε να παράγουμε δείγμα $X_1, \dots, X_n \sim \mathrm{Bin}(k, p)$. Θεωρούμε την προτείνουσα δεσμευμένη συνάρτηση πιθανότητας:

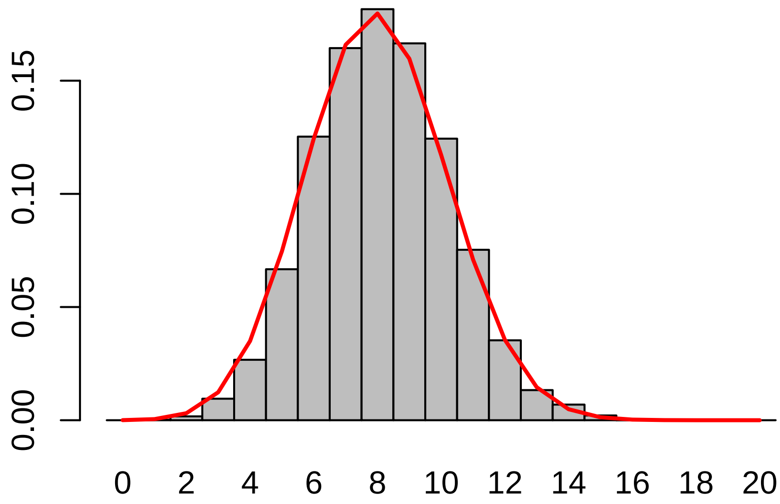
$$g(y \mid x) = \begin{cases} 0.5, & x \in \{1, 2, \dots, k-1\}, \quad y = x+1 \\ 0.5, & x \in \{1, 2, \dots, k-1\}, \quad y = x-1 \\ 1, & x = 0, \quad y = 1 \\ 1, & x = k, \quad y = k-1 \end{cases},$$

δηλαδή τον συμμετρικό τυχαίο περίπατο στον χώρο καταστάσεων $S = \{0, 1, \dots, k\}$ με ανακλαστικά φράγματα στις καταστάσεις $\{0\}$ και $\{k\}$. Τότε, υπολογίζουμε ότι:

$$\frac{f(y) g(x \mid y)}{f(x) g(y \mid x)} = \begin{cases} \frac{k-x}{x+1} \frac{p}{1-p}, & x \in \{1, 2, \dots, k-2\}, \quad y = x+1 \\ \frac{x}{k-x+1} \frac{1-p}{p}, & x \in \{2, 3, \dots, k-1\}, \quad y = x-1 \\ \frac{k}{2} \frac{p}{1-p}, & x = 0, \quad y = 1 \\ \frac{2}{k} \frac{1-p}{p}, & x = 1, \quad y = 0 \\ \frac{2}{k} \frac{p}{1-p}, & x = k-1, \quad y = k \\ \frac{k}{2} \frac{1-p}{p}, & x = k, \quad y = k-1 \end{cases}.$$

```
b = 10000
n = 10000
k = 20
```

```r
p = 0.4
X = numeric(b + n)
for (i in 2:(b + n)) {
    if (X[i - 1] == 0) {
        Y = 1
        A = k/2 * p/(1 - p)
    } else if (X[i - 1] == k) {
        Y = k - 1
        A = k/2 * (1 - p)/p
    } else {
        V = runif(1)
        if (V < 0.5) {
            Y = X[i - 1] + 1
            if (Y == k) {
                A = 2/k * p/(1 - p)
            } else {
                A = (k - X[i - 1])/(X[i - 1] + 1) * p/(1 - p)
            }
        } else {
            Y = X[i - 1] - 1
            if (Y == 0) {
                A = 2/k * (1 - p)/p
            } else {
                A = X[i - 1]/(k - X[i - 1] + 1) * (1 - p)/p
            }
        }
    }
    U = runif(1)
    X[i] = ifelse(U < A, Y, X[i - 1])
}
X = X[-(1:b)]
barplot(table(factor(X, levels = 0:k))/n, space = 0)
lines(0:k + 0.5, dbinom(0:k, k, p), col = "red", lwd = 2)
```



Παράδειγμα 5.13. Έστω τυχαία μεταβλητή $X \sim \text{Poisson}(\lambda)$. Θέλουμε να παράγουμε δείγμα $X_1, X_2, \dots, X_n$ από τη δεσμευμένη κατανομή της X δεδομένου ότι $X \leqslant k$. Για $x \in \{0, 1, \dots, k\}$, παρατηρούμε ότι $f_{X|X\leqslant k}(x) \propto \frac{\lambda^x}{x!}$. Θεωρούμε την προτείνουσα δεσμευμένη συνάρτηση πιθανότητας:

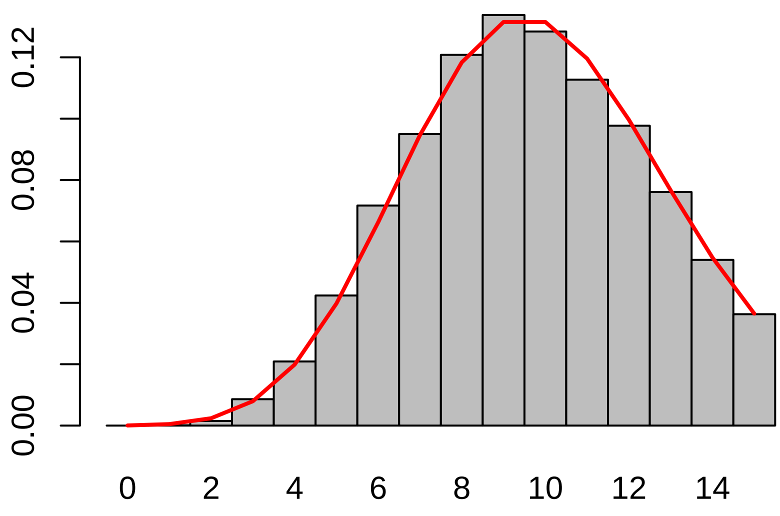
$$g(y \mid x) = \begin{cases} 0.5, & x \in \{0, 1, \dots, k-1\}, \quad y = x + 1 \\ 0.5, & x \in \{1, 2, \dots, k\}, \quad y = x - 1 \\ 0.5, & x = 0, \quad y = 0 \\ 0.5, & x = k, \quad y = k \end{cases},$$

δηλαδή τον συμμετρικό τυχαίο περίπατο στον χώρο καταστάσεων $S = \{0, 1, \dots, k\}$ με ελαστικά φράγματα στις καταστάσεις $\{0\}$ και $\{k\}$. Τότε, υπολογίζουμε ότι:

$$\frac{f(y)g(x \mid y)}{f(x)g(y \mid x)} = \begin{cases} \frac{\lambda}{x+1}, & x \in \{0, 1, \dots, k-1\}, \quad y = x + 1 \\ \frac{x}{\lambda}, & x \in \{1, 2, \dots, k\}, \quad y = x - 1 \\ 1, & x = y = 0 \\ 1, & x = y = k \end{cases}.$$

```
b = 10000
n = 10000
lambda = 10
k = 15
X = numeric(b + n)
for (i in 2:(b + n)) {
    V = runif(1)
    if (X[i - 1] == 0) {
        if (V < 0.5) {
            Y = 0
            A = 1
        } else {
            Y = 1
```

```
            A = lambda
        }
    } else if (X[i - 1] == k) {
        if (V < 0.5) {
            Y = k
            A = 1
        } else {
            Y = k - 1
            A = k/lambda
        }
    } else {
        if (V < 0.5) {
            Y = X[i - 1] + 1
            A = lambda/(X[i - 1] + 1)
        } else {
            Y = X[i - 1] - 1
            A = X[i - 1]/lambda
        }
    }
    U = runif(1)
    X[i] = ifelse(U < A, Y, X[i - 1])
}
X = X[-(1:b)]
barplot(table(factor(X, levels = 0:k))/n, space = 0)
lines(0:k + 0.5, dpois(0:k, lambda)/ppois(k, lambda), col = "red", lwd = 2)
```



Παράδειγμα 5.14. Θέλουμε να παράγουμε δείγμα $X_1, X_2, \ldots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\} + \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x+\mu)^2\right\}.$$

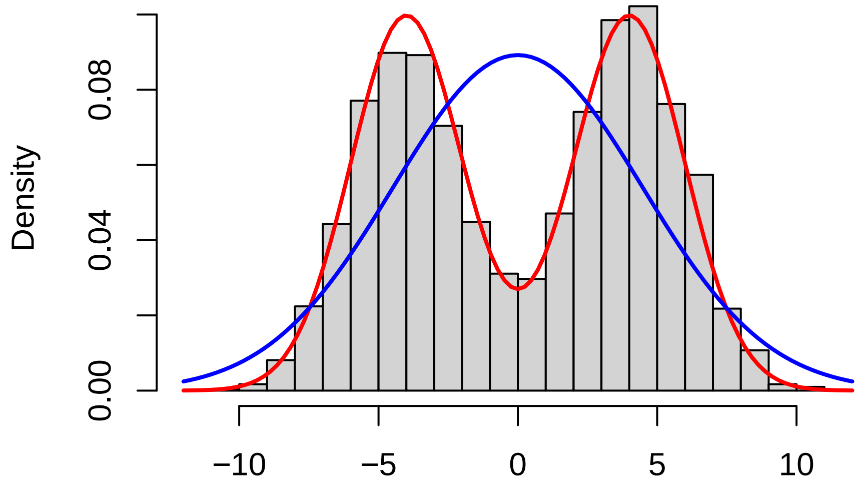
Θεωρούμε την προτείνουσα τυχαία μεταβλητή $Y \sim \mathcal{N}\left(0, \sigma^2 + \mu^2\right)$ με συνάρτηση πυκνότητας πιθανότητας:

$$g(y) = \frac{1}{\sqrt{2\pi\left(\sigma^2 + \mu^2\right)}} \exp\left\{-\frac{1}{2\left(\sigma^2 + \mu^2\right)} y^2\right\}.$$

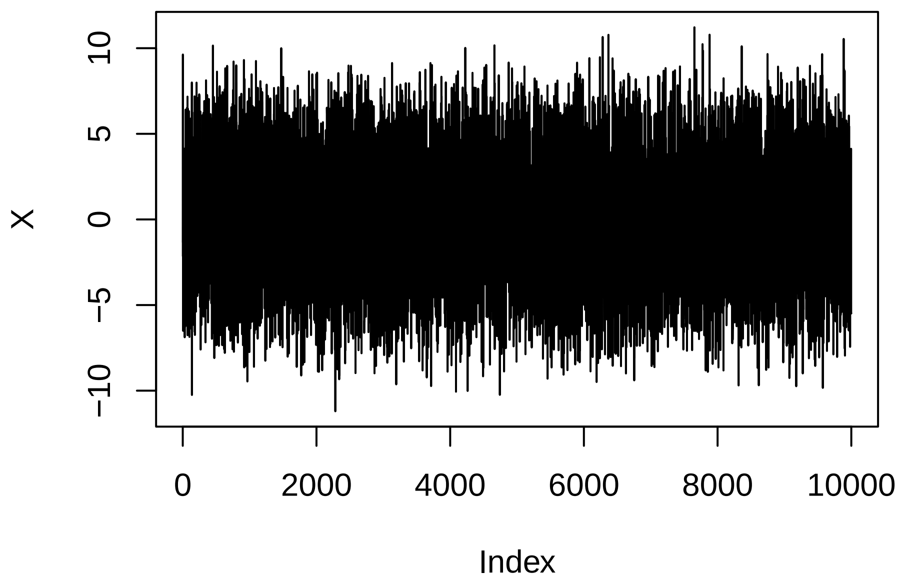
Παρατηρούμε ότι η προτείνουσα συνάρτηση πυκνότητας πιθανότητας δεν εξαρτάται από την τρέχουσα κατάσταση της Μαρκοβιανής διαδικασίας $\{X_i\}$.

```
b = 10000
n = 10000
mu = 4
sigma = 2
lambda = 1/sqrt(sigma^2 + mu^2)
M = sqrt(2 * exp(1)/pi)
accept = 0
X = numeric(b + n)
for (i in 2:(b + n)) {
    W = runif(1)
    Y = ifelse(W <= 0.5, log(2 * W)/lambda, -log(2 * (1 - W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y), lambda)/2 * U
    while (dnorm(Y, 0, sqrt(sigma^2 + mu^2)) < V) {
        W = runif(1)
        Y = ifelse(W <= 0.5, log(2 * W)/lambda, -log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y), lambda)/2 * U
    }
    A = (0.5 * dnorm(Y, mu, sigma) + 0.5 * dnorm(Y, -mu, sigma)) * dnorm(X[i -
        1], 0, sqrt(sigma^2 + mu^2))/((0.5 * dnorm(X[i - 1], mu, sigma) + 0.5 *
        dnorm(X[i - 1], -mu, sigma)) * dnorm(Y, 0, sqrt(sigma^2 + mu^2)))
    U = runif(1)
    if (U < A) {
        X[i] = Y
        if (i > b) {
            accept = accept + 1
        }
    } else {
        X[i] = X[i - 1]
    }
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(0.5 * dnorm(x, mu, sigma) + 0.5 * dnorm(x, -mu, sigma), add = TRUE, col = "red",
    lwd = 2)
```

```r
curve(dnorm(x, 0, sqrt(sigma^2 + mu^2)), add = TRUE, col = "blue", lwd = 2)
```




```r
plot(X, type = "l")
```




```r
print(accept/n)
```

```
## [1] 0.6781
```

Για τη σωστή εφαρμογή του αλγορίθμου Metropolis-Hastings με προτείνουσα συνάρτηση πυκνότητας πιθανότητας ανεξάρτητη της τρέχουσας κατάστασης της Μαρκοβιανής διαδικασίας $\{X_i\}$, πρέπει να επιλέξουμε μία κατάλληλη προτείνουσα πυκνότητα ώστε το ποσοστό αποδοχής των προτεινόμενων καταστάσεων του αλγορίθμου να είναι κοντά στο 100%.

Αν για την προτείνουσα δεσμευμένη συνάρτηση πυκνότητας πιθανότητας ισχύει ότι $g\left(y \mid x_{i-1}\right) = g\left(x_{i-1} \mid y\right)$, τότε ο αλγόριθμος καλείται **Random Walk Metropolis-Hastings**. Παρατηρούμε ότι:

$$A\left(x_{i-1}, y\right) = \min\left\{\frac{f(y)}{f\left(x_{i-1}\right)}, 1\right\}.$$

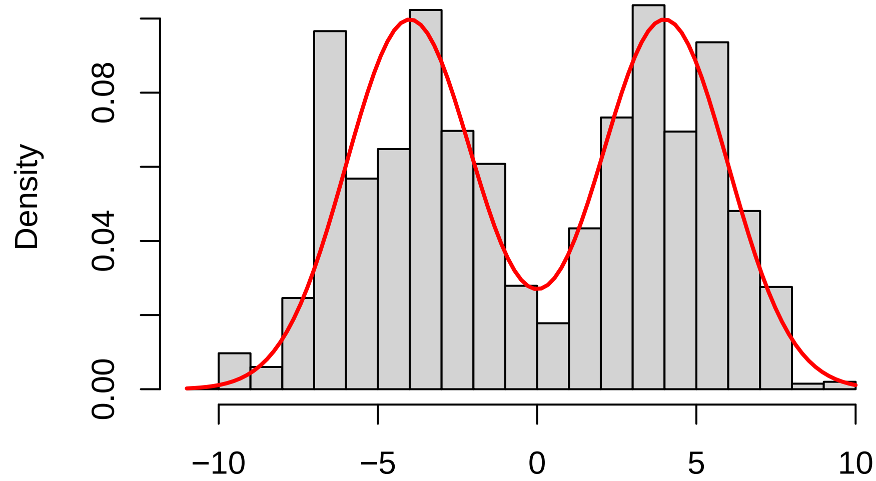
Αν $f(y) > f\left(x_{i-1}\right)$, τότε παίρνουμε ότι $A\left(x_{i-1}, y\right) = 1$. Με άλλα λόγια, η Μαρκοβιανή διαδικασία μεταβαίνει με πιθανότητα 1 σε μία προτείνουσα κατάσταση με μεγαλύτερη πυκνότητα από την τρέχουσα κατάσταση. Αν η προτείνουσα κατάσταση έχει μικρότερη πυκνότητα από την τρέχουσα κατάσταση, τότε η Μαρκοβιανή διαδικασία μεταβαίνει σε αυτή με πιθανότητα $A\left(x_{i-1}, y\right) \in (0,1)$.

Θεωρούμε την προτείνουσα τυχαία μεταβλητή $\left(Y \mid X_{i-1} = x_{i-1}\right) \sim \mathcal{N}\left(x_{i-1}, \sigma_p^2\right)$ με δεσμευμένη συνάρτηση πυκνότητας πιθανότητας:

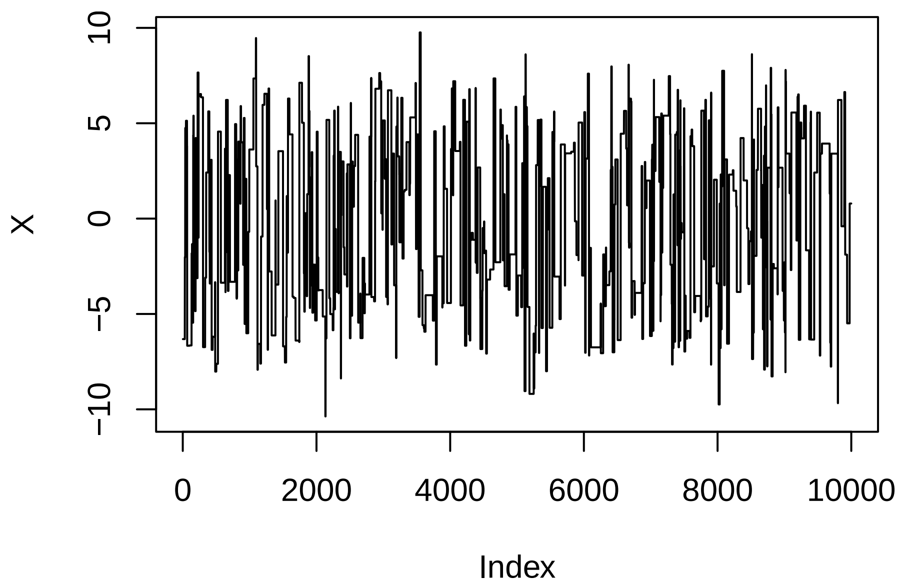
$$g\left(y \mid x_{i-1}\right) = \frac{1}{\sqrt{2\pi}\sigma_p} \exp\left\{-\frac{1}{2\sigma_p^2}\left(y - x_{i-1}\right)^2\right\}.$$

```
b = 10000
n = 10000
mu = 4
sigma = 2
sigmap = 100
lambda = 1/sigmap
M = sqrt(2 * exp(1)/pi)
accept = 0
X = numeric(b + n)
for (i in 2:(b + n)) {
    W = runif(1)
    Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 * (1 -
        W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    while (dnorm(Y, X[i - 1], sigmap) < V) {
        W = runif(1)
        Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 *
            (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    }
    A = (0.5 * dnorm(Y, mu, sigma) + 0.5 * dnorm(Y, -mu, sigma))/(0.5 * dnorm(X[i -
        1], mu, sigma) + 0.5 * dnorm(X[i - 1], -mu, sigma))
    U = runif(1)
    if (U < A) {
        X[i] = Y
        if (i > b) {
            accept = accept + 1
        }
    } else {
        X[i] = X[i - 1]
    }
}
X = X[-(1:b)]
```

```r
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(0.5 * dnorm(x, mu, sigma) + 0.5 * dnorm(x, -mu, sigma), add = TRUE, col = "red",
    lwd = 2)
```




```r
plot(X, type = "l")
```



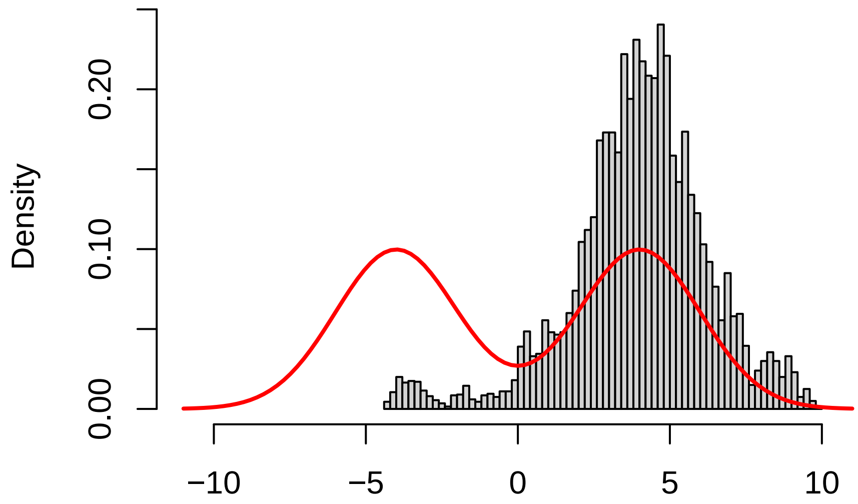

```r
print(accept/n)
```

```
## [1] 0.0502
```

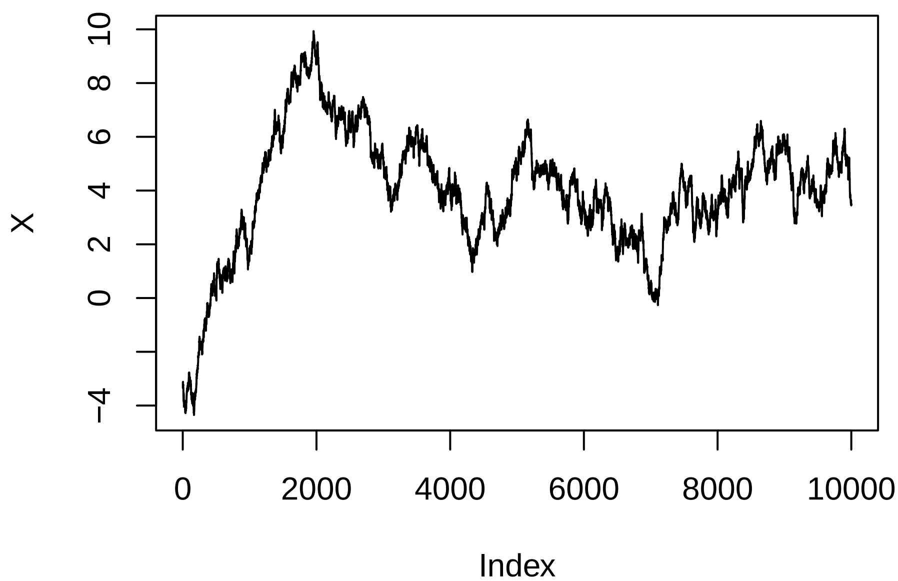
Όταν η διασπορά σ_p^2 της προτείνουσας συνάρτησης πυκνότητας πιθανότητας είναι πολύ μεγάλη, τότε παρατηρούμε ότι ο αλγόριθμος προτείνει καταστάσεις πολύ μακριά από την τρέχουσα κατάσταση της Μαρκοβιανής διαδικασίας. Αυτές οι καταστάσεις έχουν πολύ μικρή πυκνότητα, οπότε η Μαρκοβιανή διαδικασία μεταβαίνει σε αυτές με πολύ μικρή πιθανότητα. Συνεπώς, η Μαρκοβιανή διαδικασία παγιδεύεται στην ίδια κατάσταση για μεγάλες χρονικές περιόδους και δεν εξερευνάει καλά ολόκληρο το στήριγμα της κατανομής f.

```r
b = 10000
n = 10000
mu = 4
```

```r
sigma = 2
sigmap = 0.1
lambda = 1/sigmap
M = sqrt(2 * exp(1)/pi)
accept = 0
X = numeric(b + n)
for (i in 2:(b + n)) {
    W = runif(1)
    Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 * (1 -
        W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    while (dnorm(Y, X[i - 1], sigmap) < V) {
        W = runif(1)
        Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 *
            (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    }
    A = (0.5 * dnorm(Y, mu, sigma) + 0.5 * dnorm(Y, -mu, sigma))/(0.5 * dnorm(X[i -
        1], mu, sigma) + 0.5 * dnorm(X[i - 1], -mu, sigma))
    U = runif(1)
    if (U < A) {
        X[i] = Y
        if (i > b) {
            accept = accept + 1
        }
    } else {
        X[i] = X[i - 1]
    }
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlim = c(-11, 11), xlab = NA)
curve(0.5 * dnorm(x, mu, sigma) + 0.5 * dnorm(x, -mu, sigma), add = TRUE, col = "red",
    lwd = 2)
```




```r
plot(X, type = "l")
```



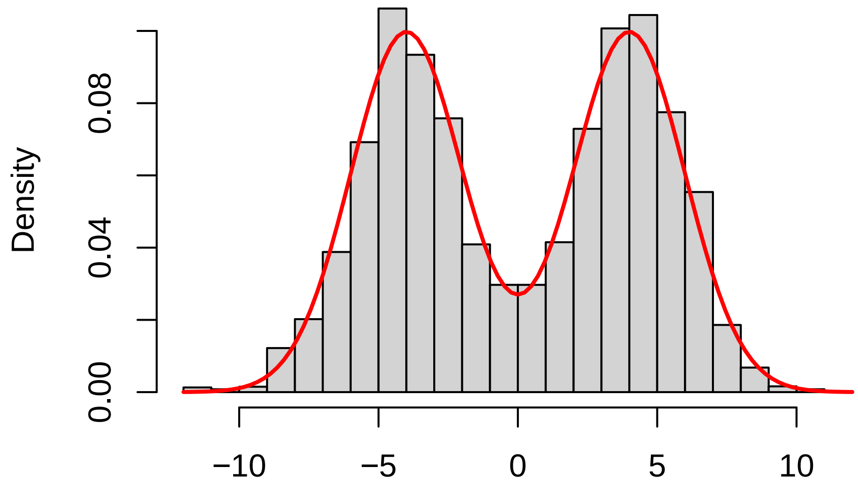

```r
print(accept/n)
```

```
## [1] 0.9874
```

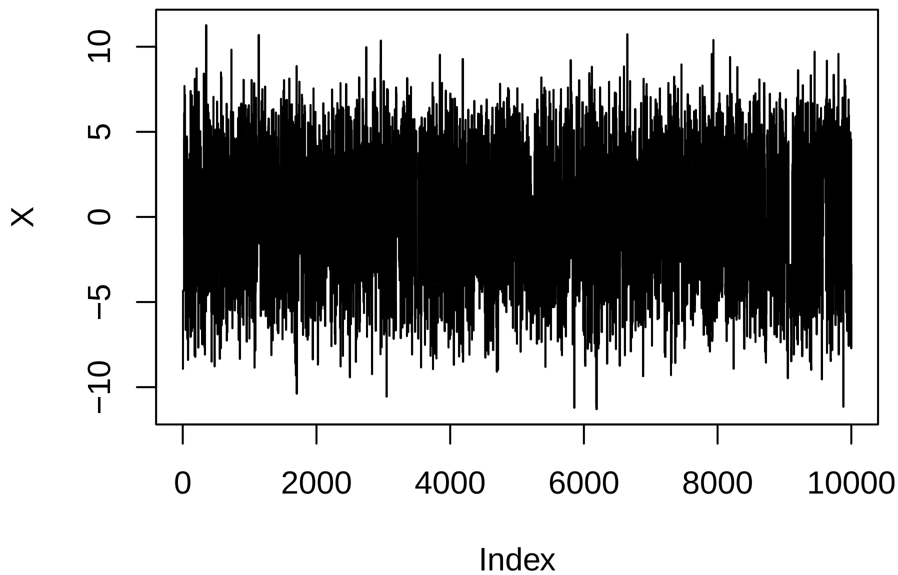
Όταν η διασπορά σ_p^2 της προτείνουσας συνάρτησης πυκνότητας πιθανότητας είναι πολύ μικρή, τότε παρατηρούμε ότι ο αλγόριθμος προτείνει καταστάσεις πολύ κοντά στην τρέχουσα κατάσταση της Μαρκοβιανής διαδικασίας. Εφόσον η Μαρκοβιανή διαδικασία μεταβαίνει με πιθανότητα 1 σε προτεινόμενες καταστάσεις με μεγαλύτερη πυκνότητα από την τρέχουσα κατάσταση, τείνει να μεταβαίνει προς την κοντινότερη κορυφή της κατανομής f. Συνεπώς, η Μαρκοβιανή διαδικασία παγιδεύεται γύρω από μία κορυφή της f και δεν εξερευνάει ολόκληρο το στήριγμα της f.

```r
b = 10000
n = 10000
mu = 4
sigma = 2
sigmap = 10
lambda = 1/sigmap
```

```
M = sqrt(2 * exp(1)/pi)
accept = 0
X = numeric(b + n)
for (i in 2:(b + n)) {
    W = runif(1)
    Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 * (1 -
        W))/lambda)
    U = runif(1)
    V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    while (dnorm(Y, X[i - 1], sigmap) < V) {
        W = runif(1)
        Y = ifelse(W <= 0.5, X[i - 1] + log(2 * W)/lambda, X[i - 1] - log(2 *
            (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(Y - X[i - 1]), lambda)/2 * U
    }
    A = (0.5 * dnorm(Y, mu, sigma) + 0.5 * dnorm(Y, -mu, sigma))/(0.5 * dnorm(X[i -
        1], mu, sigma) + 0.5 * dnorm(X[i - 1], -mu, sigma))
    U = runif(1)
    if (U < A) {
        X[i] = Y
        if (i > b) {
            accept = accept + 1
        }
    } else {
        X[i] = X[i - 1]
    }
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(0.5 * dnorm(x, mu, sigma) + 0.5 * dnorm(x, -mu, sigma), add = TRUE, col = "red",
    lwd = 2)
```




```r
plot(X, type = "l")
```




```r
print(accept/n)
```

```
## [1] 0.4041
```

Για τη σωστή εφαρμογή του αλγορίθμου Random Walk Metropolis-Hastings πρέπει να επιλέξουμε μία κατάλληλη τιμή για την διασπορά σ_p^2 της προτείνουσας συνάρτησης πυκνότητας πιθανότητας ώστε το ποσοστό αποδοχής των προτεινόμενων καταστάσεων του αλγορίθμου να είναι κοντά στο 50%.

Αύξηση Δεδομένων

Θέλουμε να παράγουμε δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας f. Υποθέτουμε ότι η εφαρμογή της μεθόδου αντίστροφου μετασχηματισμού δεν είναι εφικτή. Θεωρούμε δύο τυχαίες μεταβλητές $X \sim f$ και Y. Αρχικά, υποθέτουμε ότι είναι εύκολο να προσομοιώσουμε από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $f_{X|Y}$. Τότε, υπολογίζουμε ότι:

$$f_{Y|X}(y \mid x) = \frac{f_Y(y) f_{X|Y}(x \mid y)}{f(x)} \propto f_Y(y) f_{X|Y}(x \mid y).$$

Εναλλακτικά, υποθέτουμε ότι είναι εύκολο να προσομοιώσουμε από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $f_{Y|X}$. Τότε, υπολογίζουμε ότι:

$$f_{X|Y}(x \mid y) = \frac{f_X(x) f_{Y|X}(y \mid x)}{f(y)} \propto f_X(x) f_{Y|X}(y \mid x).$$

Σε οποιαδήποτε από αυτές τις δύο περιπτώσεις, μπορούμε στη συνέχεια να εφαρμόσουμε έναν δειγματολήπτη Gibbs για να προσομοιώσουμε εναλλάξ από τις δεσμευμένες κατανομές του X δεδομένου του Y και του Y δεδομένου του X.

Algorithm 5.4 Αύξηση Δεδομένων

Είσοδος: Δεσμευμένες συναρτήσεις πυκνότητας πιθανότητας $f_{X|Y}$, $f_{Y|X}$, ζητούμενο burn-in b και μέγεθος δείγματος n.

1: Θεωρούμε μία αρχική τιμή X_1.

2: Για $i = 2, 3, \dots, b + n$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε την Y από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $f_{Y|X}(y \mid X_{i-1})$.

ii: Προσομοιώνουμε τη X_i από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $f_{X|Y}(x \mid Y)$.

Έξοδος: Τυχαίο δείγμα $X_{b+1}, X_{b+2}, \dots, X_{b+n}$ από την f.

Παράδειγμα 5.15. Θέλουμε να παράγουμε δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας:

$$f(x) = \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\} + \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x+\mu)^2\right\}.$$

Θεωρούμε δύο τυχαίες μεταβλητές $X \sim f$ και $Y \sim \text{Bernoulli}(0.5)$. Υποθέτουμε ότι $(X \mid Y = 0) \sim \mathcal{N}(\mu, \sigma^2)$ και $(X \mid Y = 1) \sim \mathcal{N}(-\mu, \sigma^2)$. Τότε, υπολογίζουμε ότι:

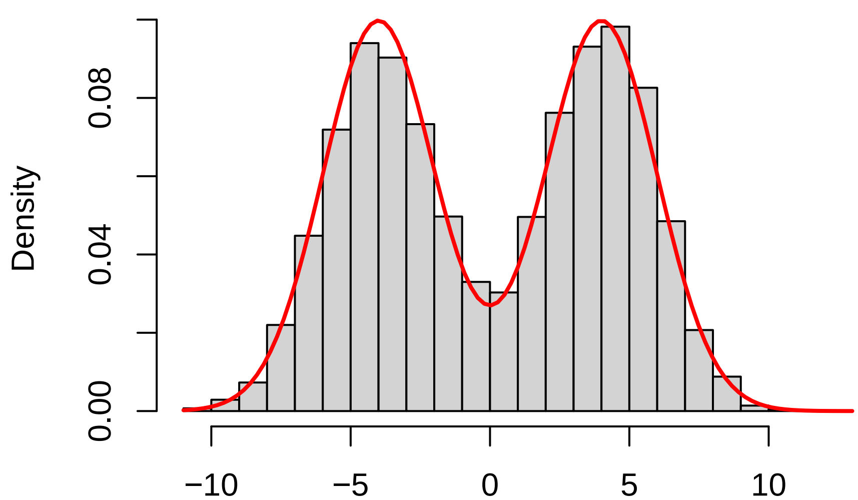
$$\mathbb{P}(Y = 0 \mid X = x) \propto \mathbb{P}(Y = 0) f_{X|Y}(x \mid 0) \propto \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\},$$

$$\mathbb{P}(Y = 1 \mid X = x) \propto \mathbb{P}(Y = 1) f_{X|Y}(x \mid 1) \propto \exp\left\{-\frac{1}{2\sigma^2}(x+\mu)^2\right\},$$

$$(Y \mid X = x) \sim \text{Bernoulli}\left(\frac{\exp\left\{-\frac{1}{2\sigma^2}(x+\mu)^2\right\}}{\exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\} + \exp\left\{-\frac{1}{2\sigma^2}(x+\mu)^2\right\}}\right) \equiv \text{Bernoulli}\left(\frac{1}{e^{2\mu x/\sigma^2} + 1}\right).$$

```
b = 10000
n = 10000
mu = 4
sigma = 2
lambda = 1/sigma
M = sqrt(2 * exp(1)/pi)
X = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(1)
    Y = U < 1/(exp(2 * mu * X[i - 1]/sigma^2) + 1)
```

```r
    if (Y == 0) {
        W = runif(1)
        X[i] = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 - W))/lambda)
        U = runif(1)
        V = M * dexp(abs(X[i] - mu), lambda)/2 * U
        while (dnorm(X[i], mu, sigma) < V) {
            W = runif(1)
            X[i] = ifelse(W <= 0.5, mu + log(2 * W)/lambda, mu - log(2 * (1 -
                W))/lambda)
            U = runif(1)
            V = M * dexp(abs(X[i] - mu), lambda)/2 * U
        }
    } else {
        W = runif(1)
        X[i] = ifelse(W <= 0.5, -mu + log(2 * W)/lambda, -mu - log(2 * (1 -
            W))/lambda)
        U = runif(1)
        V = M * dexp(abs(X[i] + mu), lambda)/2 * U
        while (dnorm(X[i], -mu, sigma) < V) {
            W = runif(1)
            X[i] = ifelse(W <= 0.5, -mu + log(2 * W)/lambda, -mu - log(2 * (1 -
                W))/lambda)
            U = runif(1)
            V = M * dexp(abs(X[i] + mu), lambda)/2 * U
        }
    }
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
curve(0.5 * dnorm(x, mu, sigma) + 0.5 * dnorm(x, -mu, sigma), add = TRUE, col = "red",
    lwd = 2)
```

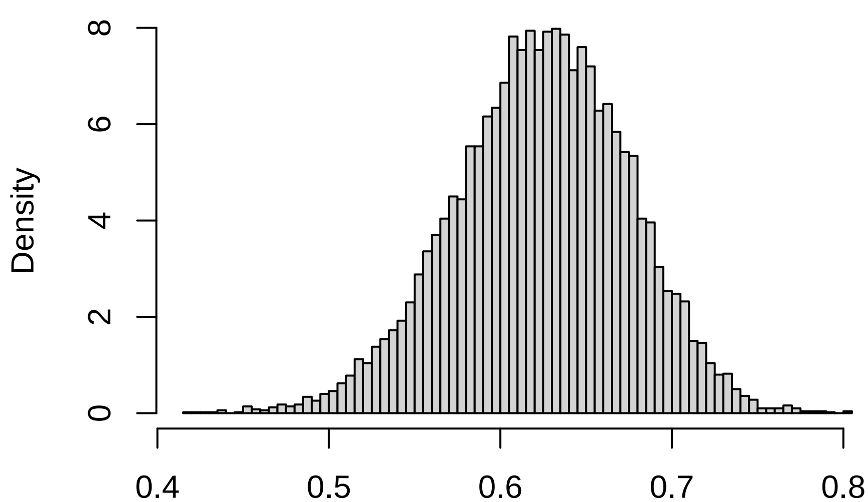


Παράδειγμα 5.16. Θέλουμε να παράγουμε δείγμα $X_1, X_2, \dots, X_n$ από τη συνάρτηση πυκνότητας πιθανότητας $f(x) \propto x^{34}(1-x)^{38}(2+x)^{125}$ για $x \in [0,1]$. Θεωρούμε δύο τυχαίες μεταβλητές $X \sim f$ και Y. Υποθέτουμε ότι $(Y \mid X = x) \sim \mathrm{Bin}\left(125, \frac{x}{2+x}\right)$, Για $y \in \{0, 1, \dots, 125\}$, υπολογίζουμε ότι:

$$
\begin{aligned}
f_{X|Y}(x \mid y) &\propto f(x) f_{Y|X}(y \mid x) \\
&\propto x^{34}(1-x)^{38}(2+x)^{125}\binom{125}{y}\left(\frac{x}{2+x}\right)^{y}\left(1-\frac{x}{2+x}\right)^{125-y} \propto x^{y+34}(1-x)^{38}.
\end{aligned}
$$

Δηλαδή, $(X \mid Y = y) \sim \mathrm{Beta}(y+35, 39)$.

```r
b = 10000
n = 10000
X = numeric(b + n)
for (i in 2:(b + n)) {
    U = runif(125)
    Y = sum(U < X[i - 1]/(2 + X[i - 1]))
    M = dbeta((Y + 34)/(Y + 72), Y + 35, 39)
    X[i] = runif(1)
    U = runif(1)
    V = M * U
    while (dbeta(X[i], Y + 35, 39) < V) {
        X[i] = runif(1)
        U = runif(1)
        V = M * U
    }
}
X = X[-(1:b)]
hist(X, "FD", freq = FALSE, main = NA, xlab = NA)
```



Simulated Annealing

Θέλουμε να μεγιστοποιήσουμε τη συνάρτηση $h : S \to \mathbb{R}$ με $\int_S e^{h(x)} dx < \infty$. Ορίζουμε $h^* = \max_{x \in S} h(x)$ και $M = \{x \in S : h(x) = h^*\}$. Για $i \in \mathbb{N}$, θεωρούμε τη συνάρτηση πυκνότητας πιθανότητας:

$$f_i(x) \propto e^{\lambda_i h(x)} \propto e^{\lambda_i [h(x) - h^*]}.$$

Παρατηρούμε ότι $h(x) - h^* < 0$ για $x \notin M$ και $h(x) - h^* = 0$ για $x \in M$. Αν $\lambda_i \to \infty$, τότε παίρνουμε ότι:

$$\lim_{i \to \infty} f_i(x) \propto \mathbb{1}_{\{x \in M\}}.$$

Αν $X_i \sim f_i$ για $i \in \mathbb{N}$, τότε η ακολουθία τυχαίων μεταβλητών $\{X_i\}$ συγκλίνει κατά κατανομή σε μία τυχαία μεταβλητή που ακολουθεί την ομοιόμορφη κατανομή στο σύνολο M. Για την προσομοίωση από τη συνάρτηση πυκνότητας πιθανότητας f_i χρησιμοποιούμε συνήθως τον αλγόριθμο Random Walk Metropolis-Hastings με $\lambda_i = \lambda_1 \log i$ ή $\lambda_i = \lambda_1 r^{i-1}$ για $\lambda_1 > 0$ και $r > 1$.

Algorithm 5.5 Simulated Annealing

Είσοδος: Συνάρτηση h, προτείνουσα δεσμευμένη συνάρτηση πυκνότητας πιθανότητας g, ακολουθία λ_i και ζητούμενο μέγεθος δείγματος n.

1: Θεωρούμε μία αρχική τιμή X_1.

2: Για $i = 2, 3, \dots, n$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε την Y από τη δεσμευμένη συνάρτηση πυκνότητας πιθανότητας $g\left(y \mid X_{i-1}\right)$.

ii: Προσομοιώνουμε $U \sim \text{Unif}[0, 1]$ και υπολογίζουμε την πιθανότητα αποδοχής:

$$A\left(X_{i-1}, Y\right) = \min\left\{e^{\lambda_i [h(Y) - h(X_{i-1})]}, 1\right\}.$$

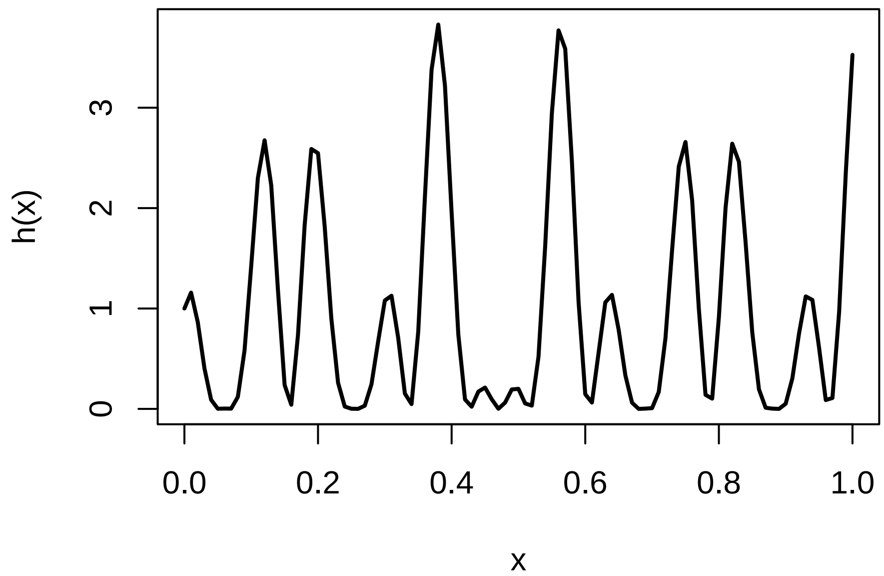
iii: Αν $U < A\left(X_{i-1}, Y\right)$, τότε θέτουμε $X_i = Y$. Διαφορετικά, θέτουμε $X_i = X_{i-1}$.

Έξοδος: Μέγιστη τιμή $h^* = \max_i h(X_i)$.

Παράδειγμα 5.17. Θέλουμε να μεγιστοποιήσουμε τη συνάρτηση $h(x) = [\cos(50x) + \sin(20x)]^2$ για $x \in [0, 1]$. Θεωρούμε την προτείνουσα τυχαία μεταβλητή $(Y \mid X_{i-1} = x_{i-1}) \sim \text{Unif}\left[x_{i-1} - s_i, x_{i-1} + s_i\right]$ με δεσμευμένη συνάρτηση πυκνότητας πιθανότητας:

$$g\left(y \mid x_{i-1}\right) = \frac{1}{2 s_i}.$$

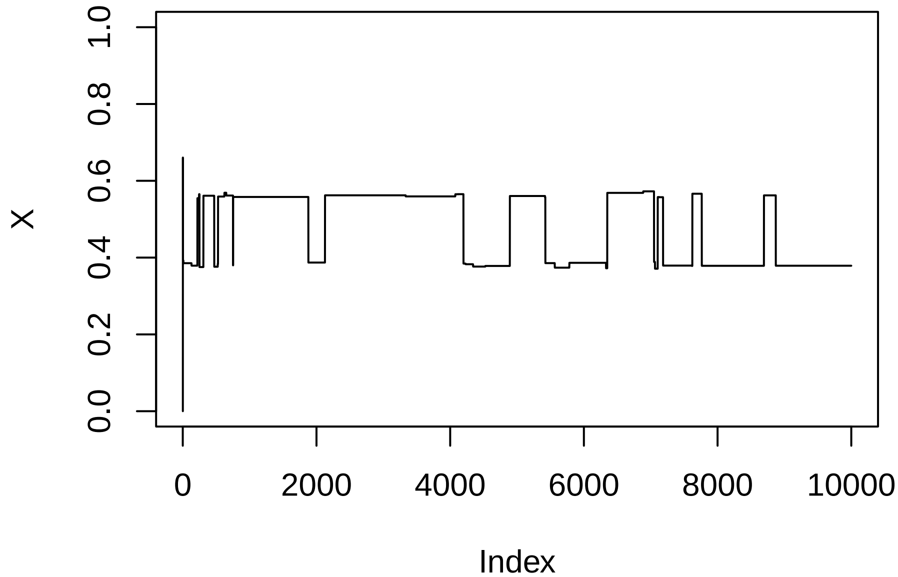
```
h = function(x) {
    ifelse(x >= 0 & x <= 1, (cos(50 * x) + sin(20 * x))^2, 0)
}

curve(h(x), lwd = 2)
```




```r
optimize(h, c(0, 1), maximum = TRUE)
```

```
## $maximum
## [1] 0.379125
##
## $objective
## [1] 3.832543
```

```r
n = 10000
lambda1 = 1
X = numeric(n)
for (i in 2:n) {
    lambda = lambda1 * log(i)
    s = sqrt(lambda)
    V = runif(1)
    Y = 2 * s * V + X[i - 1] - s
    logA = lambda * (h(Y) - h(X[i - 1]))
    U = runif(1)
    X[i] = ifelse(log(U) < logA, Y, X[i - 1])
}
plot(X, type = "l", ylim = c(0, 1))
```




```
hstar = max(h(X))
print(hstar)
```

```
## [1] 3.832543
```

```
I = which(h(X) == hstar)
print(unique(X[I]))
```

```
## [1] 0.3791546
```

6 Μέθοδος Bootstrap

Ορισμός 6.1. Έστω τυχαίο δείγμα $X_1, X_2, \dots, X_n$. Για $x \in \mathbb{R}$, ορίζουμε την εμπειρική συνάρτηση κατανομής του τυχαίου δείγματος ως:

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{X_i \leqslant x\}}.$$

Έστω τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από συνάρτηση κατανομής F με παράμετρο θ. Θεωρούμε μία εκτιμήτρια $T(X)$ του θ. Θέλουμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}[h(T)]$. Υποθέτουμε ότι δεν είναι αποδοτικό να προσομοιώσουμε από την F. Αντιθέτως, μπορούμε εύκολα να προσομοιώσουμε τυχαία δείγματα $X_*^{(1)}, X_*^{(2)}, \dots, X_*^{(B)}$ από την εμπειρική συνάρτηση κατανομής F_n του τυχαίου δείγματος $X_1, X_2, \dots, X_n$. Αν $T_j^* = T\left(X_*^{(j)}\right)$ για $j = 1, 2, \dots, B$, τότε μπορούμε να εκτιμήσουμε τη μέση τιμή $\mathbb{E}[h(T)]$ ως:

$$\frac{1}{B}\sum_{j=1}^{B} h\left(T_j^*\right).$$

Algorithm 6.1 Μη-Παραμετρικό Bootstrap

Είσοδος: Τυχαίο δείγμα X, στατιστική συνάρτηση $T(X)$ και μέγεθος δείγματος bootstrap B.

Για $j = 1, 2, \dots, B$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε $U_1, \dots, U_n \sim \text{Unif}[0,1]$ και θέτουμε $I_i = \lfloor nU_i \rfloor + 1$ για $i = 1, 2, \dots, n$.

ii: Θέτουμε $X_*^{(j)} = \left(X_{I_1}, X_{I_2}, \dots, X_{I_n}\right)$ και $T_j^* = T\left(X_*^{(j)}\right)$.

Έξοδος: Bootstrap στατιστικές συναρτήσεις $T_1^*, T_2^*, \dots, T_B^*$.

Παράδειγμα 6.1. Έστω ακολουθία ανεξάρτητων τυχαίων μεταβλητών $U_1, U_2, \dots \sim \text{Unif}[0,1]$. Ορίζουμε την τυχαία μεταβλητή:

$$X = \sup\left\{k \in \mathbb{N} : U_1 < U_2 < \dots < U_{k-1}\right\}.$$

Θεωρούμε την παράμετρο $\theta = \sqrt{\mathbb{E}\left(X^2\right)}$ και τυχαίο δείγμα $X_1, X_2, \dots, X_n$ από την κατανομή του X. Ορίζουμε μία εκτιμήτρια του θ ως:

$$T(X) = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n} X_i^2}.$$

Θέλουμε να εκτιμήσουμε τη μέση τιμή, τη διασπορά, τη μεροληψία και το μέσο τετραγωνικό σφάλμα της εκτιμήτριας $T(X)$. Επιπλέον, θέλουμε να κατασκευάσουμε ένα $100(1-\alpha)\%$ διάστημα εμπιστοσύνης για το θ.

```
X = c(2, 4, 3, 2, 2, 2, 2, 3, 3, 2)
n = length(X)
t = sqrt(mean(X^2))
B = 10000
Tstar = numeric(B)
for (j in 1:B) {
    U = runif(n)
    I = floor(n * U) + 1
```

```r
    Xstar = X[I]
    Tstar[j] = sqrt(mean(Xstar^2))
}
Tbar = mean(Tstar)
print(Tbar)
```

```
## [1] 2.577908
```

```r
VarT = mean((Tstar - Tbar)^2)
print(VarT)
```

```
## [1] 0.05363952
```

```r
bias = mean(Tstar - t)
print(bias)
```

```
## [1] -0.01052766
```

```r
MSE = mean((Tstar - t)^2)
print(MSE)
```

```
## [1] 0.05375035
```

Ορίζουμε:

$$\overline{T}^* = \frac{1}{B}\sum_{j=1}^{b} T_j^*, \quad S_*^2 = \frac{1}{B}\sum_{j=1}^{B}\left(T_j^* - \overline{T}^*\right)^2.$$

Τότε, παίρνουμε το εξής κανονικό bootstrap διάστημα εμπιστοσύνης για το θ:

$$\left[\overline{T}^* - z_{1-\alpha/2}S_*, \overline{T}^* + z_{1-\alpha/2}S_*\right].$$

Εναλλακτικά, μπορούμε να χρησιμοποιήσουμε το εξής ποσοστημοριακό bootstrap διάστημα εμπιστοσύνης για το θ:

$$\left[T^*_{(\lfloor B\alpha/2\rfloor)}, T^*_{(\lceil B(1-\alpha/2)\rceil)}\right].$$

Τέλος, μπορούμε να χρησιμοποιήσουμε το εξής βασικό bootstrap διάστημα εμπιστοσύνης για το θ:

$$\left[2T - T^*_{(\lceil B(1-\alpha/2)\rceil)}, 2T - T^*_{(\lfloor B\alpha/2\rfloor)}\right].$$

```r
alpha = 0.05
I = c(Tbar - qnorm(1 - alpha/2) * sqrt(VarT), Tbar + qnorm(1 - alpha/2) * sqrt(VarT))
print(I)
```

```
## [1] 2.123976 3.031840
```

```r
Tstar = sort(Tstar)
I = c(Tstar[floor(B * alpha/2)], Tstar[ceiling(B * (1 - alpha/2))])
print(I)
```

```
## [1] 2.121320 3.016621
```

```
I = c(2 * t - Tstar[ceiling(B * (1 - alpha/2))], 2 * t - Tstar[floor(B * alpha/2)])
print(I)
```

```
## [1] 2.160251 3.055551
```

Παράδειγμα 6.2. Έστω τυχαίο δείγμα $X_1, \dots, X_n \sim \mathcal{N}(\mu, \sigma^2)$. Γνωρίζουμε ότι η στατιστική συνάρτηση του μονόπλευρου ελέγχου υποθέσεων $H_0 : \mu = \mu_0$ vs. $H_1 : \mu < \mu_0$ ισούται με:

$$T(X) = \frac{\overline{X} - \mu_0}{S/\sqrt{n}}.$$

```
n = 20
mu = 0
sigma = 2
U = runif(n/2)
D = -2 * log(U)
V = runif(n/2)
Theta = 2 * pi * V
Z = sqrt(D) * c(cos(Theta), sin(Theta))
X = sigma * Z + mu
```

Algorithm 6.2 Κανονικός Έλεγχος Υποθέσεων με Χρήση Μη-Παραμετρικού Bootstrap

Είσοδος: Τυχαίο δείγμα X, στατιστική συνάρτηση $T(X)$ και μέγεθος δείγματος bootstrap B.

1: Για $j = 1, 2, \dots, B$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε $U_1, \dots, U_n \sim \text{Unif}[0, 1]$ και θέτουμε $I_i = \lfloor nU_i \rfloor + 1$ για $i = 1, 2, \dots, n$.

ii: Θέτουμε $X_*^{(j)} = \left(X_{I_1}, X_{I_2}, \dots, X_{I_n}\right)$.

iii: Υπολογίζουμε τον δειγματικό μέσο $\overline{X}_j^*$ και την τετραγωνική ρίζα S_j^* της δειγματικής διασποράς του $X_*^{(j)}$.

iv: Υπολογίζουμε τη στατιστική συνάρτηση:

$$T_j^* = \frac{\overline{X}_j^* - \overline{X}}{S_j^*/\sqrt{n}}.$$

2: Υπολογίζουμε:

$$N^* = \sum_{j=1}^{B} \mathbb{1}_{\{T_j^* \leqslant T\}}, \quad p^* = \frac{N^* + 1}{B + 1}.$$

Έξοδος: Εκτιμώμενο p-value p^* του ελέγχου υποθέσεων.

```
mu0 = 1
alpha = 0.05
Xbar = mean(X)
S = sd(X)
```

```r
t = (Xbar - mu0) * sqrt(n)/S
pval = pt(t, n - 1)
print(pval)
```

```
## [1] 0.01179182
```

```r
B = 10000
Tstar = numeric(B)
for (j in 1:B) {
    U = runif(n)
    I = floor(n * U) + 1
    Xstar = X[I]
    Xbarstar = mean(Xstar)
    Sstar = sd(Xstar)
    Tstar[j] = (Xbarstar - Xbar) * sqrt(n)/Sstar
}
pval = (sum(Tstar <= t) + 1)/(B + 1)
print(pval)
```

```
## [1] 0.00909909
```

Algorithm 6.3 Κανονικός Έλεγχος Υποθέσεων με Χρήση Παραμετρικού Bootstrap

Είσοδος: Τυχαίο δείγμα X, στατιστική συνάρτηση $T(X)$ και μέγεθος δείγματος bootstrap B.

1: Υπολογίζουμε την εκτίμηση μέγιστης πιθανοφάνειας του σ^2 υπό την $H_0 : \mu = \mu_0$ ως:

$$\hat{\sigma}_0^2 = \frac{1}{n}\sum_{i=1}^{n}\left(X_i - \mu_0\right)^2.$$

2: Για $j = 1, 2, \dots, B$, επαναλαμβάνουμε τα παρακάτω βήματα:

i: Προσομοιώνουμε τυχαίο δείγμα $X_*^{(j)}$ μεγέθους n από την κατανομή $N\left(\mu_0, \hat{\sigma}_0^2\right)$.

ii: Υπολογίζουμε τον δειγματικό μέσο $\overline{X}_j^*$ και την τετραγωνική ρίζα S_j^* της δειγματικής διασποράς του $X_*^{(j)}$.

iii: Υπολογίζουμε τη στατιστική συνάρτηση:

$$T_j^* = \frac{\overline{X}_j^* - \mu_0}{S_j^*/\sqrt{n}}.$$

3: Υπολογίζουμε:

$$N^* = \sum_{j=1}^{B}\mathbb{1}_{\{T_j^* \leqslant T\}}, \quad p^* = \frac{N^*+1}{B+1}.$$

Έξοδος: Εκτιμώμενο p-value p^* του ελέγχου υποθέσεων.

```
sigma0 = sqrt(mean((X - mu0)^2))
for (j in 1:B) {
    U = runif(n/2)
    D = -2 * log(U)
    V = runif(n/2)
    Theta = 2 * pi * V
    Z = sqrt(D) * c(cos(Theta), sin(Theta))
    Xstar = sigma0 * Z + mu0
    Xbarstar = mean(Xstar)
    Sstar = sd(Xstar)
    Tstar[j] = (Xbarstar - mu0) * sqrt(n)/Sstar
}
pval = (sum(Tstar <= t) + 1)/(B + 1)
print(pval)
```

```
## [1] 0.01129887
```