Méthodes de transformation

Loi normale (Box-Muller)

Loi Normale $(0, 1)$:

On sait que si :

X_1 suit une loi uniforme $(0, 1)$

X_2 suit une loi uniforme $(0, 1)$

X_1 et X_2 indépendantes

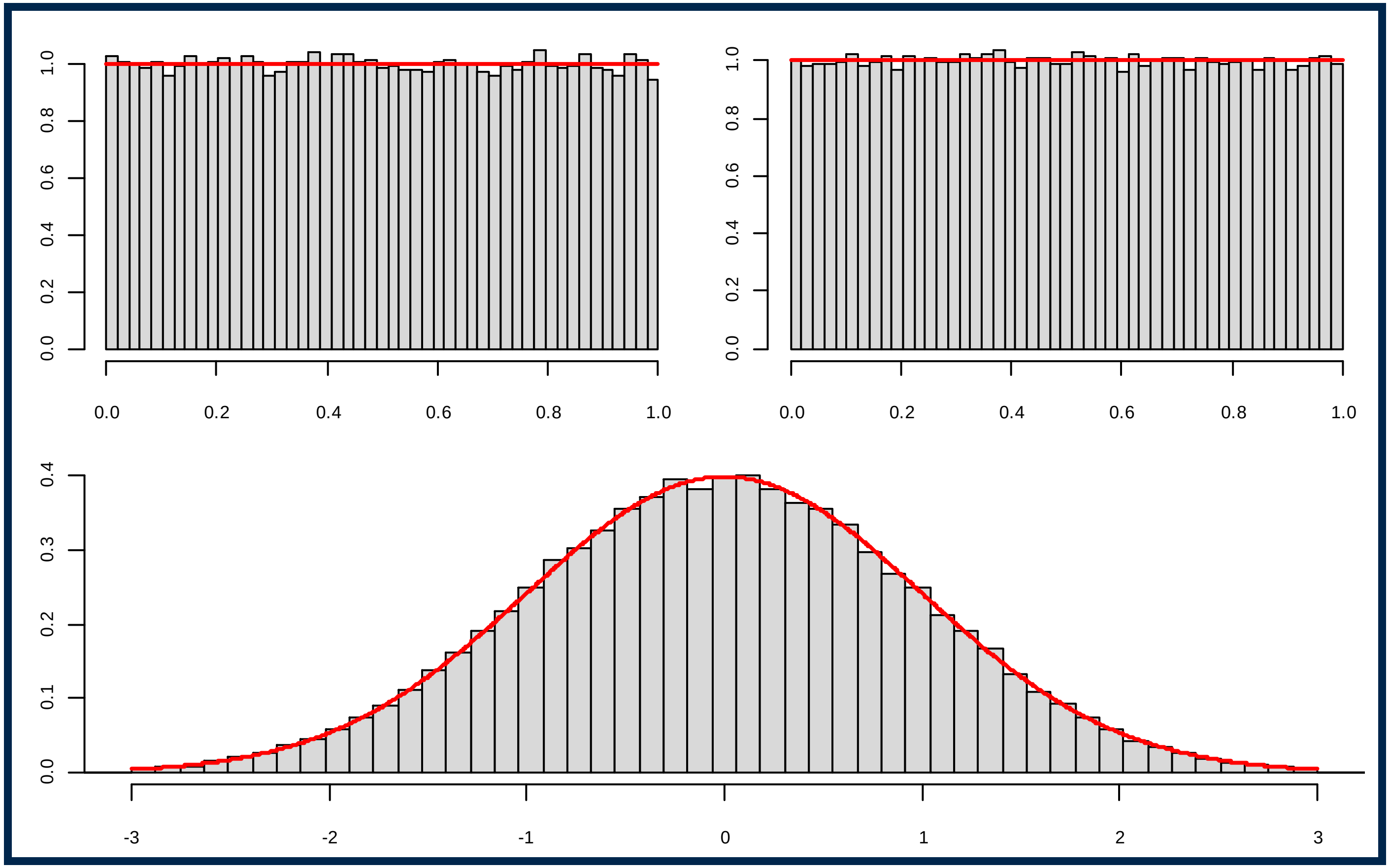
Alors :

$\sqrt{-2\ln\left(X_1\right)} \cdot \cos\left(2\pi X_2\right)$ suit une loi Normale $(0, 1)$

Méthode de transformation

Loi normale (Box-Muller) – Implémentation R

```
n <- 100000; x1 <- runif(n); x2 <- runif(n)
truehist(sqrt(-2*log(x1))*cos(2*pi*x2))
```



Méthodes *ad hoc*

Loi de Cauchy

Loi de Cauchy $(x_0 = 0, a = 1)$:

On sait que si :

X_1 suit une loi Normale $(0, 1)$
X_2 suit une loi Normale $(0, 1)$
X_1 et X_2 indépendantes

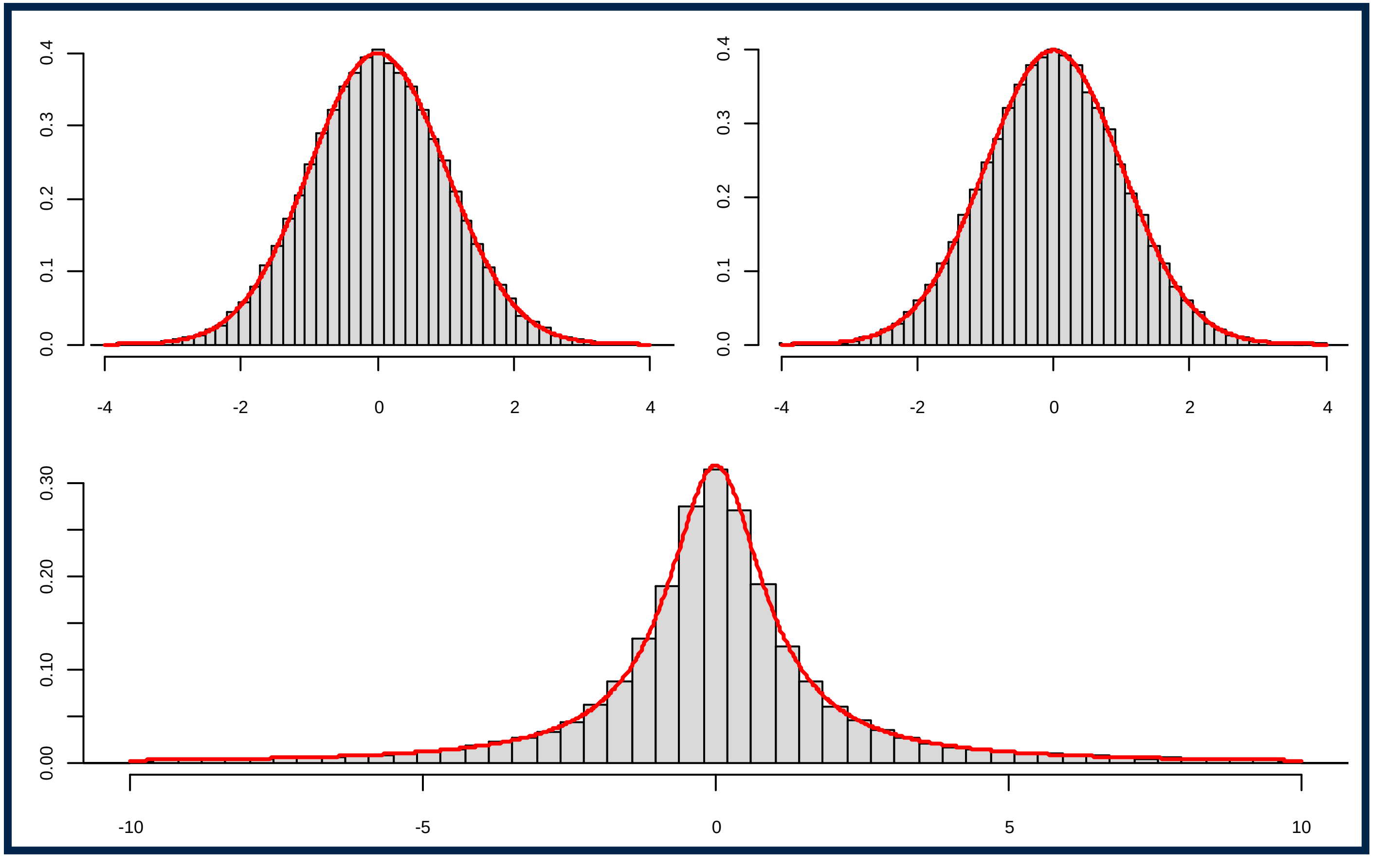
Alors :

X_1 / X_2 suit une loi de Cauchy $(0, 1)$

Méthodes *ad hoc*

Loi de Cauchy – Implémentation R

```
n <- 100000; x1 <- rnorm(n); x2 <- rnorm(n)
truehist(x1 / x2)
```



Méthodes *ad hoc*

Loi de χ^2

Loi de $\chi^2\ (n)$:

On sait que si :

$X_1, X_2\ , \ldots, X_n$ sont n v.a.r. i.i.d. Normale $(0, 1)$

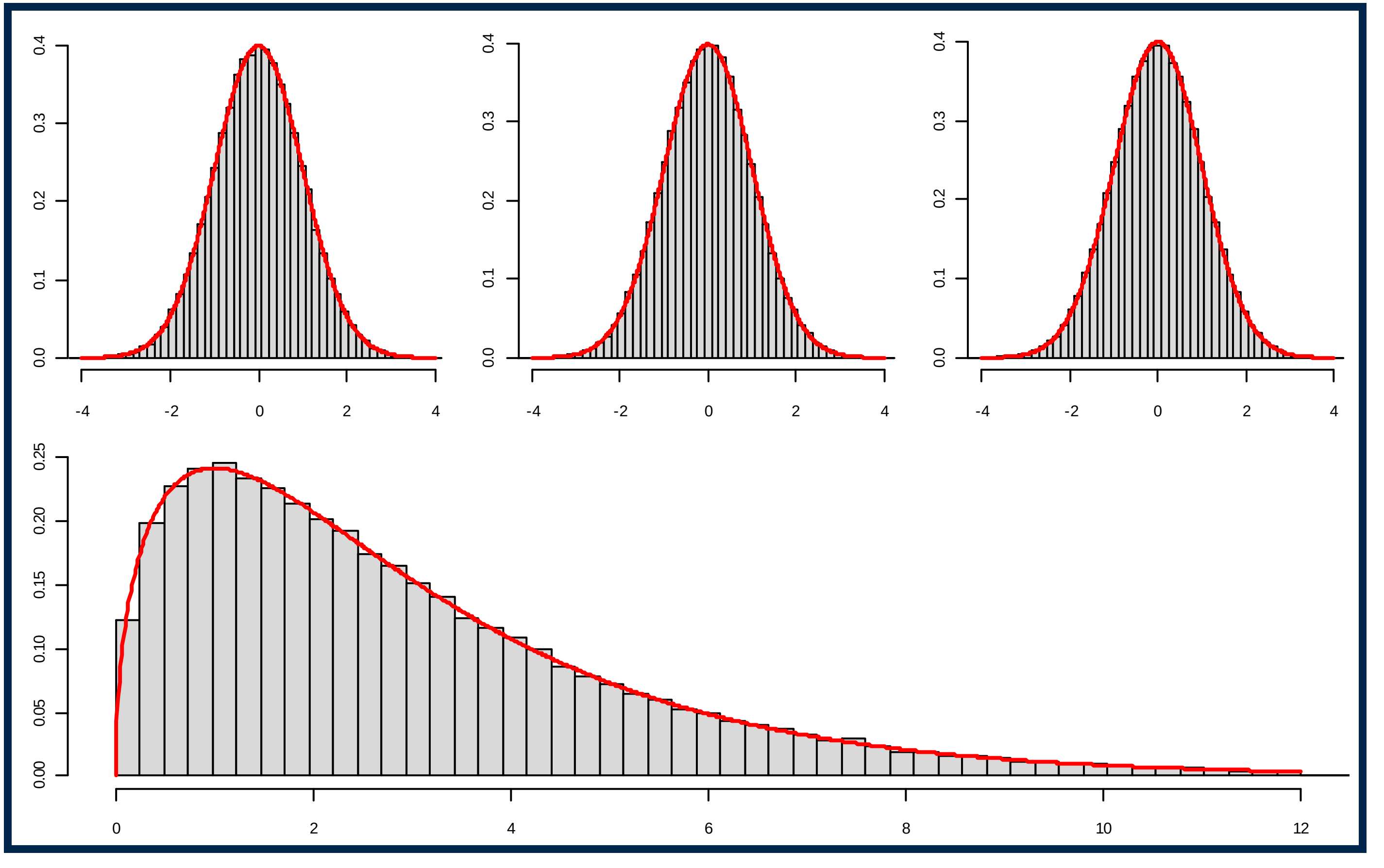
Alors :

$X_1 + X_2 + \ldots + X_n$ suit une loi de $\chi^2\ (n)$

Méthodes *ad hoc*

Loi de χ^2 – Implémentation R

```
n <- 100000; x1 <- rnorm(n); x2 <- rnorm(n); x3 <- rnorm(n)
truehist(x1^2 + x2^2 + x3^2)
```



Méthodes *ad hoc*

Loi de Student

Loi de Student (n) :

On sait que si :

X_1 suit une loi Normale $(0, 1)$

X_2 suit une loi de χ^2 (n)

X_1 et X_2 indépendantes

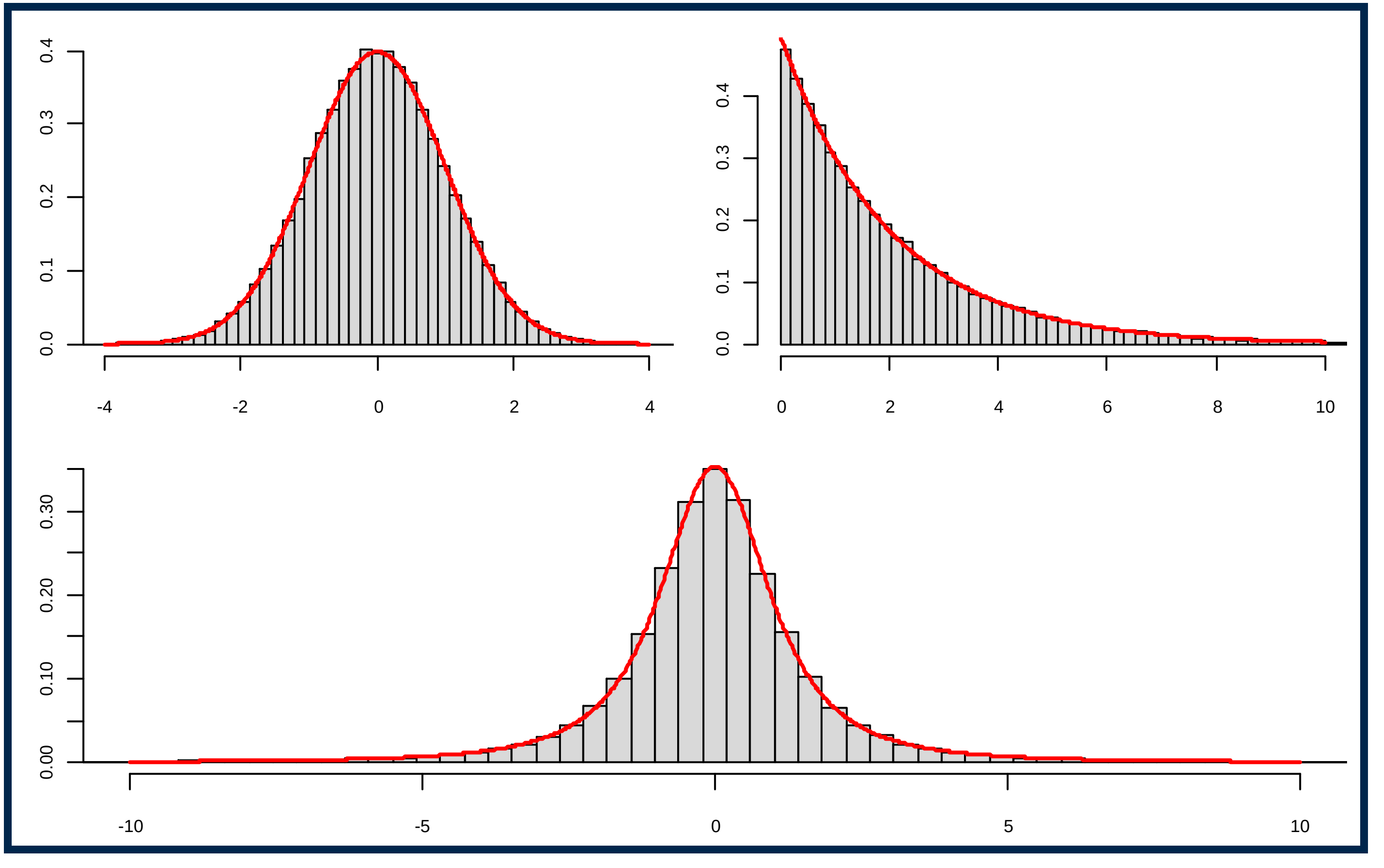
Alors :

X_1 / sqrt(X_2 / n) suit une loi de Student (n)

Méthodes *ad hoc*

Loi de Student – Implémentation R

```
n <- 100000; x1 <- rnorm(n); x2 <- rchisq(n, 2)
truehist(x1 / sqrt(x2 / 2))
```



Méthodes *ad hoc*

Loi de Beta (α, β)

Loi de Beta (α, β) :

On sait que si :

X_1 suit une loi Gamma (α, 1)

X_1 suit une loi Gamma (β, 1)

X_1 et X_2 indépendantes

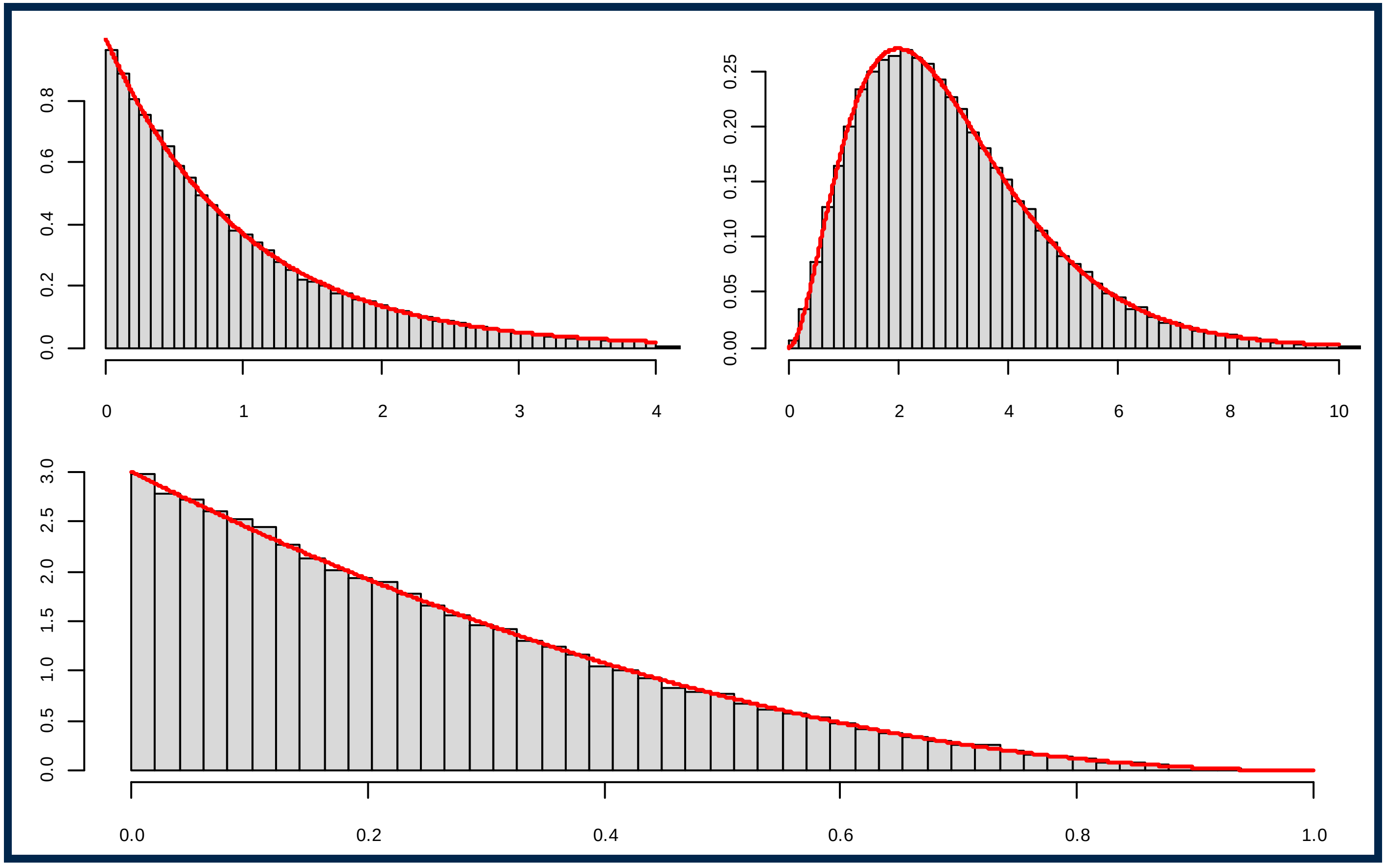
Alors :

$X_1 / (X_1 + X_2)$ suit une loi Beta (α, β)

Méthodes *ad hoc*

Loi Beta – Implémentation R

```
n <- 100000; x1 <- rgamma(n, 1, 1); x2 <- rgamma(n, 3, 1)
truehist(x1 / (x1 + x2))
```



Lois multidimensionnelles

Loi à simuler : $(X_1, X_2, \ldots, X_p) \sim f_{1, 2, \ldots, p}$

avec (éventuellement) X_i non indépendante de X_j

Il suffit de remarquer que :

$$f_{1, 2, \ldots, p}(x_1, x_2, \ldots, x_p) = f_1(x_1)\; f_{2|1}(x_2 \mid x_1) \ldots f_{p|1, 2, \ldots, p\text{-}1}(x_p \mid x_1, x_2, \ldots, x_{p-1})$$

Simulation :

- Simuler $X_1 \sim f_1$ Valeur simulée : x_1
- Simuler $X_2 \sim f_{2|X1=x1}$ Valeur simulée : x_2
- etc.

Lois multidimensionnelles

Exemple : loi multinomiale

Loi à simuler : $(X_1, X_2, X_3) \sim \text{Multinomiale}(n, p_1, p_2, p_3)$

Simulation :

- Simuler $X_1 \sim \text{Binomiale}\ (n, p_1)$

  Valeur simulée : x_1

- Simuler $X_2 \sim \text{Binomiale}\ (n - x_1, p_2 / (1 - p_1))$

  Valeur simulée : x_2

- Simuler $X_3 \sim \text{Binomiale}\ (n - x_1 - x_2, p_3 / (1 - p_1 - p_2) = 1)$

  Valeur simulée : $x_3 = n - x_1 - x_2$

Lois multidimensionnelles

Exemple : loi bi-normale centrée-réduite (ρ)

Loi à simuler : $(X_1, X_2) \sim$ Bi-Normale centrée-réduite (ρ)

Simulation :

- Simuler $X_1 \sim$ Normale $(0,1)$

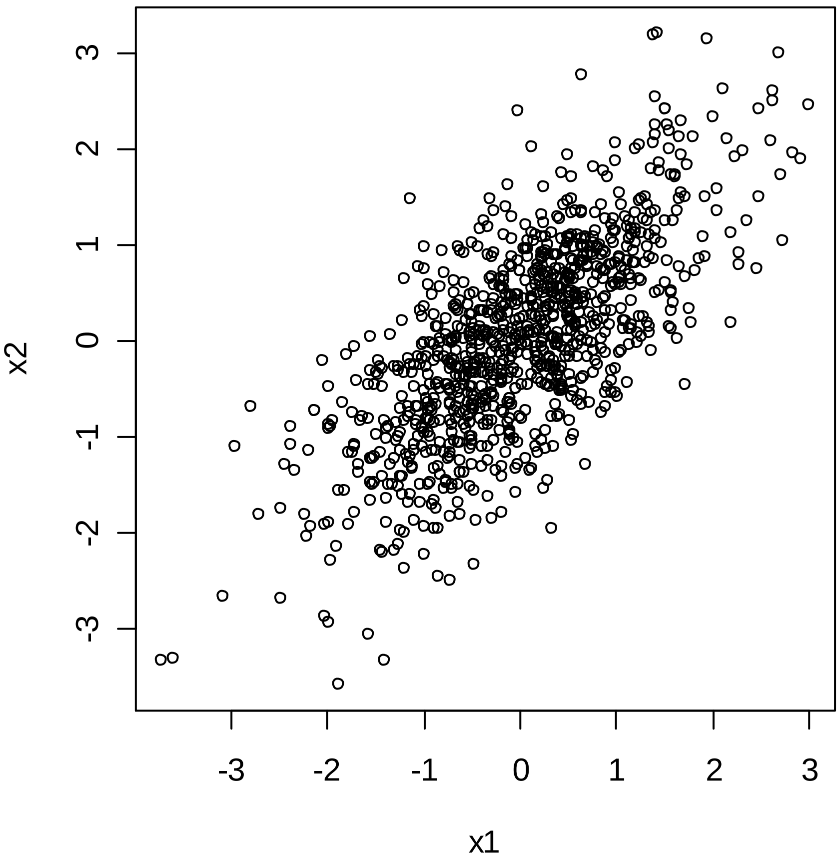
  Valeur simulée : x_1

- Simuler $X_2 \sim$ Normale $(\rho x_1, 1- \rho^2)$

  Valeur simulée : x_2

Lois multidimensionnelles

Exemple : loi bi-normale – Implémentation R




```
n <- 1000
rho <- 0.7
x1 <- rnorm(n,0,1)
x2 <- rnorm(n,x1*rho,sqrt(1-rho^2))
plot(x1,x2)
```

Simulations de variables aléatoires

Méthode du rejet

f fonction de densité d'une variable de support D à simuler

g fonction de densité d'une variable X de même support D :

- que l'on sait simuler
- qui vérifie : $f(x) \leq m \cdot g(x)$

- Simuler (i.i.d.): $X_1, X_2, \ldots \sim g$ et $U_1, U_2, \ldots \sim$ *Uniforme sur [0,1]*

- Accepter : x_i si $u_i \leq \dfrac{f(x_i)}{m \cdot g(x_i)}$

En d'autres termes la loi conditionnelle de X sachant $U \leq \dfrac{f(X)}{m \cdot g(X)}$ a comme densité f

La probabilité d'acceptation est : $1 / m$

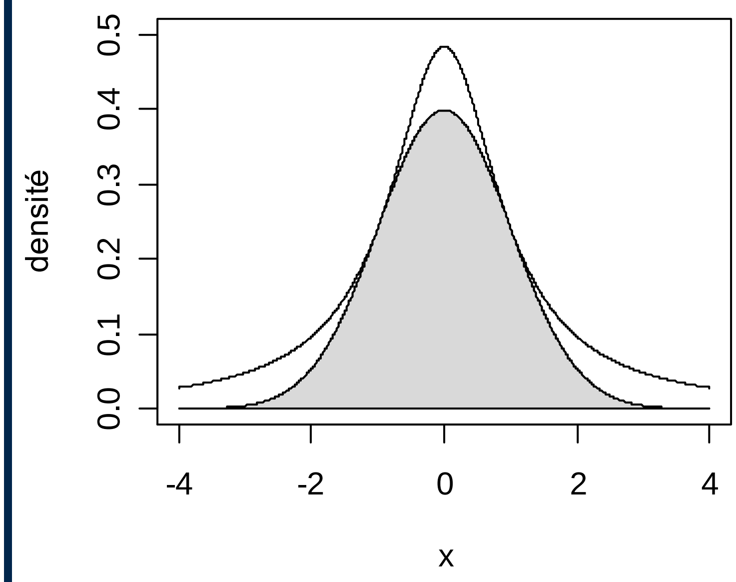
Méthode du rejet

Exemple : loi normale

Loi Normale (0,1) $$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right)$$

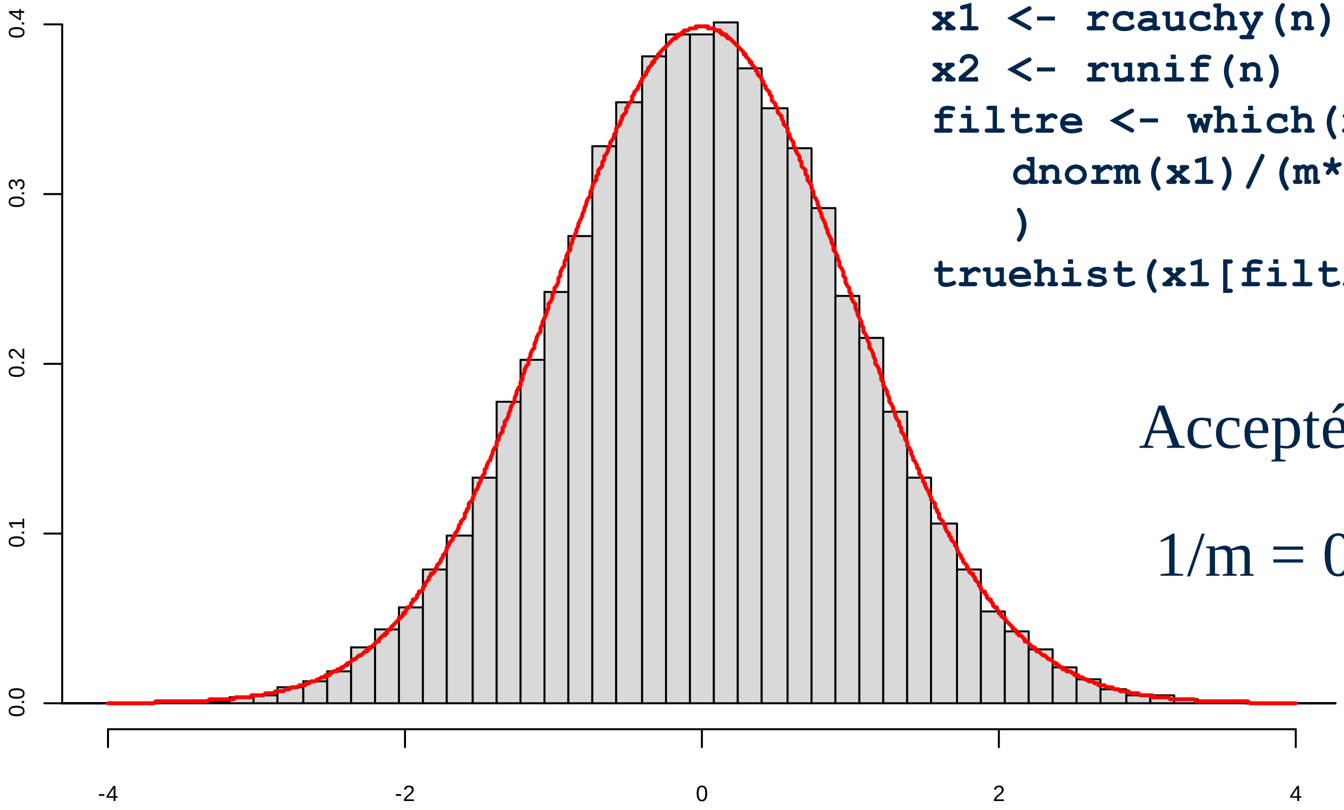
Loi de Cauchy (0, 1) $$g(x) = \frac{1}{\pi\left(1+x^2\right)}$$

$$\frac{f(x)}{g(x)} = \sqrt{\frac{\pi}{2}} \cdot \left(1+x^2\right) \cdot \exp\left(-\frac{1}{2}x^2\right) \leq \sqrt{\frac{2\pi}{e}}$$



Méthode du rejet

Exemple : loi normale – Implémentation R




```
n <- 100000
m <- sqrt(2*pi/exp(1))
x1 <- rcauchy(n)
x2 <- runif(n)
filtre <- which(x2 <=
    dnorm(x1)/(m*dcauchy(x1))
    )
truehist(x1[filtre])
```

Acceptées = 66041

1/m = 0.6577446

Algorithme de Metropolis-Hastings

Le problème :

Générer un échantillon de valeurs de densité cible f

Condition nécessaire :

Savoir calculer $f(x)$ pour tout x

Initialisation : valeur x_0

Loi de proposition : $g(y \mid x)$

Pour chaque x_i, on propose une valeur y_i suivant la loi de proposition

On accepte ou on rejette y_i suivant une règle d'acceptation-rejet :

Si rejet : $x_{i+1} = x_i$

Si acceptation : $x_{i+1} = y_i$

Algorithme de Metropolis-Hastings

Règle d'acceptation-rejet

Acceptation de y_i avec un probabilité :

$$p = \min\left(1, \frac{f(y_i)\,g(x_i \mid y_i)}{f(x_i)\,g(y_i \mid x_i)}\right)$$

La loi de proposition peut être :

- symétrique : $g(y \mid x) = g(x \mid y)$ $\qquad p = \min\left(1, \frac{f(y_i)}{f(x_i)}\right)$

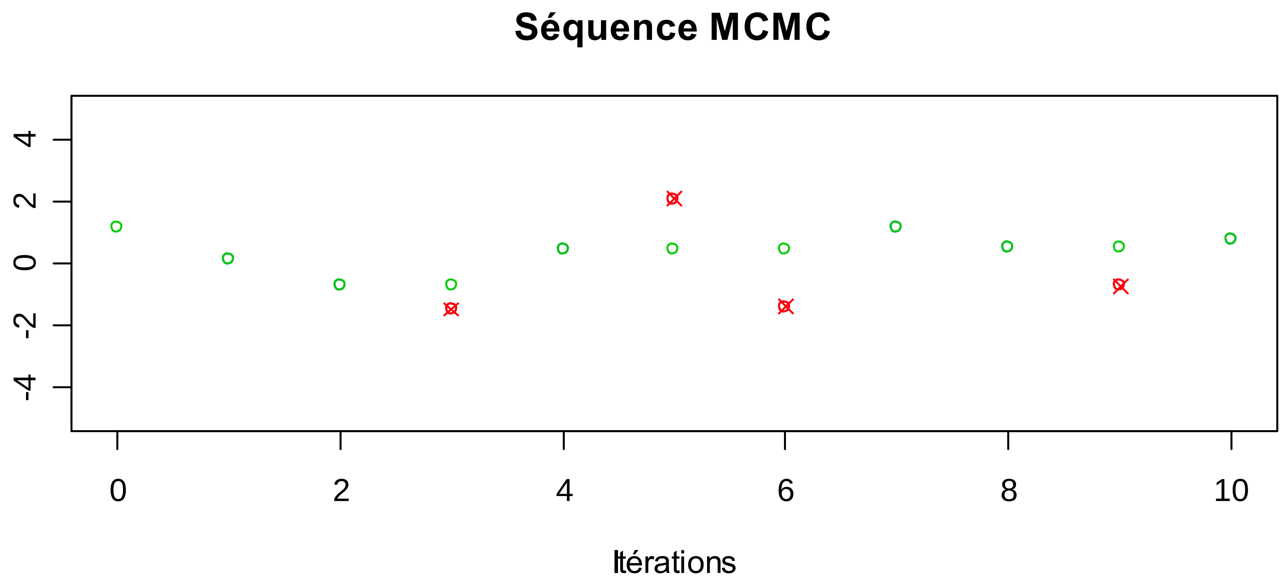
- indépendante : $g(y \mid x)$ ne dépend pas de x

Algorithme de Metropolis-Hastings

Exemple

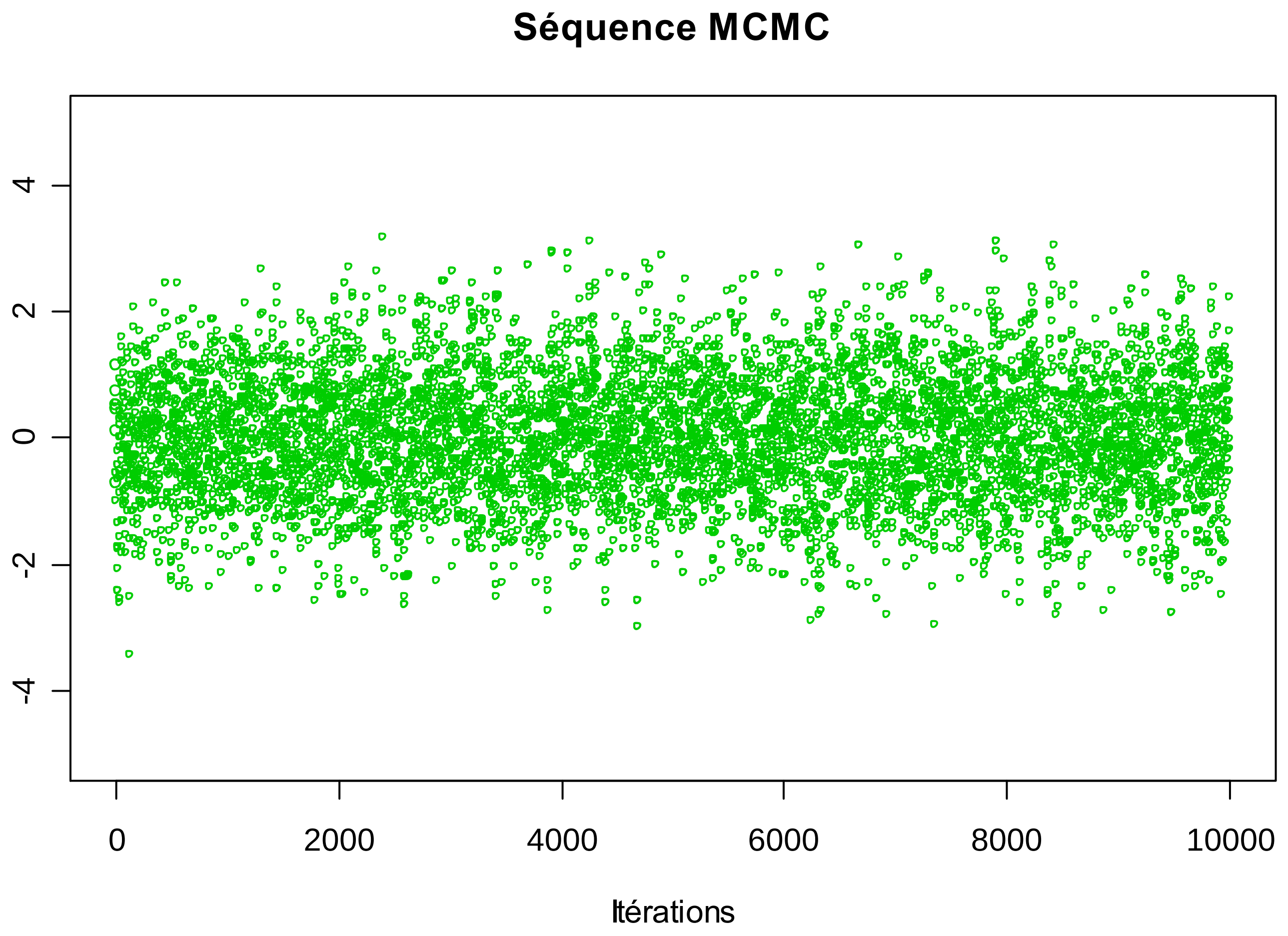
Densité cible : $f \sim N(0,1)$

Loi de proposition : $g(y|x)$ Uniforme sur $[x\text{-}\delta, x+\delta]$



Algorithme de Metropolis-Hastings

Exemple



Algorithme de Metropolis-Hastings

Exemple