

Simulations de variables aléatoires

Pierre Gloaguen

07/04/2020

Annonces:

- ▶ Premier rendu pour le 13 Avril à midi (exo5 du TD1)
- ▶ À faire en binome et à rendre sur Ecampus
- ▶ TD sur échantillonnage préférentiel en ligne sur ma page.
- ▶ Rendre l'exo 3 pour le 21 Avril au soir.

Rappel des précédents

- ▶ Nécessité en statistiques d'évaluer des espérances;
- ▶ Principes des méthodes de Monte Carlo:
 - ▶ Classiques et échantillonnage préférentiel.
- ▶ Approximation d'intégrales (espérances) par simulation de variables aléatoires.
- ▶ Fonctionne grâce à la loi des grands nombres, IC grâce au TCL.

Rappel des précédents

- ▶ Nécessité en statistiques d'évaluer des espérances;
- ▶ Principes des méthodes de Monte Carlo:
 - ▶ Classiques et échantillonnage préférentiel.
- ▶ Approximation d'intégrales (espérances) par simulation de variables aléatoires.
- ▶ Fonctionne grâce à la loi des grands nombres, IC grâce au TCL.
- ▶ Comment simule t'on ces lois?
 - ▶ Lois usuelles implémentées dans R (ou autre. . .)
 - ▶ Comment est ce fait?
 - ▶ Pour des lois non usuelles, comment faire?

Objectif du cours

- ▶ Comment simuler une loi uniforme continue avec un ordinateur?
 - ▶ Générateurs pseudo aléatoires;
- ▶ Comment simuler des lois génériques à partir de lois uniformes?
 - ▶ Méthode d'inversion;
- ▶ Comment simuler des lois non classique à partir de lois simulables?
 - ▶ Méthode d'acceptation rejet.

Générateurs pseudo aléatoires

Simulation de variables aléatoires par ordinateur

Objectif: Simuler une suite de variables aléatoires X_1, \dots, X_M telles qu'elles soient:

- ▶ Distribuées selon une même loi donnée:
- ▶ Mutuellement indépendantes.

Simulation de variables aléatoires par ordinateur

Objectif: Simuler une suite de variables aléatoires X_1, \dots, X_M telles qu'elles soient:

- ▶ Distribuées selon une même loi donnée:
- ▶ Mutuellement indépendantes.
- ▶ Une telle simulation est faite selon un algorithme **déterministe**;

Simulation de variables aléatoires par ordinateur

Objectif: Simuler une suite de variables aléatoires X_1, \dots, X_M telles qu'elles soient:

- ▶ Distribuées selon une même loi donnée:
- ▶ Mutuellement indépendantes.
- ▶ Une telle simulation est faite selon un algorithme **déterministe**;
- ▶ À l'heure actuelle, un ordinateur ne peut que **mimer l'aléa**;

Simulation de variables aléatoires par ordinateur

Objectif: Simuler une suite de variables aléatoires X_1, \dots, X_M telles qu'elles soient:

- ▶ Distribuées selon une même loi donnée:
- ▶ Mutuellement indépendantes.
- ▶ Une telle simulation est faite selon un algorithme **déterministe**;
- ▶ À l'heure actuelle, un ordinateur ne peut que **mimer l'aléa**;
- ▶ **Mimer l'aléa:** Pour un échantillon de loi donnée:
 - ▶ Passer les tests statistiques usuels d'adéquations (test du χ^2 , test de Kolmogorov Smirnov);
 - ▶ Passer les tests d'indépendances usuels (test du χ^2 , test de corrélation linéaire, ...).

Générateur de la loi uniforme $\mathcal{U}[0, 1]$

- ▶ En pratique, la loi “atomique” est la loi $\mathcal{U}[0, 1]$;

Générateur de la loi uniforme $\mathcal{U}[0, 1]$

- ▶ En pratique, la loi “atomique” est la loi $\mathcal{U}[0, 1]$;
- ▶ Si on sait simuler selon cette loi, par différentes méthodes on pourra se ramener aux autres.

Générateur de la loi uniforme $\mathcal{U}[0, 1]$

- ▶ En pratique, la loi “atomique” est la loi $\mathcal{U}[0, 1]$;
- ▶ Si on sait simuler selon cette loi, par différentes méthodes on pourra se ramener aux autres.
- ▶ Nécessité d'un algorithme permettant de mimer un échantillon i.i.d. de loi $\mathcal{U}[0, 1]$.

Algorithme de congruence linéaire

Algorithme basé sur 4 données initiales choisies par l'utilisateur:

- ▶ Un entier $m > 0$, appelé *module*;
- ▶ Un entier $0 < a < m$ appelé *multiplicateur*;
- ▶ Un entier $0 \leq c < m$ appelé *incrément*;
- ▶ Un entier $0 \leq x_0 < m$ appelé *graine*.

Algorithme de congruence linéaire

Algorithme basé sur 4 données initiales choisies par l'utilisateur:

- ▶ Un entier $m > 0$, appelé *module*;
- ▶ Un entier $0 < a < m$ appelé *multiplicateur*;
- ▶ Un entier $0 \leq c < m$ appelé *incrément*;
- ▶ Un entier $0 \leq x_0 < m$ appelé *graine*.

On créera alors une suite de nombres x_1, \dots, x_n en utilisant la relation de récurrence

$$x_k = (ax_{k-1} + c) \text{ modulo } m.$$

Algorithme de congruence linéaire

Algorithme basé sur 4 données initiales choisies par l'utilisateur:

- ▶ Un entier $m > 0$, appelé *module*;
- ▶ Un entier $0 < a < m$ appelé *multiplicateur*;
- ▶ Un entier $0 \leq c < m$ appelé *incrément*;
- ▶ Un entier $0 \leq x_0 < m$ appelé *graine*.

On créera alors une suite de nombres x_1, \dots, x_n en utilisant la relation de récurrence

$$x_k = (ax_{k-1} + c) \text{ modulo } m.$$

On définit enfin les nombres u_1, \dots, u_n dans l'intervalle $[0, 1]$:

$$u_k = \frac{x_k}{m}, \quad 1 \leq k \leq n.$$

Algorithme de congruence linéaire

```
mon_runif <- function(n , a, m, c, x0){  
  echantillon <- rep(NA, n + 1)  
  # %% est l'opérateur modulo  
  x_vals[1] <- (x0 %% m) # Initialisation  
  for(k in 2:(n + 1)){ # Iteration  
    x_vals[k] <- (a * x_vals[k - 1] + c) %% m  
  }  
  u_vals <- x_vals / m # Mise entre 0 et 1  
  return(u_vals[-1])  
  # On ne retourne pas la graine  
}
```

Choix de a , m , c , x_0

- ▶ À x_0 fixé, la séquence obtenue est **toujours la même**.
 - ▶ En pratique, elle n'est pas demandée à l'utilisateur, mais obtenue en interne.
 - ▶ Exemple: nombre de millisecondes (modulo m) écoulé depuis le 1er Janvier 1970.

Choix de a , m , c , x_0

- ▶ À x_0 fixé, la séquence obtenue est **toujours la même**.
 - ▶ En pratique, elle n'est pas demandée à l'utilisateur, mais obtenue en interne.
 - ▶ Exemple: nombre de millisecondes (modulo m) écoulé depuis le 1er Janvier 1970.
- ▶ Doit couvrir "tout" $[0, 1]$:
 - ▶ $\Rightarrow m$ grand;

Choix de a , m , c , x_0

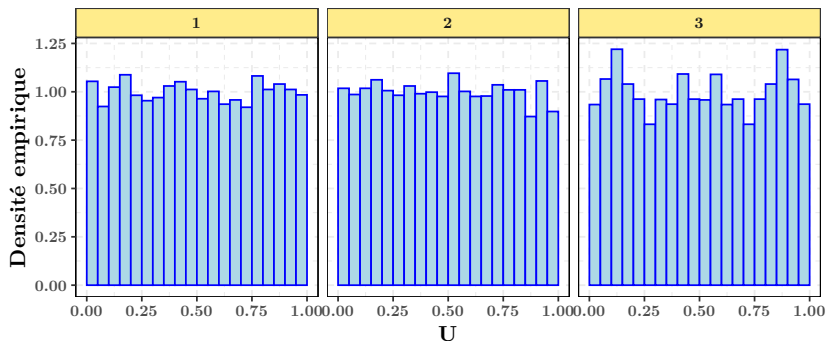
- ▶ À x_0 fixé, la séquence obtenue est **toujours la même**.
 - ▶ En pratique, elle n'est pas demandée à l'utilisateur, mais obtenue en interne.
 - ▶ Exemple: nombre de millisecondes (modulo m) écoulé depuis le 1er Janvier 1970.
- ▶ Doit couvrir "tout" $[0, 1]$:
 - ▶ $\Rightarrow m$ grand;
- ▶ La suite est nécessairement **périodique**!
 - ▶ On veut une période "invisible" (mimer l'indépendance);
 - ▶ $\Rightarrow a$ grand **et** relativement premier à m .

Choix de a , m , c , x_0

- ▶ À x_0 fixé, la séquence obtenue est **toujours la même**.
 - ▶ En pratique, elle n'est pas demandée à l'utilisateur, mais obtenue en interne.
 - ▶ Exemple: nombre de millisecondes (modulo m) écoulé depuis le 1er Janvier 1970.
- ▶ Doit couvrir "tout" $[0, 1]$:
 - ▶ $\Rightarrow m$ grand;
- ▶ La suite est nécessairement **périodique**!
 - ▶ On veut une période "invisible" (mimer l'indépendance);
 - ▶ $\Rightarrow a$ grand **et** relativement premier à m .
- ▶ Considération algorithmiques sur le modulo.
- ▶ Voir références poly.

Choix important (distribution)

3 jeux de paramètres (voir TD), donnant 3 suites de 10000 valeurs entre 0 et 1



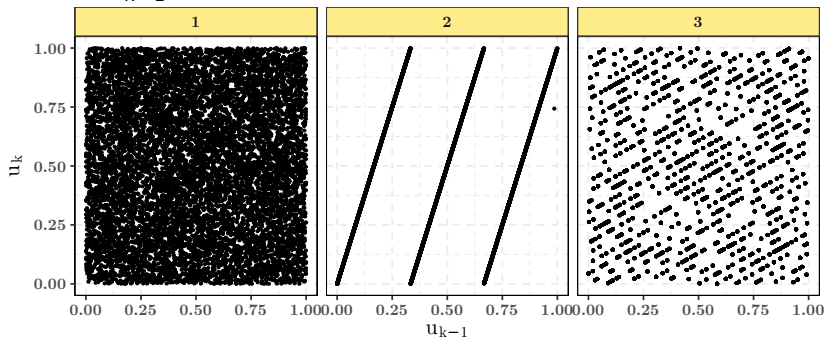
Au risque 5%, avec un test de Kolmogorov-Smirnoff, on rejette

- H_0 : Echantillon de loi uniforme $\mathcal{U}[0, 1]$

seulement pour l'échantillon 3.

Choix important (indépendance)

On regarde, pour les 3 échantillons, la valeur de u_k en fonction de celle de u_{k-1} :



Il y a une forte autocorrélation empirique dans les deux derniers échantillons!

Loi uniforme générique

- ▶ Si on sait simuler $U \sim \mathcal{U}[0, 1]$, alors, pour $a < b \in \mathbb{R}$

$$(b - a)U + a \sim \mathcal{U}[a, b]$$

- ▶ Dans R, la simulation d'une loi uniforme est faite avec `runif`;
- ▶ Dans la suite: on suppose qu'on sait simuler selon $\mathcal{U}[0, 1]$.

Méthode d'inversion

Rappel: Fonction de répartition

Soit X une variable aléatoire à valeurs réelles. Pour tout réel x , on appelle fonction de répartition de X la fonction F_X :

$$\begin{array}{ll} \mathbb{R} & \mapsto [0, 1] \\ x & \mapsto F_X(x) = \mathbb{P}(X \leq x) \end{array}$$

Rappel: Fonction de répartition

Soit X une variable aléatoire à valeurs réelles. Pour tout réel x , on appelle fonction de répartition de X la fonction F_X :

$$\begin{aligned}\mathbb{R} &\mapsto [0, 1] \\ x &\mapsto F_X(x) = \mathbb{P}(X \leq x)\end{aligned}$$

Une fonction de répartition F_X est caractérisée par les propriétés suivantes:

1. F_X est partout continue à droite, i.e. pour tout $x \in \mathbb{R}$:

$$\lim_{\substack{h \rightarrow 0 \\ > 0}} F(x + h) = F(x)$$

2. F_X est croissante.

3. $\lim_{x \rightarrow -\infty} F_X(x) = 0, \lim_{x \rightarrow +\infty} F_X(x) = 1$

Ainsi, toute fonction F sur \mathbb{R} satisfaisant ces conditions est une fonction de répartition.

Exemple: Fonction de répartition

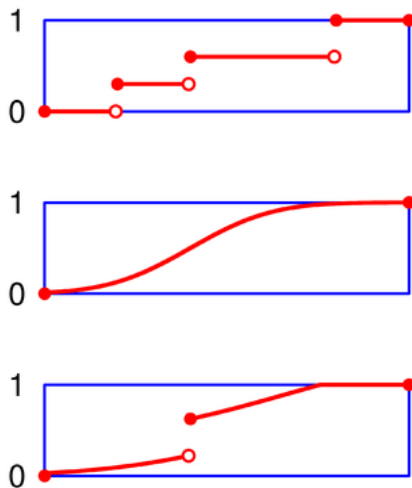


Figure 1: Exemples de fonction de répartition pour une variable aléatoire discrète (haut), continue (centre) ou avec atome (bas). Source *Wikipedia*.

Inverse généralisée de F (fonction quantile)

Soit F une fonction de répartition, on appelle inverse généralisée de F , notée, F^{-1} la fonction:

$$\begin{aligned}]0, 1[&\mapsto \mathbb{R} \\ u &\mapsto F^{-1}(u) = \inf \{z \in \mathbb{R} \text{ tel que } F(z) \geq u\} \end{aligned}$$

Pour une variable aléatoire X , la fonction F_X^{-1} est également appelée *fonction quantile* de la variable aléatoire X . On convient que $F_X^{-1}(0)$ et $F_X^{-1}(1)$ sont la plus petite et la plus grande des valeurs du support de X (éventuellement infinies).

Inverse généralisée

Remarque: Dans le cas d'une fonction de répartition F continue et strictement croissante sur \mathbb{R} , la fonction F^{-1} est simplement l'inverse de F .

Méthode d'inversion

Supposon qu'on ait une variable aléatoire X dont on connaît la fonction de répartiion F , comment simuler X ?

Méthode d'inversion

Supposon qu'on ait une variable aléatoire X dont on connaît la fonction de répartiion F , comment simuler X ?

Exemple: $X \sim \mathcal{Exp}(\lambda)$:

- **Densité:** $f(x) = \lambda e^{-\lambda x} \mathbf{1}_{x \geq 0}$

Méthode d'inversion

Supposon qu'on ait une variable aléatoire X dont on connaît la fonction de répartiion F , comment simuler X ?

Exemple: $X \sim \mathcal{Exp}(\lambda)$:

► **Densité:** $f(x) = \lambda e^{-\lambda x} \mathbf{1}_{x \geq 0}$

► **Fonction de répartition:**

$$F(x) = \int_{-\infty}^x f(z) dz = (1 - e^{-\lambda x}) \mathbf{1}_{x \geq 0}$$

Méthode d'inversion

Supposon qu'on ait une variable aléatoire X dont on connaît la fonction de répartiion F , comment simuler X ?

Exemple: $X \sim \mathcal{Exp}(\lambda)$:

► **Densité:** $f(x) = \lambda e^{-\lambda x} \mathbf{1}_{x \geq 0}$

► **Fonction de répartition:**

$$F(x) = \int_{-\infty}^x f(z) dz = (1 - e^{-\lambda x}) \mathbf{1}_{x \geq 0}$$

► **Inverse généralisée:** $0 < u < 1$, $F^{-1}(u) = -\frac{\ln(1-u)}{\lambda}$

Méthode d'inversion

Supposon qu'on ait une variable aléatoire X dont on connaît la fonction de répartiion F , comment simuler X ?

Exemple: $X \sim \mathcal{Exp}(\lambda)$:

► **Densité:** $f(x) = \lambda e^{-\lambda x} \mathbf{1}_{x \geq 0}$

► **Fonction de répartition:**

$$F(x) = \int_{-\infty}^x f(z) dz = (1 - e^{-\lambda x}) \mathbf{1}_{x \geq 0}$$

► **Inverse généralisée:** $0 < u < 1$, $F^{-1}(u) = -\frac{\ln(1-u)}{\lambda}$

Méthode d'inversion Soit F une fonction de répartition. Soit F^{-1} son inverse généralisée. Soit U une variable aléatoire de loi uniforme sur $[0, 1]$, alors la variable aléatoire

$$X := F^{-1}(U)$$

admet F comme fonction de répartition.

Exemple de méthode d'inversion

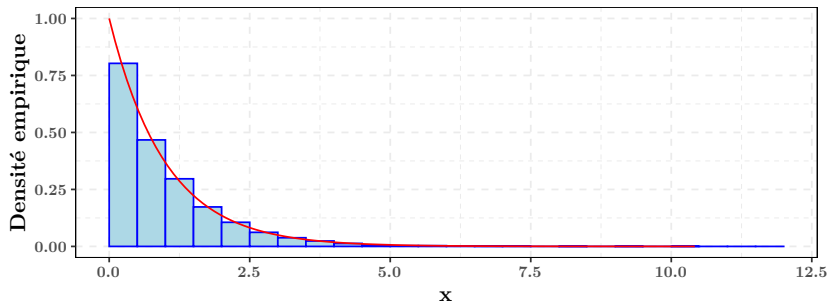
$$F^{-1}(u) = -\frac{\ln(1-u)}{\lambda}$$

```
mon_rexp <- function(n, lambda){  
  # On simule selon une loi uniforme  
  us <- runif(n) # Echantillon IID U[0,1]  
  # On applique la fonction quantile a l'échantillon  
  - log(1 - us) / lambda  
}
```

Exemple de méthode d'inversion

$$F^{-1}(u) = -\frac{\ln(1-u)}{\lambda}$$

Echantillon de taille 10000, $\lambda = 1$



Preuve de la méthode d'inversion

On veut montrer que, pour tout $x \in \mathbb{R}$, si $U \sim \mathcal{U}[0, 1]$, alors

$$\mathbb{P}(F^{-1}(U) \leq x) = F(x).$$

Preuve de la méthode d'inversion

On veut montrer que, pour tout $x \in \mathbb{R}$, si $U \sim \mathcal{U}[0, 1]$, alors

$$\mathbb{P}(F^{-1}(U) \leq x) = F(x).$$

Montrons tout d'abord que, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Leftrightarrow u \leq F(x). \quad (1)$$

Preuve de la méthode d'inversion

On veut montrer que, pour tout $x \in \mathbb{R}$, si $U \sim \mathcal{U}[0, 1]$, alors

$$\mathbb{P}(F^{-1}(U) \leq x) = F(x).$$

Montrons tout d'abord que, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Leftrightarrow u \leq F(x). \quad (1)$$

En effet, si on y parvient, il restera à conclure en se servant de la définition d'une loi uniforme:

$$\mathbb{P}(F^{-1}(U) \leq x) \stackrel{\text{par (1)}}{=} \mathbb{P}(U \leq F(x)) \stackrel{\text{car } U \sim \mathcal{U}[0,1]}{=} F(x).$$

Preuve de la méthode d'inversion

Montrons d'abord que, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Rightarrow u \leq F(x).$$

Preuve de la méthode d'inversion

Montrons d'abord que, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Rightarrow u \leq F(x).$$

\Rightarrow Soient $u \in]0, 1[$ et $x \in \mathbb{R}$ tels que $F^{-1}(u) \leq x$.

Par croissance de F , on a donc:

$$F\left(F^{-1}(u)\right) \leq F(x)$$

Or, en se souvenant que par définition

$$F^{-1}(u) = \inf \{z \in \mathbb{R} \text{ tel que } F(z) \geq u\},$$

on a donc directement

$$u \leq F\left(F^{-1}(u)\right) \leq F(x)$$

Preuve de la méthode d'inversion

Montrons maintenant, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Leftrightarrow u \leq F(x).$$

Preuve de la méthode d'inversion

Montrons maintenant, pour tout $u \in]0, 1[$

$$\forall x \in \mathbb{R}, F^{-1}(u) \leq x \Leftrightarrow u \leq F(x).$$

\Leftarrow Soient $u \in]0, 1[$ et $x \in \mathbb{R}$ tels que $u \leq F(x)$.

Ainsi, $x \in \{z \in \mathbb{R} \text{ tel que } F(z) \geq u\}$, donc $F^{-1}(u) \leq x$.

Intérêt de la méthode d'inversion

Ainsi, pour une variable aléatoire à valeurs dans \mathbb{R} , on sait simuler si on connaît l'inverse généralisée de sa fonction de répartition.

Méthode d'acceptation rejet

Exemple motivant

On veut simuler selon la densité $f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}} \mathbf{1}_{x \geq 0}$.

La fonction quantile n'a pas d'expression analytique. La méthode d'inversion ne peut être appliquée.

Exemple motivant

On veut simuler selon la densité $f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}} \mathbf{1}_{x \geq 0}$.

La fonction quantile n'a pas d'expression analytique. La méthode d'inversion ne peut être appliquée.

Principe de la méthode d'acceptation rejet:

- ▶ Simuler des *candidats* selon une autre loi qu'on sait simuler.

Exemple motivant

On veut simuler selon la densité $f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}} \mathbf{1}_{x \geq 0}$.

La fonction quantile n'a pas d'expression analytique. La méthode d'inversion ne peut être appliquée.

Principe de la méthode d'acceptation rejet:

- ▶ Simuler des *candidats* selon une autre loi qu'on sait simuler.
- ▶ Choisir parmi les candidats grâce à la loi uniforme.

Méthode d'acceptation rejet (proposition)

Soit f et g deux densités sur \mathbb{R}^d . On suppose qu'il existe une constante M telle que

$$\forall x \in \mathbb{R}^d \quad f(x) \leq Mg(x)$$

On note

$$0 \leq r(x) := \frac{f(x)}{Mg(x)} \leq 1.$$

Méthode d'acceptation rejet (proposition)

Soit f et g deux densités sur \mathbb{R}^d . On suppose qu'il existe une constante M telle que

$$\forall x \in \mathbb{R}^d \quad f(x) \leq Mg(x)$$

On note

$$0 \leq r(x) := \frac{f(x)}{Mg(x)} \leq 1.$$

-Soient $(Y_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de densité g et $(U_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de loi uniforme sur $[0, 1]$.

Méthode d'acceptation rejet (proposition)

Soit f et g deux densités sur \mathbb{R}^d . On suppose qu'il existe une constante M telle que

$$\forall x \in \mathbb{R}^d \quad f(x) \leq Mg(x)$$

On note

$$0 \leq r(x) := \frac{f(x)}{Mg(x)} \leq 1.$$

- Soient $(Y_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de densité g et $(U_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de loi uniforme sur $[0, 1]$. - On note T la variable aléatoire (à valeurs dans \mathbb{N}^*):

$$T = \inf \{n, \text{ tel que } U_n \leq r(Y_n)\}.$$

.

Méthode d'acceptation rejet (proposition)

Soit f et g deux densités sur \mathbb{R}^d . On suppose qu'il existe une constante M telle que

$$\forall x \in \mathbb{R}^d \quad f(x) \leq Mg(x)$$

On note

$$0 \leq r(x) := \frac{f(x)}{Mg(x)} \leq 1.$$

- Soient $(Y_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de densité g et $(U_n)_{n \geq 1}$ une suite de variables aléatoires i.i.d. de loi uniforme sur $[0, 1]$. - On note T la variable aléatoire (à valeurs dans \mathbb{N}^*):

$$T = \inf \{n, \text{ tel que } U_n \leq r(Y_n)\}.$$

. - Alors, la variable aléatoire $X := Y_T$ (T -ième valeur de la suite $(Y_n)_{n \geq 1}$) a pour densité f .

Méthode d'acceptation rejet (algorithme)

On veut tirer un échantillon X de densité f . On ne sait simuler que selon la densité g . On suppose qu'il existe une constante M telle que

$$\forall x \in \mathbb{R}^d \quad f(x) \leq M g(x)$$

Algorithme Condition \leftarrow FALSE

▶ Tant que not Condition:

- ▶ Tirer $Y \sim g(y)$;
- ▶ Tirer (indépendemment) $U \sim \mathcal{U}[0, 1]$;
- ▶ Si

$$U \leq \frac{f(Y)}{Mg(Y)},$$

alors on pose Condition \leftarrow TRUE et $X = Y$

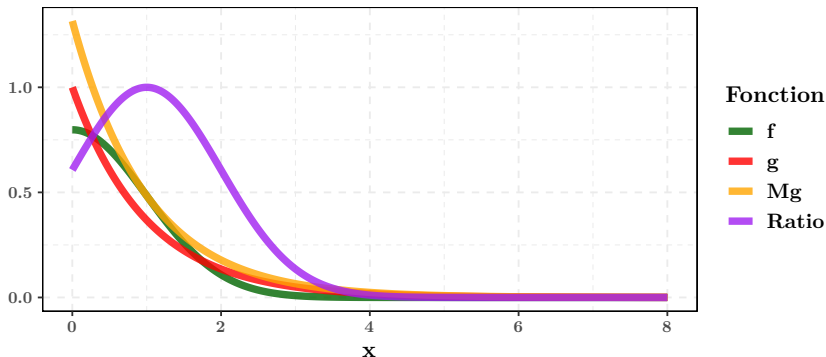
- ▶ Sinon Condition \leftarrow FALSE

En sortie, $X \sim f(x)$.

Méthode d'acceptation rejet: Exemple

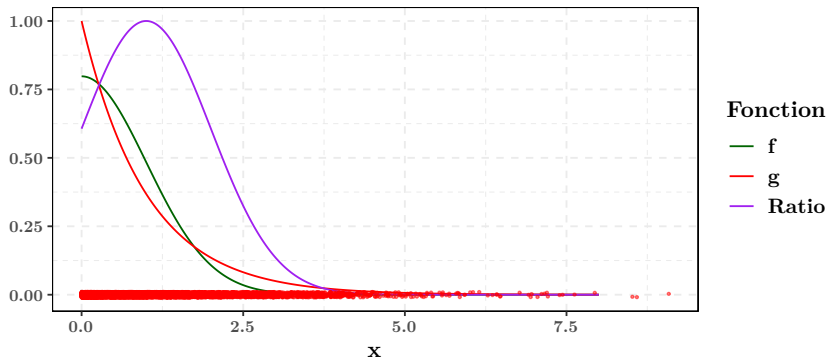
- ▶ On veut simuler selon $f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}} \mathbf{1}_{x \geq 0}$
- ▶ On considère $g(x) = e^{-x} \mathbf{1}_{x \geq 0}$ (densité d'un $\mathcal{Exp}(\lambda = 1)$)
- ▶ On montre que

$$\forall x \in \mathbb{R}, f(x) \leq \overbrace{\sqrt{\frac{2}{\pi}} e^{\frac{1}{2}}}^M g(x)$$



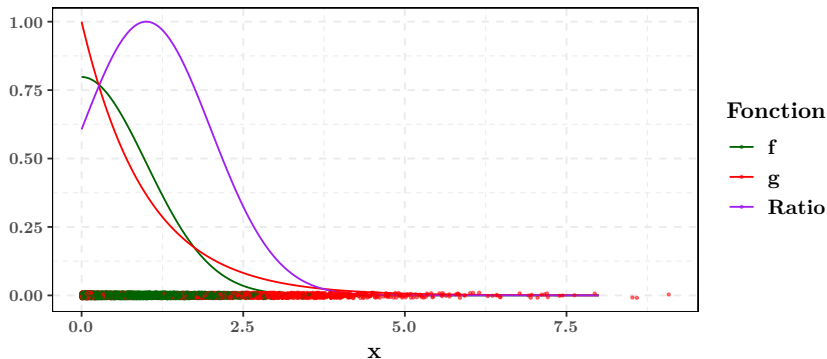
Méthode d'acceptation rejet: Exemple

- On simule 10000 points selon g



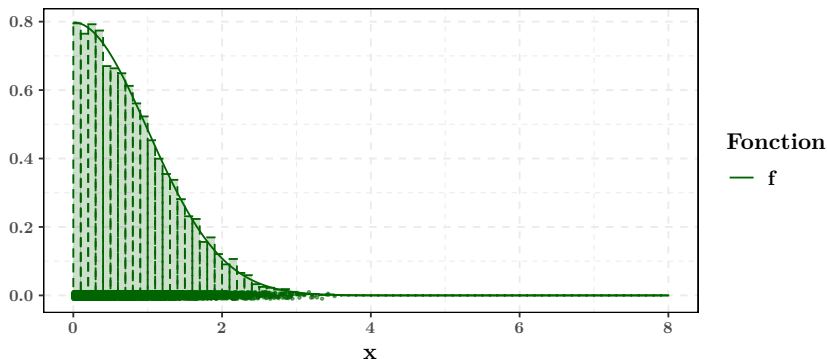
Méthode d'acceptation rejet: Exemple

- ▶ On simule 10000 points selon g
- ▶ On accepte avec une probabilité donnée par le ratio



Méthode d'acceptation rejet: Exemple

- ▶ On simule 10000 points selon g
- ▶ On accepte avec une probabilité donnée par le ratio
- ▶ Les points acceptés sont i.i.d. de densité f .



Remarque

Sur l'exemple précédent, au lieu de faire un “tant que”, on a simulé 10000 points et on n'a retenu que les acceptés.

Remarque

Sur l'exemple précédent, au lieu de faire un “tant que”, on a simulé 10000 points et on n'a retenu que les acceptés.

- ▶ Proportion empirique acceptée: 0.761
- ▶ D'un autre côté, on a $1/M = 0.76$

Preuve de la méthode d'acceptation rejet

Preuve à connaître!

- ▶ Voir le poly de cours.
- ▶ Analogue de la preuve sera demandée en devoir.

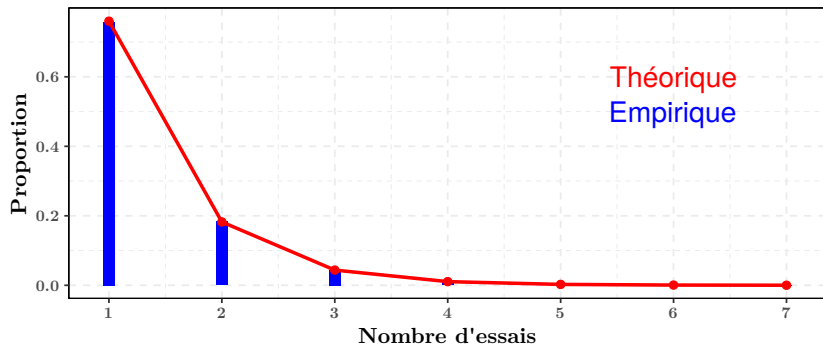
Code R

```
get_one_sample <- function(){  
  condition <- FALSE  
  while(!condition){  
    y <- simulate_g(...) # Simulation selon g  
    u <- runif(1) # Uniform  
    # On suppose que f, g, et M existent  
    condition <- u <= f(y) / (M * g(y))  
  }  
  return(y)  
}
```

Loi du temps d'attente

On s'arrête au premier temps tel qu'une uniforme est inférieure au ratio observée.

La loi du temps d'attente (voir preuve) est une **loi géométrique** sur \mathbb{N}^* de paramètre $\frac{1}{M}$.



Vecteurs aléatoires

Pour simuler un vecteur aléatoire (X, Y) , on pourra utiliser (voir poly et TD pour des exemples):

- ▶ Conditionnement;
- ▶ Changement de variables

Changements de variables pour densité

Soit un couple de variables aléatoires (U, V) de densité $f_{U,V}(u, v)$ définie sur $E_{UV} \subset \mathbb{R}^2$ et un couple de variables aléatoires (X, Y) à valeurs dans $E_{XY} \subset \mathbb{R}^2$. Supposons qu'il existe une application ϕ , C^1 , inversible, et d'inverse C^1 , tel que $(X, Y) = \phi(U, V)$, alors la densité jointe de (X, Y) est donnée par:

$$f_{X,Y}(x, y) = f_{U,V}(\phi^{-1}(x, y)) |\det J_{\phi^{-1}}(x, y)|$$

où J_{ϕ} désigne la matrice jacobienne d'une application $\phi(u, v)$:

$$J_{\phi}(u, v) = \begin{pmatrix} \frac{\delta \phi_1}{\delta u}(u, v) & \frac{\delta \phi_1}{\delta v}(u, v) \\ \frac{\delta \phi_2}{\delta u}(u, v) & \frac{\delta \phi_2}{\delta v}(u, v) \end{pmatrix}$$