MÉTHODES MONTE CARLO

Projet – Adrien Passuello & Axel Sauvaget

2022–2023

☐ Exercice 1

Soient X et Y deux variables aléatoires réelles indépendantes telles que :

- ▷ X suit la loi gamma de densité par rapport à la mesure de Lebesgue sur $\mathbb{R}_+^*$:

$$f(x) = \frac{1}{\Gamma(\alpha)\theta^\alpha} x^{\alpha-1} \exp\left(-\frac{x}{\theta}\right), \quad \text{avec } \alpha = 3 \quad \text{et} \quad \theta = 2$$

- ▷ Y suit la loi de densité par rapport à la mesure de Lebesgue sur $\mathbb{R}$:

$$g(y) = \frac{y-\mu+2b}{4b^2} \mathbb{1}_{\{y\in[\mu-2b,\mu]\}} + \frac{1}{2b} \exp\left(-\frac{y-\mu}{b}\right) \mathbb{1}_{\{y>\mu\}} \quad \text{avec } \mu = -13 \quad \text{et} \quad b = 2$$

L'objectif de cet exercice est d'estimer :

$\delta = \mathbb{P}\left[X + Y > t\right]$, avec $t = 0$.

Graine utilisée : 243

⚠ sur R les résultats peuvent différer alors que la graine est la même

▷ Méthode Monte Carlo n°1

1. (♠) Décrire une méthode pour simuler suivant la loi de (X, Y) et écrire le code R associé. Vous préciserez les différents calculs effectués.

Comme les variables X et Y sont indépendantes, la densité du couple (X, Y) est le produit des densités marginales de X et Y, *i.e.* $d : (x, y) \in \mathbb{R}_+^* \times \mathbb{R} \longmapsto d(x, y) = f(x)g(y)$ est la densité du couple.

Donc pour simuler suivant la loi du couple (X, Y), il suffit de simuler suivant f et g.

Pour simuler n variables aléatoires suivant f, on utilise le générateur aléatoire de la librairie de base de R :
$\longrightarrow$ `rgamma(n,shape = alpha, scale = theta)`.

Pour simuler n variables aléatoires suivant g, on va appliquer la méthode de la fonction inverse.

► **Étape 1 :** Calculons la fonction de répartition de g :

- ▷ Si $y < \mu - 2b$, alors $\boxed{G(y) = 0}$.

- ▷ Si $y \in [\mu - 2b, \mu]$, alors $G(y) = \int_{-\infty}^{y} g(x)\, dx = \int_{\mu-2b}^{y} \frac{x}{4b^2} - \frac{\mu-2b}{4b^2}\, dx = \frac{1}{4b^2}\left[\frac{x^2}{2}\right]_{\mu-2b}^{y} - \frac{\mu-2b}{4b^2}\left[x\right]_{\mu-2b}^{y}$

$$= \frac{1}{8b^2}\left[y^2 - 2y(\mu-2b) + (\mu-2b)^2\right] = \boxed{\frac{1}{8b^2}\Big(y - (\mu-2b)\Big)^2}$$

▷ Si $y > \mu$, alors $G(y) = \displaystyle\int_{-\infty}^{y} g(x)\,dx = \int_{\mu-2b}^{\mu} \frac{x}{4b^2} - \frac{\mu-2b}{4b^2}\,dx + \int_{\mu}^{y} \frac{1}{2b}\exp\left(-\frac{x-\mu}{b}\right)dx$

$$= \frac{1}{8b^2}\Big(\mu-(\mu-2b)\Big)^2 + \frac{1}{2b}\left[-b\exp\left(-\frac{x-\mu}{b}\right)\right]_{\mu}^{y} = \frac{1}{2} - \frac{1}{2}\exp\left(-\frac{y-\mu}{b}\right) + \frac{1}{2}$$

$$= \boxed{1 - \frac{1}{2}\exp\left(-\frac{y-\mu}{b}\right)}$$

Donc pour tout $y \in \mathbb{R}$, $G(y) = \begin{cases} 0 & \text{si } y < \mu - 2b \\ \dfrac{1}{8b^2}\Big(y-(\mu-2b)\Big)^2 & \text{si } y \in [\mu-2b, \mu] \\ 1 - \dfrac{1}{2}\exp\left(-\dfrac{y-\mu}{b}\right) & \text{si } y > \mu \end{cases}$

i.e pour tout $y \in \mathbb{R}$, $\boxed{G(y) = \dfrac{1}{8b^2}\Big(y-(\mu-2b)\Big)^2 \mathbb{1}_{\{y\in[\mu-2b,\mu]\}} + \left(1-\dfrac{1}{2}\exp\left(-\dfrac{y-\mu}{b}\right)\right)\mathbb{1}_{\{y>\mu\}}}$

► **Étape 2 :** Calculons l'inverse généralisée de G, notée $G^{\leftarrow}$:

▷ Si $u \in \left[0, \frac{1}{2}\right]$, alors comme G est bijective sur $[\mu-2b, \mu]$, on a pour tout $y \in [\mu-2b, \mu]$:

$$u = G(y) \iff u = \frac{1}{8b^2}\Big(y-(\mu-2b)\Big)^2 \iff \sqrt{8b^2u} = \Big|y-(\mu-2b)\Big|$$

$$\overset{\text{car } y\geq\mu-2b}{\iff} 2b\sqrt{2u} = y-(\mu-2b) \iff 2b\sqrt{2u}+\mu-2b = y \iff \boxed{y = 2b\big(\sqrt{2u}-1\big)+\mu}$$

▷ Si $u \in \left]\frac{1}{2}, 1\right[$, alors comme G est bijective sur $]\mu, +\infty[$, on a pour tout $y > \mu$:

$$u = G(y) \iff u = 1-\frac{1}{2}\exp\left(-\frac{y-\mu}{b}\right) \iff 2(1-u) = \exp\left(-\frac{y-\mu}{b}\right)$$

$$\iff \ln\big(2(1-u)\big) = -\frac{y-\mu}{b} \iff -b\ln\big(2(1-u)\big) = y-\mu \iff \boxed{y = \mu - b\ln\big(2(1-u)\big)}$$

Donc pour tout $u \in [0,1[$,

$G^{\leftarrow}(u) = \begin{cases} 2b\big(\sqrt{2u}-1\big)+\mu & \text{si } u \in \left[0, \frac{1}{2}\right] \\ \mu - b\ln\big(2(1-u)\big) & \text{si } u \in \left]\frac{1}{2}, 1\right[\end{cases}$ (NB : si $u \to 1$, on considère que $\ln(1-u) \to +\infty$)

i.e. pour tout $u \in [0,1[$, $\boxed{G^{\leftarrow}(u) = \Big(2b\big(\sqrt{2u}-1\big)+\mu\Big)\mathbb{1}_{\{u\in[0,\frac{1}{2}]\}} + \Big(\mu - b\ln\big(2(1-u)\big)\Big)\mathbb{1}_{\{u\in]\frac{1}{2},1[\}}}$

Algorithme de l'inverse généralisée :

Pour simuler n variables aléatoires suivant la densité g, on répète n fois le schéma suivant :

▷ on simule $U \sim \mathcal{U}\big([0,1]\big)$

▷ on calcule $G^{\leftarrow}(U) \sim g$

```r
                        #### Algorithme de l'inverse généralisé ####
set.seed(243) # on définit une graine (attention sur R les résultats peuvent différer
# alors que la graine est la même)
# Densité f d'une loi gamma de paramètre de forme alpha = 3 et d'échelle theta = 2
densite_f <- function(x,alpha = 3,theta = 2){
  return(dgamma(x, shape = alpha, scale = theta))
}

# Densité g de paramètre mu = -13 et b = 2
densite_g <- function(y, mu = -13, b = 2){
  return((y-mu+2*b)/(4*b*b)*(y>= mu-2*b)*(y<=mu)+exp(-(y-mu)/b)/(2*b)*(y>mu))
}

# Fonction de répartition de G calculée théoriquement
fdr_G <- function(y,mu = -13, b = 2){
  return((y-(mu-2*b))^2/(8*b*b)*(mu-2*b<=mu)*(y <=mu)+(1-exp(-(y-mu)/b)/2)*(y>mu))
}

# Fonction inverse généralisée de G calculée théoriquement
G_fleche <- function(u,mu = -13, b = 2){
  return((2*b*(sqrt(2*u)-1)+mu)*(0<=u)*(u<=1/2)+(mu-b*log(2*(1-u)))*(u>1/2)*(u<1))
}

# Simulation de n v.a. de densité g (via la la méthode de la fonction inverse)
rgen_g <- function(n, mu = -13, b = 2){
  # On simule n v.a. uniforme sur [0,1]
  U <- runif(n)
  # On retourne G_flèche(U) car si U ~ U([0,1]), alors G_flèche(U) et Y sont de même loi
  return(G_fleche(U))
}

# Simulation de n v.a. de densité f (via la fonction de R rgamma)
rgen_f <- function(n, alpha = 3, theta = 2){
  return(rgamma(n, shape = alpha, scale = theta))
}
```

2. (♠) Vérifier graphiquement que votre méthode et votre code sont corrects.

```r
                              #### Vérification graphique ####
# Package de couleurs
library(viridisLite)
c_ <- plasma(5)

# On va simuler 1000 v.a. suivant f et g
n <- 1000

# Affichage de 2 graphiques
par(mfrow = c(1,2))
```

```r
# Application de l'algorithme de la fonction inverse
g <- rgen_g(n)
# histogramme de l'échantillon de v.a. suivant g
hist(g, freq = FALSE, main = "Distribution de l'échantillon", ylab = "Fréquence",
     xlab = "y", ylim = c(0,0.30), breaks = 40,col = "grey70", border = "grey40")
# Superposition de la courbe de la densité g
lines(x <- seq(min(g),max(g), 0.001), densite_g(x), col = c_[4], lwd = 3)
# Légende
legend("topright", "Densité g", col = c_[4], lwd = 3, box.lty = 1, inset = c(0.01,0.01),
       bg="Grey95", box.col = "Grey50")

# Application de la fonction rgamma
f <- rgen_f(n)
# histogramme de l'échantillon de v.a. suivant f
hist(f, freq = FALSE, main = "Distribution de l'échantillon", ylab = "Fréquence",
     xlab = "x", ylim = c(0,0.30), breaks = 40,col = "grey70", border = "grey40")
# Superposition de la courbe de la densité f
lines(x <- seq(min(f),max(f), 0.001), densite_f(x), col = c_[3], lwd = 3)
# Légende
legend("topright", "Densité f", col = c_[3], lwd = 3, box.lty = 1, inset = c(0.01,0.01),
       bg="Grey95", box.col = "Grey50")
```

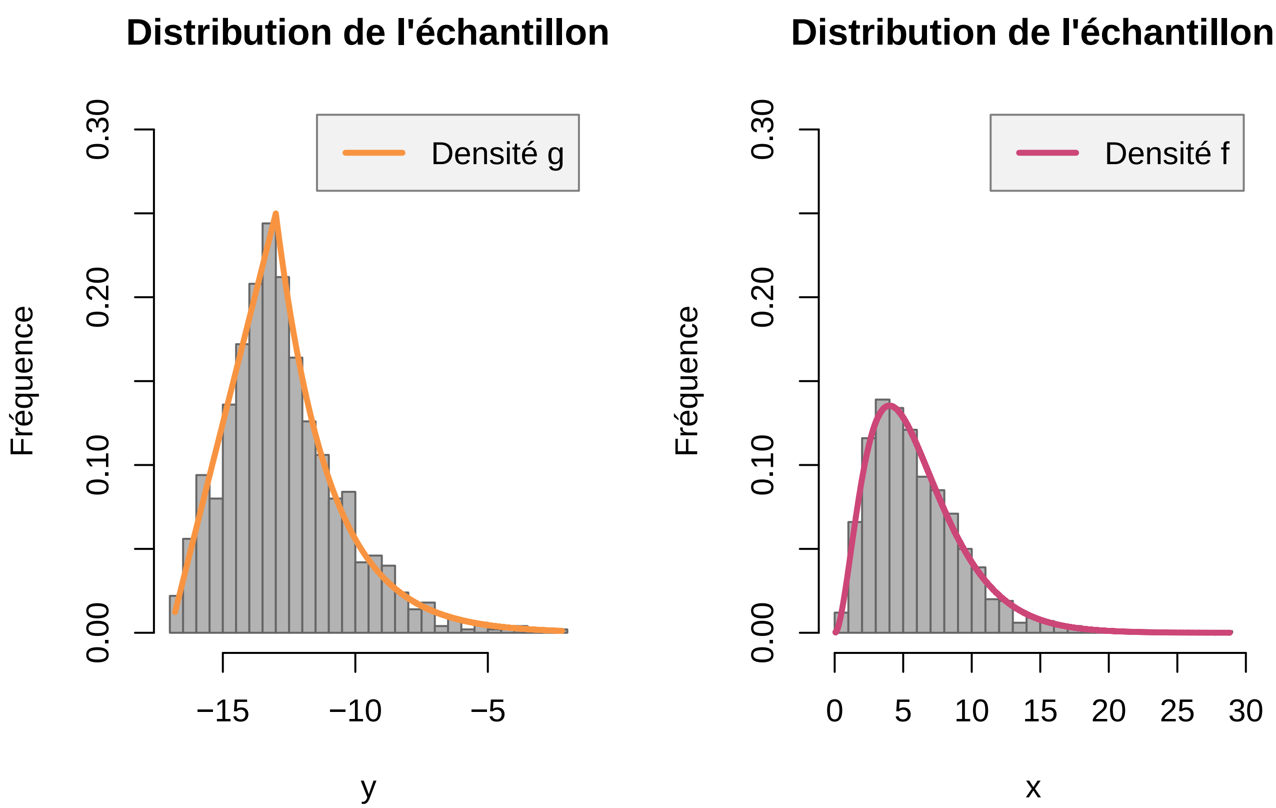

FIGURE 1 – Vérification graphique de bonne simulation des densités f et g pour $n = 1000$

On peut voir que les histogrammes des échantillons et les courbes des densités f et g sont bien en adéquation. On a donc bien simulé suivant f et g.

3. Écrire l'estimateur Monte Carlo $\widehat{p}_n$ de δ correspondant à une suite de n variables aléatoires $(X_k, Y_k)_{k\geq 1}$ i.i.d. suivant la loi de (X, Y).

Soit $t = 0$, on a $\delta = \mathbb{P}\left[X + Y > t\right] = \mathbb{E}\left[\mathbb{1}_{\{X+Y>t\}}\right]$

Donc en posant $h : (x, y) \mapsto \mathbb{1}_{\{x+y>t\}}$, on cherche à estimer $\delta = \mathbb{E}\left[h(X, Y)\right]$.

On pose donc l'estimateur Monte Carlo $\widehat{p}_n$ de δ suivant :

$$\boxed{\widehat{p}_n = \frac{1}{n}\sum_{k=1}^{n} \mathbb{1}_{\{X_k+Y_k>t\}}} \quad \text{avec } (X_k, Y_k)_{k\geq 1} \text{ i.i.d. suivant la loi de } (X, Y)$$

4. (♠) Implémenter un code `R` qui fournit, pour $n = 1000$, une estimation de δ, de l'erreur quadratique moyenne de $\widehat{p}_n$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

Nota Bene :

Pour les questions **(4.)**, **(7.)**, **(10.)**, **(12.)**, **(17.)** et **(20.)**, on fera appel à la fonction suivante pour renvoyer pour chaque estimation de δ, un dataframe regroupant l'estimateur, son erreur quadratique moyenne et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```
# Fonction qui prend une fonction h et un niveau de confiance en argument
# et renvoie un tableau contenant l'estimateur, son erreur quadratique moyenne
# et un intervalle de confiance asymptotique bilatéral pour delta au niveau 1-alpha
# Par défaut on pose alpha = 5% pour avoir un intervalle de confiance au niveau 95%
MC <- function(h, alpha = 0.05){
  hbar <- mean(h) # estimateur
  var <- var(h) # erreur quadratique moyenne (variance de l'estimateur)
  IC1 <- round(mean(h)-qnorm(1-alpha/2)*sqrt(var(h)/n),8)# borne inférieure de l'IC
  IC2 <- round(mean(h)+qnorm(1-alpha/2)*sqrt(var(h)/n),8)# borne supérieure de l'IC
  # création d'un data.frame (tableau)
  d1 <- data.frame(c("", "" ,IC1)) # première colonne avec uniquement IC1
  d2 <- data.frame(c(hbar, var ,IC2)) # seconde colonne avec hbar, var et IC2
  D <- cbind(d1,d2) # fusion des colonnes
  # renommage des lignes
  rownames(D) <- c("Estimateur", "Erreur quadratique moyenne",
                   "Intervalle de confiance")
  # renommage des colonnes
  names(D) <- c("", "")
  return(D)
}
```

```
                    #### Estimation de delta n°1 ####
p_n <- function(n, t = 0){
  X <- rgen_f(n) # On simule X suivant f
```

```
  Y <- rgen_g(n) # On simule X suivant f
  h <- 1*(X+Y>t) # On calcule h définie par l'indicatrice telle que X+Y>t
  return(h) # on renvoit la fonction h
}

#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(p_n(n))

##
## Estimateur                                  0.07000000
## Erreur quadratique moyenne                  0.06516517
## Intervalle de confiance    0.0541782 0.08582180
```

L'estimation de δ par l'estimateur Monte Carlo classique $\widehat{p}_n$ donne un intervalle de confiance trop grand et une estimation trop volatile quand on fait plusieurs itérations. On va améliorer cette approximation par la suite.

5. (♠) Vérifier empiriquement si l'hypothèse du régime asymptotique $\widehat{p}_n$ est satisfaite.

Vérifier si l'hypothèse du régime asymptotique de $\widehat{p}_n$ est satisfaite consiste à vérifier si :

$$\sqrt{\frac{n}{\sigma^2}}\left(\widehat{p}_n-\delta\right)\xrightarrow[n\to+\infty]{\mathscr{L}}\mathcal{N}(0,1)$$

Or σ^2 est inconnu, mais on a $\widehat{\sigma}_n^2 \xrightarrow[n\to+\infty]{} \sigma^2$, donc par le théorème de Slutsky, il suffit de vérifier que :

$$\sqrt{\frac{n}{\widehat{\sigma}_n^2}}\left(\widehat{p}_n-\delta\right)\xrightarrow[n\to+\infty]{\mathscr{L}}\mathcal{N}(0,1) \text{ où } \widehat{\sigma}_n^2 \text{ est l'erreur quadratique moyenne de } \widehat{p}_n$$

```
      #### Vérification empirique de l'hypothèse de régime asymptotique de p_n ####
n <- 1000 # prenons n = 1000
# p_hat sera une vecteur qui sera rempli de n simulations de p_n
p_hat <- rep(0,n) # on part de p_hat un vecteur rempli de 0
for (i in 1:n){ # pour i variant de 1 à n, on modifie les n éléments de p_hat
  p_hat[i] <- mean(p_n(n)) # le i-ème élément de p_hat est modifié par p_n(n)
}
# On applique la formule du TCL : (p_n - E[p_1])/sqrt(V(p_1))
TCL <- (p_hat - mean(p_hat))/sqrt(var(p_hat))

# Affichage d'un seul graphique
par(mfrow = c(1,1))
# Histogramme de l'échantillon nommé TCL
hist(TCL, freq = FALSE, main = "Régime asymptotique", ylab = "Fréquence",xlab = "x",
     ylim = c(0,0.50), xlim = c(-3,3), breaks = 20,col = "grey70", border = "grey40")
# Superposition de la courbe de la densité d'une loi normale centrée et réduite
lines(x<-seq(-3,3,0.01),dnorm(x),col = c_[2], lwd = 3)
# Légende
legend("topright", "Densité d'une N(0,1)", col = c_[2], lwd = 3, box.lty = 1,
       inset = c(0.01,0.01), bg="Grey95", box.col = "Grey50")
```

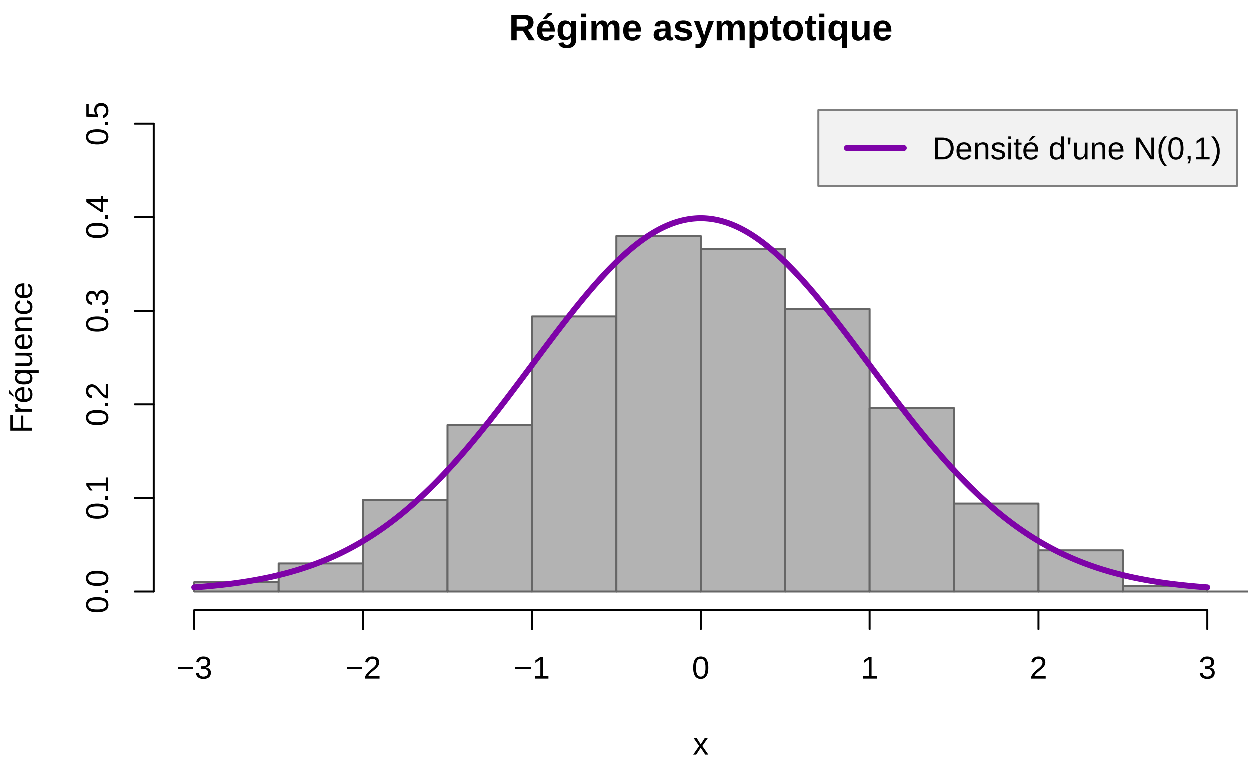

FIGURE 2 – Vérification empirique de l'hypothèse de régime asymptotique

On peut voir que la simulation est en accord avec la courbe gaussienne donc l'hypothèse de régime asymptotique de $\widehat{p}_n$ semble être vérifiée.

6. On note F la fonction de répartition de X. Montrer que $\delta = \mathbb{E}\left[1 - F(t-Y)\right]$ et en déduire un estimateur $\widehat{\delta}_n$ de δ correspondant à une suite de n variables aléatoires $\left(U_k\right)_{k\geq 1}$ i.i.d. suivant la loi uniforme sur $[0,1]$.

On sait que $\delta = \mathbb{P}\left[X+Y>t\right] = 1 - \mathbb{P}\left[X+Y\leq t\right] = 1 - \mathbb{E}\left[\mathbb{1}_{\{X+Y\leq t\}}\right] = 1 - \mathbb{E}\left[\mathbb{1}_{\{X\leq t-Y\}}\right]$

D'où $\delta = 1 - \int_{\mathbb{R}}\int_{\mathbb{R}_+^*} \mathbb{1}_{\{x\leq t-y\}} d(x,y)\, dxdy$ où $d(x,y) = f(x)g(y)$ est la densité du couple (X,Y).

$$
\begin{aligned}
&= 1 - \int_{\mathbb{R}}\int_{\mathbb{R}_+^*} \mathbb{1}_{\{x\leq t-y\}} f(x)g(y)\, dxdy \\
&= 1 - \int_{\mathbb{R}} g(y) \underbrace{\left(\int_{\mathbb{R}_+^*} \mathbb{1}_{\{x\leq t-y\}} f(x)\, dx\right)}_{=\mathbb{E}\left[\mathbb{1}_{X\leq t-y}\right]} dy \\
&= 1 - \int_{\mathbb{R}} g(y)\mathbb{E}\left[\mathbb{1}_{X\leq t-y}\right] dy \\
&= 1 - \int_{\mathbb{R}} g(y)\mathbb{P}\left(X\leq t-y\right) dy \\
&= 1 - \int_{\mathbb{R}} g(y)F(t-y)\, dy
\end{aligned}
$$

$$= 1 - \mathbb{E}\big[F(t-Y)\big]$$

Donc, par linéarité de l'espérance, on a $\boxed{\delta = \mathbb{E}\big[1 - F(t-Y)\big]}$.

On en déduit donc qu'en posant $\widetilde{h} : y \mapsto 1 - F(t-y)$, on cherche à estimer $\delta = \mathbb{E}\Big[\widetilde{h}(Y)\Big]$.

Or on sait que si une variable aléatoire $U \sim \mathscr{U}\big([0,1]\big)$, alors $G^{\leftarrow}(U) \overset{\mathscr{L}}{=} Y$, *i.e.* $G^{\leftarrow}(U)$ et Y sont de même loi.

Donc $\delta = \mathbb{E}\Big[\widetilde{h}\big(G^{\leftarrow}(U)\big)\Big]$ avec $U \sim \mathscr{U}\big([0,1]\big)$.

On pose donc l'estimateur Monte Carlo $\widehat{\delta}_n$ de δ suivant :

$$\boxed{\widehat{\delta}_n = \frac{1}{n}\sum_{k=1}^{n}\Big(1 - F\big(t - G^{\leftarrow}(U_k)\big)\Big)} \quad \text{avec } \big(U_k\big)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}\big([0,1]\big)$$

7. **(♠)** Implémenter un code R qui fournit, pour $n = 1000$, l'estimation $\widehat{\delta}_n$, l'erreur quadratique moyenne de $\widehat{\delta}_n$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```r
                        #### Estimation de delta n°2 ####

# Fonction de répartition F d'une loi gamma de paramètre de forme alpha = 3
# et d'échelle theta = 2
fdr_F <- function(x, alpha = 3, theta = 2){
  return(pgamma(x, shape = alpha, scale = theta))
}
# Estimation de delta_n
delta_n <- function(n, t = 0){
  Y <- rgen_g(n) # On simule n v.a. suivant g
  h <- 1 - fdr_F(t - Y) # On calcule h_tilde définie par 1 - F(t - Y)
  return(h) # on renvoit la fonction h_tilde
}
#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(delta_n(n))

##
## Estimateur                                  0.069121398
## Erreur quadratique moyenne                  0.007165967
## Intervalle de confiance    0.06387471 0.074368090
```

Cette nouvelle estimation de δ par l'estimateur $\widehat{\delta}_n$ donne une variance et un intervalle de confiance plus petits qu'avec l'estimateur $\widehat{p}_n$. On a donc amélioré l'estimation de δ.

8. Selon ces résultats, l'estimateur $\widehat{\delta}_n$ est-il plus performant en terme de variance que l'estimateur $\widehat{p}_n$? Si oui, montrer qu'on a toujours $\mathbb{V}\text{ar}\big[\widehat{\delta}_n\big] \leq \mathbb{V}\text{ar}\big[\widehat{p}_n\big]$.

D'après les résultats donnés par `R`, $\widehat{\delta}_n$ est plus performant en terme de variance que l'estimateur $\widehat{p}_n$, montrons-le théoriquement :

▸ **D'une part :** $\mathbb{V}\text{ar}\big[\widehat{\delta}_n\big] = \mathbb{V}\text{ar}\left[\frac{1}{n}\sum_{k=1}^{n}\Big(1 - F\big(t - G^{\leftarrow}(U_k)\big)\Big)\right]$ avec $(U_k)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}\big([0,1]\big)$

$$= \frac{1}{n^2}\sum_{k=1}^{n}\mathbb{V}\text{ar}\Big[1 - F\big(t - G^{\leftarrow}(U_k)\big)\Big] \quad \text{par indépendance des } (U_k)_{k\geq 1}$$

$$= \frac{1}{n}\mathbb{V}\text{ar}\Big[F\big(t - G^{\leftarrow}(U_1)\big)\Big] \quad \text{car les } (U_k)_{k\geq 1} \text{ sont identiquement distribuées}$$

Donc $\boxed{\mathbb{V}\text{ar}\big[\widehat{\delta}_n\big] = \frac{1}{n}\mathbb{V}\text{ar}\Big[F\big(t - Y\big)\Big]}$ car $G^{\leftarrow}(U_1) \overset{\mathscr{L}}{=} Y$.

▸ **D'autre part :** $\mathbb{V}\text{ar}\big[\widehat{p}_n\big] = \mathbb{V}\text{ar}\left[\frac{1}{n}\sum_{k=1}^{n}\mathbb{1}_{\{X_k + Y_k > t\}}\right]$ avec $(X_k, Y_k)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} (X, Y)$

$$= \frac{1}{n^2}\sum_{k=1}^{n}\mathbb{V}\text{ar}\Big[\mathbb{1}_{\{X_k + Y_k > t\}}\Big] \quad \text{par indépendance des } (X_k, Y_k)_{k\geq 1}$$

Donc $\boxed{\mathbb{V}\text{ar}\big[\widehat{p}_n\big] = \frac{1}{n}\mathbb{V}\text{ar}\Big[\mathbb{1}_{\{X+Y>t\}}\Big]}$ car les $(X_k, Y_k)_{k\geq 1}$ sont identiquement distribuées suivant la loi de (X, Y).

Or $\mathbb{V}\text{ar}\big[\widehat{p}_n\big] = \frac{1}{n}\mathbb{V}\text{ar}\Big[\mathbb{1}_{\{X+Y>t\}}\Big]$

$$= \frac{1}{n}\left(\mathbb{E}\Big[\mathbb{1}^2_{\{X+Y>t\}}\Big] - \left(\mathbb{E}\Big[\mathbb{1}_{\{X+Y>t\}}\Big]\right)^2\right) \quad \text{par définition de la variance}$$

$$= \frac{1}{n}\left(\mathbb{E}\Big[\mathbb{1}_{\{X+Y>t\}}\Big] - \left(\mathbb{E}\Big[\mathbb{1}_{\{X+Y>t\}}\Big]\right)^2\right) \quad \text{car } \mathbb{1}^2_{\{X+Y>t\}} = \mathbb{1}_{\{X+Y>t\}}\big(\longrightarrow \text{une indicatrice ne peut prendre que ses valeurs dans } \{0,1\}\big).$$

$$= \frac{1}{n}\left(\underbrace{\mathbb{P}\big[X + Y > t\big]}_{=\delta} - \Big(\underbrace{\mathbb{P}\big[X + Y > t\big]}_{=\delta}\Big)^2\right) \quad \text{en effet, d'après l'énoncé } \delta = \mathbb{P}\big[X + Y > t\big]$$

$$= \frac{1}{n}\Big(\mathbb{E}\big[1 - F(t - Y)\big] - \Big(\mathbb{E}\big[1 - F(t - Y)\big]\Big)^2\Big) \quad \text{d'après la question } \textbf{(6.)}, \ \delta = \mathbb{E}\big[1 - F(t - Y)\big]$$

$$= \frac{1}{n}\left(\mathbb{E}\big[1 - F(t - Y)\big] + \left(\mathbb{V}\text{ar}\Big[1 - F\big(t - Y\big)\Big] - \mathbb{E}\Big[\big(1 - F(t - Y)\big)^2\Big]\right)\right) \quad \text{car } \mathbb{V}\text{ar}[Z] - \mathbb{E}[Z^2] = -\big(\mathbb{E}[Z]\big)^2 \text{ pour toute variable aléatoire } Z$$

$$= \underbrace{\frac{1}{n}\mathbb{V}\text{ar}\Big[F\big(t - Y\big)\Big]}_{=\mathbb{V}\text{ar}\left[\widehat{\delta}_n\right]} + \frac{1}{n}\Big(1 - \mathbb{E}\big[F(t - Y)\big] - \mathbb{E}\Big[1 - 2F(t - Y) + F^2(t - Y)\Big]\Big)$$

$$= \mathbb{V}\text{ar}\Big[\widehat{\delta}_n\Big] + \frac{1}{n}\Big(\mathbb{E}\big[F(t - Y)\big] - \mathbb{E}\Big[F^2(t - Y)\Big]\Big)$$

$$= \mathbb{V}\text{ar}\Big[\widehat{\delta}_n\Big] + \frac{1}{n}\Big(\mathbb{E}\big[F(t - Y) - F^2(t - Y)\big]\Big) \quad \text{par linéarité de l'espérance}$$

$$= \mathbb{V}\text{ar}\left[\widehat{\delta}_n\right] + \frac{1}{n}\Bigg(\underbrace{\mathbb{E}\Big[\underbrace{F(t-Y)}_{\geq 0}\big(\underbrace{1-F(t-Y)}_{\geq 0 \text{ car } F(t-Y)\leq 1}\big)\Big]}_{\geq 0}\Bigg)$$

$\hookrightarrow$ car l'espérance d'une variable aléatoire positive est positive

Donc $\boxed{\mathbb{V}\text{ar}[\widehat{p}_n] \geq \mathbb{V}\text{ar}[\widehat{\delta}_n]}$

▷ Méthode Monte Carlo n°2

9. En utilisant la méthode de la variable antithétique, proposer un estimateur $\widehat{\delta}_n^A$ de δ. Montrer que $\mathbb{V}\text{ar}[\widehat{\delta}_n^A] \leq \mathbb{V}\text{ar}[\widehat{\delta}_n]$.

On a vu à la question **(6.)** que $\widetilde{h}(Y) = \widetilde{h}\big(G^{\leftarrow}(U)\big)$ car $Y \overset{\mathscr{L}}{=} G^{\leftarrow}(U)$ avec $U \sim \mathscr{U}\big([0,1]\big)$

Or si $U \sim \mathscr{U}\big([0,1]\big)$ alors $1-U \sim \mathscr{U}\big([0,1]\big)$.

Donc on pose $A : x \mapsto 1-x$ et on a :

$$\widetilde{h} \circ G^{\leftarrow} \circ A(U) = \widetilde{h}\big(G^{\leftarrow}(1-U)\big) = \widetilde{h}(Y) \quad \text{car} \quad Y \overset{\mathscr{L}}{=} G^{\leftarrow}(U) \overset{\mathscr{L}}{=} G^{\leftarrow}(1-U)$$

Donc on pose $\widehat{\delta}_n^A = \frac{1}{n}\sum_{k=1}^{n} \frac{\widetilde{h}\big(G^{\leftarrow}(U_k)\big) + \widetilde{h}\big(G^{\leftarrow}(1-U_k)\big)}{2}$

$$= \frac{1}{n}\sum_{k=1}^{n} \frac{1-F\big(t-G^{\leftarrow}(U_k)\big) + 1 - F\big(t-G^{\leftarrow}(1-U_k)\big)}{2}$$

$$\boxed{\widehat{\delta}_n^A = \frac{1}{n}\sum_{k=1}^{n}\left(1 - \frac{F\big(t-G^{\leftarrow}(U_k)\big) + F\big(t-G^{\leftarrow}(1-U_k)\big)}{2}\right)} \quad \text{avec } \big(U_k\big)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}\big([0,1]\big)$$

Montrons que $\mathbb{V}\text{ar}[\widehat{\delta}_n^A] \leq \mathbb{V}\text{ar}[\widehat{\delta}_n]$:

On a $\mathbb{V}\text{ar}[\widehat{\delta}_n^A] = \mathbb{V}\text{ar}\left[\frac{1}{n}\sum_{k=1}^{n}\left(1 - \frac{F\big(t-G^{\leftarrow}(U_k)\big) + F\big(t-G^{\leftarrow}(1-U_k)\big)}{2}\right)\right]$

$$= \frac{1}{n^2}\sum_{k=1}^{n} \mathbb{V}\text{ar}\left[1 - \frac{F\big(t-G^{\leftarrow}(U_k)\big) + F\big(t-G^{\leftarrow}(1-U_k)\big)}{2}\right]$$

$$= \frac{1}{4n^2}\sum_{k=1}^{n} \mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(U_k)\big)\Big] + \mathbb{V}\text{ar}\Big[F\big(G^{\leftarrow}(1-U_k)\big)\Big] + 2\text{Cov}\Big(F\big(t-G^{\leftarrow}(U_k)\big), F\big(t-G^{\leftarrow}(1-U_k)\big)\Big)$$

Or les $\big(U_k\big)_{k1}$ sont identiquement distribuées de loi uniforme sur $[0,1]$.

$$\Longrightarrow \mathbb{V}\text{ar}[\widehat{\delta}_n^A] = \frac{1}{4n}\Big(\mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(U)\big)\Big] + \mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(1-U)\big)\Big] + 2\text{Cov}\Big(F\big(t-G^{\leftarrow}(U)\big), F\big(t-G^{\leftarrow}(1-U)\big)\Big)\Big)$$

Or $G^{\leftarrow}(U) \overset{\mathscr{L}}{=} G^{\leftarrow}(1-U)$

$$\Longrightarrow \mathbb{V}\text{ar}[\widehat{\delta}_n^A] = \frac{1}{2n}\Big(\mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(U)\big)\Big] + \text{Cov}\Big(F\big(t-G^{\leftarrow}(U)\big), F\big(t-G^{\leftarrow}(1-U)\big)\Big)\Big)$$

Donc $\mathbb{V}\text{ar}\big[\widehat{\delta}_n^A\big] = \underbrace{\frac{1}{2n}\mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(U)\big)\Big]}_{=\frac{1}{2}\mathbb{V}\text{ar}[\widehat{\delta}_n]}(1+\rho)$ avec $\rho := \dfrac{\text{Cov}\Big(F\big(t-G^{\leftarrow}(U)\big), F\big(t-G^{\leftarrow}(1-U)\big)\Big)}{\mathbb{V}\text{ar}\Big[F\big(t-G^{\leftarrow}(U)\big)\Big]}$

Or la transformation $A : x \mapsto 1-x$ est décroissante en laissant la loi uniforme sur $[0,1]$ invariante et la fonction $\widetilde{h} : y \mapsto 1-F(t-y)$ est monotone.
Alors, d'après la *proposition 3.5* du cours, $\text{Cov}\Big(F\big(t-G^{\leftarrow}(U)\big), F\big(t-G^{\leftarrow}(1-U)\big)\Big) \leq 0$.

Donc $(1+\rho) \leq 1$ ce qui implique que $\boxed{\mathbb{V}\text{ar}\big[\widehat{\delta}_n^A\big] \leq \mathbb{V}\text{ar}\big[\widehat{\delta}_n\big]}$.

10. (♠) Implémenter un code R qui fournit, pour $n = 1000$, l'estimation $\widehat{\delta}_n^A$, l'erreur quadratique moyenne de $\widehat{\delta}_n^A$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```
                              #### Estimation de delta n°3 ####
# Estimation de delta_A_n (méthode de la variable antithétique)
delta_A_n <- function(n, t = 0){
  U <- runif(n) # on simule n v.a. uniforme
  # on calcule h_tilde(G_fleche(U))
  h <- 1- fdr_F(t-G_fleche(U))
  # on calcule h_tilde(G_fleche(A(U)) avec A : x |-> 1-x
  h_A <- 1 - fdr_F(t-G_fleche(1-U))
  # on fait la moyenne entre h_tilde(G_fleche(U)) et h_tilde(G_fleche(A(U)))
  H <- (h + h_A)/2
  return(H) # on renvoie (h•G_fleche(U)+h•G_fleche•A(U))/2
}
#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(delta_A_n(n))

##
## Estimateur                                  0.074199620
## Erreur quadratique moyenne                  0.004154136
## Intervalle de confiance      0.07020488 0.078194360
```

Cette nouvelle estimation de δ par l'estimateur $\widehat{\delta}_n^A$ donne une variance et un intervalle de confiance plus petits qu'avec les deux estimateurs précédents $\widehat{p}_n$ et $\widehat{\delta}_n$. On a donc encore amélioré l'estimation de δ.

▷ Méthode Monte Carlo n°3

On note q_ε le quantile d'ordre ε de la loi de X et on définit les fonctions :

$$h_{0,1} : \begin{array}{l} \mathbb{R} \to \mathbb{R} \\ y \mapsto y \end{array} \quad \text{et} \quad h_{0,2} : \begin{array}{l} \mathbb{R} \to \mathbb{R} \\ y \mapsto \mathbb{1}_{\{y \geq t-q_\varepsilon\}} \end{array}$$

Dans la suite on prend $\varepsilon = 0,6$.

11. (♠) Entre $h_{0,1}$ et $h_{0,2}$, quelle fonction utiliseriez-vous pour réduire la variance de $\widehat{\delta}_n$ à l'aide de la méthode de la variable de contrôle ? Écrire l'estimateur $\widehat{\delta}_n^{\text{cont}}$ associé.

D'après la *proposition 3.6* du cours, on sait que la fonction qu'il faut utiliser pour réduire la variance de $\widehat{\delta}_n$ à l'aide de la méthode de la variable de contrôle est la fonction $h_{0,i}$ telle que la grandeur $\rho\big(\widetilde{h}, h_{0,i}\big)$ soit la plus proche de 1,

avec $\rho\big(\widetilde{h}, h_{0,i}\big) = \dfrac{\operatorname{Cov}\big(\widetilde{h}(Y), h_{0,i}(Y)\big)}{\sqrt{\mathbb{V}\mathrm{ar}\big[\widetilde{h}(Y)\big]\mathbb{V}\mathrm{ar}\big[h_{0,i}(Y)\big]}}$ où Y est une variable aléatoire de densité g et pour $i = 1, 2$.

Dans ce cas, l'estimateur de variance minimale $\widehat{\delta}_n^{\text{cont}}(b)$ sera obtenu pour $b^\star = \dfrac{\operatorname{Cov}\big(\widetilde{h}(Y), h_{0,i}(Y)\big)}{\mathbb{V}\mathrm{ar}\big[h_{0,i}(Y)\big]}$

On a donc $\rho\big(\widetilde{h}, h_{0,1}\big) = \dfrac{\operatorname{Cov}\big(1 - F(t-Y), Y\big)}{\sqrt{\mathbb{V}\mathrm{ar}[1 - F(t-Y)]\mathbb{V}\mathrm{ar}[Y]}}$ et $\rho\big(\widetilde{h}, h_{0,2}\big) = \dfrac{\operatorname{Cov}\big(1 - F(t-Y), \mathbb{1}_{Y \geq t - q_\varepsilon}\big)}{\sqrt{\mathbb{V}\mathrm{ar}[1 - F(t-Y)]\mathbb{V}\mathrm{ar}[\mathbb{1}_{Y \geq t - q_\varepsilon}]}}$

Calculons à l'aide de R $\rho\big(\widetilde{h}, h_{0,1}\big)$ et $\rho\big(\widetilde{h}, h_{0,2}\big)$, et nous choisirons la fonction telle que $\rho\big(\widetilde{h}, h_{0,i}\big)$ est la plus proche de 1 :

```r
# Fonction h_tilde définie par y |--> 1-F(t-y) où F est la f.d.r. de f
h_tilde <- function(y,t=0){
  return(1- fdr_F(t-y))
}

# Fonction h_0,1 définie par y |--> y
h_0_1 <- function(y){
  return(y)
}

# Fonction h_0,2 définie par y |--> 1_{y  t - q_{epsilon}
h_0_2 <- function(y, alpha = 3, theta = 2, t = 0, epsilon = 0.6){
  q <- qgamma(epsilon, shape = alpha, scale =theta)
  return(1*(y >= t - q))
}

# Fonction permettant de calculer rho(h,h_0) (pour une v.a. Y)
rho <- function(Y,h,h_0){
  Cov_h_h_0 <- cov(h, h_0) # on calculer la covariance entre h(Y) et h_0(Y)
  var_h_0 <- var(h_0) # on calcule la variance de h_0(Y)
  var_h <- var(h) # on calcule la variance de h(Y)
  # on renvoie rho(h,h_0)
  return(Cov_h_h_0/sqrt(var_h*var_h_0))
}

# On vérifie qu'elle est le rho le plus proche de 1 entre
# rho(h_tilde,h_0_1) et rho(h_tilde,h_0_2)

# on va simuler n = 1000 v.a. suivant la loi de Y (densité g)
n <- 1000
# on va comparer les rho sur I = 100 itérations
```

```r
I <- 100
# on va compter le nombre de fois où rho(h_tilde,h_0_1) est plus
# grand que rho(h_tilde,h_0_2)
# on crée un vecteur qui sera composé de I = 100 simulations de rho(h_tilde,h_0_1)
rho_1 <- numeric(I)
# on crée un vecteur qui sera composé de I = 100 simulations de rho(h_tilde,h_0_2)
rho_2 <- numeric(I)
k <- 0 # on initialise le compteur k à 0
for (i in 1:I){ # pour i variant de 1 à I = 100,
  Y <- rgen_g(n) # on simule n v.a. suivant g
  rho_1[i] <- rho(Y,h_tilde(Y),h_0_1(Y)) # on remplit rho_1
  rho_2[i] <- rho(Y,h_tilde(Y),h_0_2(Y)) # on remplit rho_2
  # on ajoute au compteur 1 si rho(h_tilde,h_0_1) > rho(h_tilde,h_0_2) et 0 sinon
  k <- k + (rho_1[i] > rho_2[i])
}
# on renvoie le compteur k
k

## [1] 100

# on renvoie les valeurs moyennes de rho(h_tilde,h_0_1) et rho(h_tilde,h_0_2)
c(mean(rho_1),mean(rho_2))

## [1] 0.8646628 0.7475063
```

D'après les résultats du code précédent, sur 100 itérations, $\rho\big(\widetilde{h}, h_{0,1}\big) > \rho\big(\widetilde{h}, h_{0,2}\big)$ et en moyenne $\rho\big(\widetilde{h}, h_{0,1}\big) \simeq 0,8646628 > 0,7475063 \simeq \rho\big(\widetilde{h}, h_{0,2}\big)$.

Il vaut donc mieux choisir fonction $h_{0,1}$ pour réduire la variance de $\widehat{\delta}_n$.

Donc on pose :

$$\widehat{\delta}_n^{\text{cont}}(b) = \frac{1}{n}\sum_{k=1}^{n} \widetilde{h}(Y_k) - b\big(h_{0,1}(Y_k) - \mathbb{E}\big[h_{0,1}(Y_k)\big]\big) \text{ avec } \big(Y_k\big)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} Y$$

Or on sait que Y et $G^{\leftarrow}(U)$ sont de même loi pour $U \sim \mathscr{U}\big([0,1]\big)$, d'où :

$$\widehat{\delta}_n^{\text{cont}}(b) = \frac{1}{n}\sum_{k=1}^{n} \widetilde{h}\big(G^{\leftarrow}(U_k)\big) - b\Big(h_{0,1}\big(G^{\leftarrow}(U_k)\big) - m\Big) \text{ avec } \big(U_k\big) \overset{\text{i.i.d.}}{\sim} \mathscr{U}\big([0,1]\big) \text{ et } m = \mathbb{E}\Big[h_{0,1}\big(G^{\leftarrow}(U)\big)\Big]$$

Or $\mathbb{E}\Big[h_{0,1}\big(G^{\leftarrow}(U)\big)\Big] = \mathbb{E}\Big[h_{0,1}(Y)\Big] = \mathbb{E}[Y] = \int_{\mathbb{R}} y g(y)\, dy = \underbrace{\int_{\mu-2b}^{\mu} y \times \frac{y-(\mu-2b)}{4b^2}\, dy}_{=:I_1} + \underbrace{\int_{\mu}^{+\infty} \frac{y}{2b}\exp\Big(-\frac{y-\mu}{b}\Big)\, dy}_{=:I_2}$

Avec $I_1 = \left[\frac{y^3}{12b^2}\right]_{\mu-2b}^{\mu} - \frac{\mu-2b}{4b^2}\left[\frac{y^2}{2}\right]_{\mu}^{+\infty}$

$$= \frac{\mu^3}{12b^2} - \frac{(\mu-2b)^3}{12b^2} - \frac{\mu^2(\mu-2b)}{8b^2} + \frac{(\mu-2b)^3}{8b^2}$$

$$= \frac{\mu^3}{12b^2} - \frac{\mu^3 - 2b\mu^2}{8b^2} + \frac{(\mu-2b)^3}{24b^2}$$

$$= \frac{2\mu^3 - 3\mu^3 + 6b\mu^2 + \mu^3 - 6b\mu^2 + 12b^2\mu - 8b^3}{24b^2}$$

Donc $\boxed{I_1 = \frac{\mu}{2} - \frac{b}{3}}$

Et par intégration par partie, en posant :

$$\begin{cases} u(y) = y \implies u'(y) = 1 \\ v'(y) = \exp\left(-\frac{y-\mu}{b}\right) \implies v(y) = -b\exp\left(-\frac{y-\mu}{b}\right) \end{cases}$$

on a $I_2 = \frac{1}{2b}\left[-by\exp\left(-\frac{y-\mu}{b}\right)\right]_{\mu}^{+\infty} + \frac{1}{2}\int_{\mu}^{+\infty} \exp\left(-\frac{y-\mu}{b}\right) dy$

$$= -\frac{1}{2}\Bigg(\underbrace{\lim_{y\to+\infty} y\exp\left(-\frac{y-\mu}{b}\right)}_{=0 \text{ par croissances comparées}} - \underbrace{\mu\exp\left(-\frac{\mu-\mu}{b}\right)}_{=\mu}\Bigg) + \frac{1}{2}\left[-b\exp\left(-\frac{y-\mu}{b}\right)\right]_{\mu}^{+\infty}$$

$$= \frac{\mu}{2} - \frac{b}{2}\Bigg(\underbrace{\lim_{y\to+\infty} \exp\left(-\frac{y-\mu}{b}\right)}_{=0} - \underbrace{\exp\left(-\frac{\mu-\mu}{b}\right)}_{=1}\Bigg)$$

Donc $\boxed{I_2 = \frac{\mu}{2} + \frac{b}{2}}$

Ainsi, on trouve que $\mathbb{E}[Y] = I_1 + I_2 = \frac{\mu}{2} - \frac{b}{3} + \frac{\mu}{2} + \frac{b}{2}$, *i.e.* $\boxed{m = \mathbb{E}[Y] = \mu + \frac{b}{6}}$

De plus, avec R, on détermine le b optimal, noté $b^\star$:

```
# calcul du b optimal pour l'estimateur de la méthode de la variable de contrôle
b_star <- function(Y){
  # on applique la formule donnée précédemment
  return(cov(h_tilde(Y),h_0_1(Y))/var(h_0_1(Y)))
}
Y <- rgen_g(n)
b_star(Y)

## [1] 0.03452228
```

Donc l'estimateur $\widehat{\delta}_n^{\text{cont}}(b)$ est défini par :

$$\boxed{\widehat{\delta}_n^{\text{cont}}(b) = \frac{1}{n}\sum_{k=1}^{n} 1 - F\big(t - G^{\leftarrow}(U_k)\big) - b\left(G^{\leftarrow}(U_k) - \left(\mu + \frac{b}{6}\right)\right)}$$

avec $(U_k)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}([0,1])$ et $b = \dfrac{\text{Cov}\big(1 - F(t-Y), Y\big)}{\mathbb{V}\text{ar}[Y]}$

12. (♠)Implémenter un code R qui fournit, pour $n = 1000$, l'estimation $\widehat{\delta}_n^{\text{cont}}$, l'erreur quadratique moyenne de $\widehat{\delta}_n^{\text{cont}}$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```
                    #### Estimation de delta n°4 ####

# Estimation de delta_cont (méthode de la variable de contrôle)
delta_cont <- function(n, mu = -13, b=2, alpha = 3, theta = 2, epsilon = 0.6, t=0 ){
  Y <- rgen_g(n) # on simule n v.a. suivant g
  b_cont <- b_star(Y) # on calcule le b optimal
  m <- mu + b/6 # on calcule m, l'espérance de Y (calculée théoriquement)
  h <- h_tilde(Y) - b_cont*(h_0_1(Y) - m) # on calcule h_cont
  return(h)# on retourne h_cont
}

#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(delta_cont(n))

##
## Estimateur                                   0.069737025
## Erreur quadratique moyenne                   0.002529714
## Intervalle de confiance    0.06661969 0.072854360
```

Cette nouvelle estimation de δ par l'estimateur $\widehat{\delta}_n^{\text{cont}}$ donne une variance et un intervalle de confiance plus petits qu'avec tous les estimateurs précédents. On a donc encore une fois amélioré l'estimation de δ.

Pour $\left(Y_k\right)_{k\geq 1}$, une suite de variables aléatoires i.i.d. suivant g, on veut construire le meilleur estimateur de la forme :

$$\widehat{\delta}_n(\beta) = 1 - \frac{1}{n}\sum_{k=1}^{n} F(t-Y_k) - \langle \beta, \begin{pmatrix} Y_k - \mathbb{E}[Y_k] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}[Y_k \geq t-q_\varepsilon] \end{pmatrix} \rangle, \quad \text{avec} \quad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} \in \mathbb{R}^2$$

où $\langle \cdot, \cdot \rangle$ désigne le produit scalaire usuel de $\mathbb{R}^2$, *i.e.*, $\langle U, V\rangle = U^T V$.

13. Montrer que

$$\mathbb{V}\text{ar}\left[\widehat{\delta}_n(\beta)\right] = \mathbb{V}\text{ar}\left[\widehat{\delta}_n\right] - \frac{1}{n}\Big(2\langle \beta, C\rangle - \langle \beta, \Sigma\beta\rangle\Big)$$

avec $C = \begin{pmatrix} \text{Cov}\big(F(t-Y), Y\big) \\ \text{Cov}\big(F(t-Y), \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) \end{pmatrix}$ et $\Sigma = \begin{pmatrix} \mathbb{V}\text{ar}[Y] & \text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) \\ \text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) & \mathbb{V}\text{ar}\left[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] \end{pmatrix}$

$$\begin{aligned}
\text{On a } \mathbb{V}\text{ar}\left[\widehat{\delta}_n(\beta)\right] &= \mathbb{V}\text{ar}\left[1 - \frac{1}{n}\sum_{k=1}^{n} F(t-Y_k) - \langle \beta, \begin{pmatrix} Y_k - \mathbb{E}[Y_k] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}[Y_k \geq t-q_\varepsilon] \end{pmatrix} \rangle\right] \\
&= \frac{1}{n^2}\sum_{k=1}^{n} \mathbb{V}\text{ar}\left[F(t-Y_k) - \langle \beta, \begin{pmatrix} Y_k - \mathbb{E}[Y_k] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}[Y_k \geq t-q_\varepsilon] \end{pmatrix} \rangle\right] \text{ par indépendance} \\
&= \frac{1}{n^2}\sum_{k=1}^{n} \mathbb{V}\text{ar}\left[F(t-Y_k)\right] + \mathbb{V}\text{ar}\left[\langle \beta, \begin{pmatrix} Y_k - \mathbb{E}[Y_k] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}[Y_k \geq t-q_\varepsilon] \end{pmatrix} \rangle\right] \\
&\quad -2\text{Cov}\left(F(t-Y_k), \langle \beta, \begin{pmatrix} Y_k - \mathbb{E}[Y_k] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}[Y_k \geq t-q_\varepsilon] \end{pmatrix} \rangle\right)
\end{aligned}$$

Donc $\mathbb{V}\text{ar}\left[\widehat{\delta}_n(\beta)\right] = \overbrace{\frac{1}{n}\mathbb{V}\text{ar}\Big[F(t-Y)\Big]}^{=\mathbb{V}\text{ar}\left[\widehat{\delta}_n\right]} + \frac{1}{n}\overbrace{\mathbb{V}\text{ar}\left[\langle\beta, \begin{pmatrix} Y-\mathbb{E}[Y] \\ \mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon] \end{pmatrix}\rangle\right]}^{=:A}$

$$-\frac{1}{n}\times \underbrace{2\text{Cov}\left(F(t-Y), \langle\beta, \begin{pmatrix} Y-\mathbb{E}[Y] \\ \mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon] \end{pmatrix}\rangle\right)}_{=:B} \text{ car les } (Y_k)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} Y$$

▶ **Or d'une part :** $A = \mathbb{V}\text{ar}\left[\langle\beta, \begin{pmatrix} Y-\mathbb{E}[Y] \\ \mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon] \end{pmatrix}\rangle\right]$

$$= \mathbb{V}\text{ar}\left[\beta_1\Big(Y-\mathbb{E}[Y]\Big) + \beta_2\Big(\mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon]\Big)\right]$$

$$= \mathbb{V}\text{ar}\left[\beta_1 Y - \underbrace{\beta_1\mathbb{E}[Y]}_{\text{constante}} + \beta_2\mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \underbrace{\beta_2\mathbb{P}[Y\geq t-q_\varepsilon]}_{\text{constante}}\Big)\right]$$

$$= \beta_1^2\mathbb{V}\text{ar}[Y] + \beta_2^2\mathbb{V}\text{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] + 2\beta_1\beta_2\text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big)$$

$$= \begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix}^T \begin{pmatrix} \beta_1\mathbb{V}\text{ar}[Y] & \beta_2\text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) \\ \beta_1\text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) & \beta_2\mathbb{V}\text{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] \end{pmatrix}$$

$$= \begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix}^T \begin{pmatrix} \mathbb{V}\text{ar}[Y] & \text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) \\ \text{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) & \mathbb{V}\text{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] \end{pmatrix}\begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix}$$

Donc $\boxed{A = \langle\beta, \Sigma\beta\rangle}$

▶ **D'autre part :** $B = 2\text{Cov}\left(F(t-Y), \langle\beta, \begin{pmatrix} Y-\mathbb{E}[Y] \\ \mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon] \end{pmatrix}\rangle\right)$

$$= 2\text{Cov}\left(F(t-Y), \beta_1\Big(Y-\mathbb{E}[Y]\Big) + \beta_2\Big(\mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \mathbb{P}[Y\geq t-q_\varepsilon]\Big)\right)$$

$$= 2\text{Cov}\left(F(t-Y), \beta_1 Y - \underbrace{\beta_1\mathbb{E}[Y]}_{\text{constante}} + \beta_2\mathbb{1}_{\{Y\geq t-q_\varepsilon\}} - \underbrace{\beta_2\mathbb{P}[Y\geq t-q_\varepsilon]}_{\text{constante}}\right)$$

$$= 2\beta_1\text{Cov}\Big(F(t-Y), Y\Big) + 2\beta_2\text{Cov}\Big(F(t-Y), \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\Big)$$

$$= 2\begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix}^T \begin{pmatrix} \text{Cov}\Big(F(t-Y), Y\Big) \\ \text{Cov}\Big(F(t-Y), \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\Big) \end{pmatrix}$$

Donc $\boxed{B = 2\langle\beta, C\rangle}$

Or $\mathbb{V}\text{ar}[\widehat{\delta}_n(\beta)] = \mathbb{V}\text{ar}[\widehat{\delta}_n] + \frac{1}{n}A - \frac{1}{n}B$

$$= \mathbb{V}\text{ar}[\widehat{\delta}_n] - \frac{1}{n}\langle\beta, \Sigma\beta\rangle - \frac{1}{n}\times 2\langle\beta, C\rangle$$

Donc $\boxed{\mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta)\big] = \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] + \frac{1}{n}\Big(2\langle \beta, C\rangle - \langle \beta, \Sigma\beta\rangle\Big)}$

14. Calculer explicitement Σ et vérifier qu'elle est inversible.

Calculons les éléments de $\Sigma = \begin{pmatrix} \mathbb{V}\mathrm{ar}[Y] & \mathrm{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) \\ \mathrm{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big) & \mathbb{V}\mathrm{ar}\big[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big] \end{pmatrix}$

▸ Commençons par $\mathbb{V}\mathrm{ar}[Y] = \mathbb{E}[Y^2] - \Big(\mathbb{E}[Y]\Big)^2$

On a calculé précédemment $\boxed{\mathbb{E}[Y] = \mu + \frac{b}{6}}$.

De même $\mathbb{E}[Y^2] = \displaystyle\int_{\mathbb{R}} y^2 g(y)\, dy = \underbrace{\int_{\mu-2b}^{\mu} y^2 \times \frac{y-(\mu-2b)}{4b^2}\, dy}_{=:I_3} + \underbrace{\int_{\mu}^{+\infty} \frac{y^2}{2b} \exp\Big(-\frac{y-\mu}{b}\Big)\, dy}_{=:I_4}$

$$\begin{aligned}
\text{Or } I_3 &= \left[\frac{y^4}{16b^2}\right]_{\mu-2b}^{\mu} - \frac{\mu-2b}{4b^2}\left[\frac{y^3}{3}\right]_{\mu}^{+\infty} \\
&= \frac{\mu^4}{16b^2} - \frac{(\mu-2b)^4}{16b^2} - \frac{\mu^3(\mu-2b)}{12b^2} + \frac{(\mu-2b)^4}{12b^2} \\
&= \frac{3\mu^4 - 4\mu^4 + 8b\mu^3 + \mu^4 - 8b\mu^3 + 24b^2\mu^2 - 32b^3\mu + 16b^4}{48b^2} \\
&= \frac{24\mu^2 - 32b\mu + 16b^2}{48}
\end{aligned}$$

Donc $\boxed{I_3 = \frac{\mu^2}{2} - \frac{2}{3}b\mu + \frac{b^2}{3}}$

Et par intégration par partie, en posant :

$$\begin{cases} u(y) = y^2 \implies u'(y) = 2y \\ v'(y) = \exp\Big(-\frac{y-\mu}{b}\Big) \implies v(y) = -b\exp\Big(-\frac{y-\mu}{b}\Big) \end{cases}$$

on a $I_4 = \frac{1}{2b}\left[-by^2\exp\Big(-\frac{y-\mu}{b}\Big)\right]_{\mu}^{+\infty} - \frac{1}{2b}\times(-2b)\displaystyle\int_{\mu}^{+\infty} y\exp\Big(-\frac{y-\mu}{b}\Big)\, dy$

Idem par intégration par partie, en posant :

$$\begin{cases} \widetilde{u}(y) = y \implies \widetilde{u}'(y) = 1 \\ \widetilde{v}'(y) = \exp\Big(-\frac{y-\mu}{b}\Big) \implies \widetilde{v}(y) = -b\exp\Big(-\frac{y-\mu}{b}\Big) \end{cases}$$

on a $I_4 = -\frac{1}{2}\Bigg(\underbrace{\lim_{y\to+\infty} y^2\exp\Big(-\frac{y-\mu}{b}\Big)}_{=0 \text{ par croissances comparées}} - \underbrace{\mu^2\exp\Big(-\frac{\mu-\mu}{b}\Big)}_{=\mu^2}\Bigg) + \left[-by\exp\Big(-\frac{y-\mu}{b}\Big)\right]_{\mu}^{+\infty} + b\displaystyle\int_{\mu}^{+\infty}\exp\Big(-\frac{y-\mu}{b}\Big)\, dy$

$$= \frac{\mu^2}{2} - b\Big(\underbrace{\lim_{y\to+\infty} y\exp\Big(-\frac{y-\mu}{b}\Big)}_{=0 \text{ par croissances comparées}} - \underbrace{\mu\exp\Big(-\frac{\mu-\mu}{b}\Big)}_{=\mu}\Big) + b\Big[-b\exp\Big(-\frac{y-\mu}{b}\Big)\Big]_{\mu}^{+\infty}$$

$$= \frac{\mu^2}{2} + b\mu - b^2\Big(\underbrace{\lim_{y\to+\infty} \exp\Big(-\frac{y-\mu}{b}\Big)}_{=0} - \underbrace{\exp\Big(-\frac{\mu-\mu}{b}\Big)}_{=1}\Big)$$

Donc $\boxed{I_4 = \frac{\mu^2}{2} + b\mu + b^2}$

Ainsi on trouve que $\mathbb{E}[Y^2] = I_3 + I_4 = \frac{\mu^2}{2} - \frac{2}{3}b\mu + \frac{b^2}{3} + \frac{\mu^2}{2} + b\mu + b^2$, *i.e.* $\boxed{\mathbb{E}[Y^2] = \mu^2 + \frac{b\mu}{3} + \frac{4}{3}b^2}$

D'où $\mathbb{V}\mathrm{ar}[Y] = \mathbb{E}[Y^2] - \big(\mathbb{E}[Y]\big)^2$

$$= \mu^2 + \frac{b\mu}{3} + \frac{4}{3}b^2 - \Big(\mu + \frac{b}{6}\Big)^2$$

$$= \mu^2 + \frac{b\mu}{3} + \frac{4}{3}b^2 - \mu^2 + \frac{b\mu}{3} - \frac{b^2}{36}$$

Donc $\boxed{\mathbb{V}\mathrm{ar}[Y] = \frac{47b^2}{36}}$

► Maintenant, calculons $\mathbb{V}\mathrm{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = \mathbb{E}[\mathbb{1}^2_{\{Y\geq t-q_\varepsilon\}}] - \big(\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}]\big)^2$

D'une part on a, $\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = \mathbb{P}[Y \geq t - q_\varepsilon]$

$$= 1 - \mathbb{P}[Y \leq t - q_\varepsilon]$$

$$= 1 - G(t - q_\varepsilon)$$

Or avec les valeurs des paramètres qui nous sont donnés, on a $t - q_\varepsilon = -q_{0,6} \simeq -6,21 > -13 = \mu$

D'où $\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = 1 - \Big(1 - \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big)\Big)$, *i.e.* $\boxed{\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big)}$

D'autre part, on a $\mathbb{E}[\mathbb{1}^2_{\{Y\geq t-q_\varepsilon\}}] = \mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}]$ car $\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}$ ne prend que 0 ou 1 comme valeurs (dont leurs carrés sont égaux à eux-mêmes).

d'où $\mathbb{V}\mathrm{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = \mathbb{E}[\mathbb{1}^2_{\{Y\geq t-q_\varepsilon\}}] - \big(\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}]\big)^2$

$$= \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big) - \Big(\frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big)\Big)^2$$

$$= \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big) - \frac{1}{4}\exp\Big(-2\times\frac{t-q_\varepsilon-\mu}{b}\Big)$$

Donc $\boxed{\mathbb{V}\mathrm{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] = \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big)\Big(1 - \frac{1}{2}\exp\Big(-\frac{t-q_\varepsilon-\mu}{b}\Big)\Big)}$

► Enfin calculons $\mathrm{Cov}(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}) = \mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] - \mathbb{E}[Y]\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}]$

On a
$$\begin{aligned}\mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] &= \int_{\mathbb{R}} y\mathbb{1}_{\{y\geq t-q_\varepsilon\}} g(y)\,dy \\ &= \int_{t-q_\varepsilon}^{+\infty} y g(y)\,dy\end{aligned}$$

Or on a vu que pour les valeurs des paramètres données, on a $t-q_\varepsilon = -q_{0,6} \simeq -6,21 > -13 = \mu$

Donc $\mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] = \displaystyle\int_{t-q_\varepsilon}^{+\infty} \frac{y}{2b}\exp\left(-\frac{y-\mu}{b}\right)dy$

D'où par intégration par partie, en posant :

$$\begin{cases} \tilde{\tilde{u}}(y) = y \implies \tilde{\tilde{u}}'(y) = 1 \\ \tilde{\tilde{v}}'(y) = \exp\left(-\dfrac{y-\mu}{b}\right) \implies \tilde{\tilde{v}}(y) = -b\exp\left(-\dfrac{y-\mu}{b}\right) \end{cases}$$

On a
$$\begin{aligned}\mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] &= \frac{1}{2b}\left[-by\exp\left(-\frac{y-\mu}{b}\right)\right]_{t-q_\varepsilon}^{+\infty} + \frac{1}{2}\int_{t-q_\varepsilon}^{+\infty}\exp\left(-\frac{y-\mu}{b}\right)dy \\ &= -\frac{1}{2}\Bigg(\underbrace{\lim_{y\to+\infty} y\exp\left(-\frac{y-\mu}{b}\right)}_{=0 \text{ par croissances comparées}} - (t-q_\varepsilon)\exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right)\Bigg) + \frac{1}{2}\left[-b\exp\left(-\frac{y-\mu}{b}\right)\right]_{t-q_\varepsilon}^{+\infty} \\ &= \frac{1}{2}(t-q_\varepsilon)\exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right) - \frac{b}{2}\Bigg(\underbrace{\lim_{y\to+\infty}\exp\left(-\frac{y-\mu}{b}\right)}_{=0} - \exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right)\Bigg)\end{aligned}$$

D'où $\boxed{\mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] = \dfrac{1}{2}(t-q_\varepsilon+b)\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right)}$

D'où
$$\begin{aligned}\mathrm{Cov}(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}) &= \mathbb{E}\left[Y\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\right] - \mathbb{E}[Y]\mathbb{E}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}] \\ &= \frac{1}{2}(t-q_\varepsilon+b)\exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right) - \left(\mu+\frac{b}{6}\right)\times\frac{1}{2}\exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right) \\ &= \frac{1}{2}\exp\left(-\frac{t-q_\varepsilon-\mu}{b}\right)\left(t-q_\varepsilon+b-\mu-\frac{b}{6}\right)\end{aligned}$$

Donc $\boxed{\mathrm{Cov}(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}) = \dfrac{1}{2}\left(t-q_\varepsilon-\mu+\dfrac{5}{6}b\right)\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right)}$

Ainsi, on a
$$\Sigma = \begin{pmatrix} \dfrac{47}{36}b^2 & \dfrac{1}{2}\left(t-q_\varepsilon-\mu+\dfrac{5}{6}b\right)\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right) \\ \dfrac{1}{2}\left(t-q_\varepsilon-\mu+\dfrac{5}{6}b\right)\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right) & \dfrac{1}{2}\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right)\left(1-\dfrac{1}{2}\exp\left(-\dfrac{t-q_\varepsilon-\mu}{b}\right)\right) \end{pmatrix}$$

Calculons le déterminant de cette matrice sur R pour vérifier si la matrice est inversible ou non :

```
                #### Vérification que Sigma est inversible ####
# Valeurs des paramètres
b <- 2
```

```
mu <- -13
epsilon <- 0.6
q <- qgamma(epsilon,shape = 3, scale = 2)
t <- 0

# On calcule les éléments (variances et covariance) de Sigma à l'aide de leur formule
# théorique qu'on a calculé
Cov_Y_Ind <- 1/2*exp(-(t-q-mu)/b)*(t-q-mu+5*b/6)
Var_Ind <- 1/2*exp(-(t-q-mu)/b)*(1-1/2*exp(-(t-q-mu)/b))
Var_Y <- 47*b*b/36

# On crée la matrice Sigma
Sigma <- matrix(c(Var_Y,Cov_Y_Ind,Cov_Y_Ind,Var_Ind),2)
Sigma

##            [,1]       [,2]
## [1,] 5.2222222 0.14186164
## [2,] 0.1418616 0.01649517

# On calcule le déterminant de Sigma
det(Sigma)

## [1] 0.06601673

# On calcule l'inverse de Sigma
Sigma_inv <- solve(Sigma)
Sigma_inv

##             [,1]      [,2]
## [1,]  0.2498635 -2.148874
## [2,] -2.1488742 79.104528
```

Le déterminant de Σ est environ égal à $0,06601673 \neq 0$, donc Σ est bien inversible.

15. Montrer que

$$\beta^\star = \underset{\beta\in\mathbb{R}^2}{\operatorname{argmin}}\ \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta)\big] = \Sigma^{-1}C \quad \text{et} \quad \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta^\star)\big] \leq \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big]$$

On cherche $\beta^\star = \underset{\beta\in\mathbb{R}^2}{\operatorname{argmin}}\ \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta)\big]$

$$\iff \beta^\star = \underset{\beta\in\mathbb{R}^2}{\operatorname{argmin}}\ \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] - \frac{1}{n}\Big(2\langle\beta,C\rangle - \langle\beta,\Sigma\beta\rangle\Big)$$

$$\iff \beta^\star = \underset{\beta\in\mathbb{R}^2}{\operatorname{argmin}}\ -\frac{1}{n}\Big(2\langle\beta,C\rangle - \langle\beta,\Sigma\beta\rangle\Big) \ \Big(\text{car } \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\ k - f(x) = \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\ -f(x), \text{ avec } k\in\mathbb{R}^n \text{ fixé}\Big)$$

Donc $\beta^\star = \underset{\beta\in\mathbb{R}^2}{\operatorname{argmax}}\Big(2\langle\beta,C\rangle - \langle\beta,\Sigma\beta\rangle\Big)\ \Big(\text{car } \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\ -f(x) = \underset{x\in\mathbb{R}^n}{\operatorname{argmax}}\ f(x)\Big)$

D'après la question **(13.)**, $2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle = 2\beta_1 \overbrace{\mathrm{Cov}\big(F(t-Y), Y\big)}^{=:a} + 2\beta_2 \overbrace{\mathrm{Cov}\big(F(t-Y), \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big)}^{=:b}$

$$-\Big(\beta_1^2 \underbrace{\mathbb{V}\mathrm{ar}[Y]}_{=:c} + 2\beta_1\beta_2 \underbrace{\mathrm{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big)}_{=:d} + \beta_2^2 \underbrace{\mathbb{V}\mathrm{ar}[\mathbb{1}_{Y\geq t-q_\varepsilon}]}_{=:e}\Big)$$

On a $\mathbb{R}^2$ est ouvert la condition nécessaire d'optimalité est donc $\nabla_\beta\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big) = 0_{\mathbb{R}^2}$

$$i.e. \begin{cases} \dfrac{\partial\big(2\langle\beta^\star, C\rangle - \langle\beta^\star, \Sigma\beta^\star\rangle\big)}{\partial\beta_1} = 0 \\ \\ \dfrac{\partial\big(2\langle\beta^\star, C\rangle - \langle\beta^\star, \Sigma\beta^\star\rangle\big)}{\partial\beta_2} = 0 \end{cases} \iff \begin{cases} 2a - \big(2c\beta_1^\star + 2d\beta_2^\star\big) = 0 \\ 2b - \big(2e\beta_2^\star + 2d\beta_1^\star\big) = 0 \end{cases} \iff \begin{cases} \beta_1^\star = \dfrac{a - d\beta_2^\star}{c} & (1) \\ b - e\beta_2^\star - d \times \dfrac{a - d\beta_2^\star}{c} = 0 & (2) \end{cases}$$

Or (2) donne : $bc - ce\beta_2^\star - ad + d^2\beta_2^\star = 0$, *i.e.* $\beta_2^\star(d^2 - ce) = ad - bc$ et donc $\boxed{\beta_2^\star = \dfrac{ad - bc}{d^2 - ce}}$

D'où dans (1) on a : $\beta_1^\star = \dfrac{a}{c} - d \times \dfrac{ad - bc}{c(d^2 - ce)}$, *i.e.* $\beta_1^\star = \dfrac{ad^2 - ace}{c(d^2 - ce)} - \dfrac{ad^2 - bcd}{c(d^2 - ce)}$ et donc $\boxed{\beta_1^\star = \dfrac{bd - ae}{d^2 - ce}}$

De plus, on a $\mathrm{Hess}_\beta\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big) = \begin{pmatrix} \dfrac{\partial^2\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big)}{\partial\beta_1^2} & \dfrac{\partial^2\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big)}{\partial\beta_1\partial\beta_2} \\ \\ \dfrac{\partial^2\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big)}{\partial\beta_2\partial\beta_1} & \dfrac{\partial^2\big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\big)}{\partial\beta_2^2} \end{pmatrix} = \begin{pmatrix} -2c & -2d \\ -2d & -2e \end{pmatrix}$

D'où pour $u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in \mathbb{R}^2$ <u>non nul</u> quelconque, on a :

$$\begin{aligned} u^T \mathrm{Hess}_\beta\Big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\Big) u &= \begin{pmatrix} u_1 & u_2 \end{pmatrix} \begin{pmatrix} -2c & -2d \\ -2d & -2e \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \\ &= \begin{pmatrix} -2cu_1 - 2du_2 & -2du_1 - 2eu_2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \\ &= -2\big(cu_1^2 + 2du_1u_2 + eu_2^2\big) \\ &= -2\Big(\mathbb{V}\mathrm{ar}[Y]u_1^2 + 2\mathrm{Cov}\big(Y, \mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big)u_1u_2 + \mathbb{V}\mathrm{ar}[\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}]u_2^2\Big) \end{aligned}$$

$$u^T \mathrm{Hess}_\beta\Big(2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle\Big) u = -2\underbrace{\mathbb{V}\mathrm{ar}\big[u_1Y + u_2\mathbb{1}_{\{Y\geq t-q_\varepsilon\}}\big]}_{>0} < 0$$

Donc la matrice Hessienne de l'application $\beta \longmapsto 2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle$ est définie négative.

L'application $\beta \longmapsto 2\langle\beta, C\rangle - \langle\beta, \Sigma\beta\rangle$ est donc strictement concave.

La condition suffisante d'optimalité est donc bien vérifiée, et l'unique argument qui maximise cette application est $\beta^\star = \begin{pmatrix} \dfrac{bd-ae}{d^2-ce} \\ \dfrac{ad-bc}{d^2-ce} \end{pmatrix} = \dfrac{1}{ce-d^2}\begin{pmatrix} ae-bd \\ -ad+bc \end{pmatrix} = \dfrac{1}{ce-d^2}\begin{pmatrix} e & -d \\ -d & c \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix}$

Or il est clair qu'avec les notations utilisées :

- ▷ $a = \mathrm{Cov}\big(F(t-Y), Y\big)$
- ▷ $b = \mathrm{Cov}\big(F(t-Y), \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big)$
- ▷ $c = \mathbb{V}\mathrm{ar}\big[Y\big]$
- ▷ $d = \mathrm{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big)$
- ▷ $e = \mathbb{V}\mathrm{ar}\big[\mathbb{1}_{Y \geq t-q_\varepsilon}\big]$

On a $\Sigma = \begin{pmatrix} \mathbb{V}\mathrm{ar}\big[Y\big] & \mathrm{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) \\ \mathrm{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) & \mathbb{V}\mathrm{ar}\big[\mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big] \end{pmatrix} = \begin{pmatrix} c & d \\ d & e \end{pmatrix}$ et $C = \begin{pmatrix} \mathrm{Cov}\big(F(t-Y), Y\big) \\ \mathrm{Cov}\big(F(t-Y), \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$

On a également vu qu'à la question **(14.)**, Σ est inversible car son déterminant est non nul.

Or le déterminant de Σ est donné par : $\det\big(\Sigma\big) = ce - d^2$, et on a donc $\Sigma^{-1} = \dfrac{1}{ce-d^2}\begin{pmatrix} e & -d \\ -d & c \end{pmatrix}$

Donc on trouve que $\boxed{\beta^\star = \underset{\beta \in \mathbb{R}^2}{\mathrm{argmax}}\Big(2\langle \beta, C\rangle - \langle \beta, \Sigma\beta\rangle\Big) = \underset{\beta \in \mathbb{R}^2}{\mathrm{argmin}}\ \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta)\big] = \Sigma^{-1}C}$.

► Maintenant, montrons que $\mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta^\star)\big] \leq \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big]$:

On vient de voir que $\beta^\star$ est l'unique argument qui minimise $\mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta)\big]$.

Or on a d'après la question **(13.)**,
$$\begin{aligned} \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n(\beta^\star)\big] &= \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] - \frac{1}{n}\Big(2\langle \beta^\star, C\rangle - \langle \beta^\star, \Sigma\beta^\star\rangle\Big) \\ &= \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] - \frac{1}{n}\Big(2\langle \Sigma^{-1}C, C\rangle - \langle \Sigma^{-1}C, \underbrace{\Sigma\Sigma^{-1}}_{=I_2}C\rangle\Big) \\ &= \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] - \frac{1}{n}\Big(2C^T\Sigma^{-1}C - C^T\Sigma^{-1}C\Big) \\ &= \mathbb{V}\mathrm{ar}\big[\widehat{\delta}_n\big] - \frac{1}{n}C^T\Sigma^{-1}C \end{aligned}$$

Or on définit la norme $x \in \mathbb{R}^2 \longmapsto \|x\|_{\Sigma^{-1}} = \sqrt{x^T\Sigma^{-1}x}$

Il s'agit bien d'une norme sur $\mathbb{R}^2$ car :

- ▷ Pour tout $x \in \mathbb{R}^2$, $\|x\|_{\Sigma^{-1}} = \sqrt{x^T\Sigma^{-1}x} \geq 0$ car le terme $\sqrt{x^T\Sigma^{-1}x}$ est la racine carrée de la somme des carrés des coordonnées de x pondérés par les éléments de la matrice Σ^{-1}, qui sont tous positifs.
- ▷ Si $x = 0_{\mathbb{R}^2}$, alors $\|x\|_{\Sigma^{-1}} = \sqrt{x^T\Sigma^{-1}x} = \sqrt{0_{\mathbb{R}^2}^T\Sigma^{-1}0_{\mathbb{R}^2}} = 0$.

  Réciproquement, si $\|x\|_{\Sigma^{-1}} = 0$ alors $\sqrt{x^T\Sigma^{-1}x} = 0$, *i.e.* $x^T\Sigma^{-1}x$ et comme Σ^{-1} est définie positive comme inverse d'une matrice variance (matrice de précision), alors nécessairement, on a $x = 0_{\mathbb{R}^2}$.
- ▷ Pour tout $(x, \lambda) \in \mathbb{R}^2, \mathbb{R}$, on a $\|\lambda x\|_{\Sigma^{-1}} = \sqrt{(\lambda x)^T\Sigma^{-1}(\lambda x)} = \sqrt{\lambda^2 \times x^T\Sigma^{-1}x} = |\lambda|\sqrt{x^T\Sigma^{-1}x} = |\lambda|\|x\|_{\Sigma^{-1}}$
- ▷ Pour tout $x, y \in \mathbb{R}^2$, on a $\|x+y\|_{\Sigma^{-1}} = \sqrt{(x+y)^T\Sigma^{-1}(x+y)} \leq \sqrt{x^T\Sigma^{-1}x} + \sqrt{y^T\Sigma^{-1}y} = \|x\|_{\Sigma^{-1}} + \|y\|_{\Sigma^{-1}}$

Donc on a $\mathbb{V}\text{ar}\big[\widehat{\delta}_n(\beta^\star)\big] = \mathbb{V}\text{ar}\big[\widehat{\delta}_n\big] - \dfrac{1}{n}C^T\Sigma^{-1}C = \mathbb{V}\text{ar}\big[\widehat{\delta}_n\big] \underbrace{- \dfrac{1}{n}\overbrace{\|C\|^2_{\Sigma^{-1}}}^{\geq 0}}_{\leq 0}$.

Donc on trouve bien que $\boxed{\mathbb{V}\text{ar}\big[\widehat{\delta}_n(\beta^\star)\big] \leq \mathbb{V}\text{ar}\big[\widehat{\delta}_n\big]}$.

16. L'estimateur $\widehat{\delta}_n(\beta^\star)$ est-il préférable en terme de variance à un estimateur basé sur la méthode de la variable de contrôle utilisant $h_0 : y \mapsto h_{0,1}(y) + h_{0,2}(y)$? Justifier rigoureusement votre réponse.

Si on choisit $h_0 : y \mapsto h_{0,1}(y) + h_{0,2}(y)$ comme fonction pour la méthode de la variable de contrôle, alors l'estimateur s'écrit :

$$\widehat{\delta}_n^{\text{cont}}(b) = \frac{1}{n}\sum_{k=1}^{n} 1 - F(t - Y_k) - b\Big(h_0(Y_k) - \mathbb{E}\big[h_0(Y_k)\big]\Big) \text{ avec } b = \frac{\text{Cov}\big(h(Y), h_0(Y)\big)}{\mathbb{V}\text{ar}\big[h_0(Y)\big]}$$

Or on a :

- $\text{Cov}\big(h(Y), h_0(Y)\big) = \text{Cov}\big(F(t-Y), Y\big) + \text{Cov}\big(F(t-Y), \mathbb{1}_{\{Y \geq t - q_\varepsilon\}}\big) = \langle C, \mathbf{1}_2\rangle$ avec $\mathbf{1}_2 = \begin{pmatrix}1\\1\end{pmatrix}$
- $\mathbb{V}\text{ar}\big[h_0(Y)\big] = \mathbb{V}\text{ar}\big[Y\big] + \mathbb{V}\text{ar}\big[\mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big] + 2\text{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big)$

$$= \Big(\mathbb{V}\text{ar}\big[Y\big] + \text{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) \quad \text{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) + \mathbb{V}\text{ar}\big[\mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big]\Big)\begin{pmatrix}1\\1\end{pmatrix}$$

$$= \begin{pmatrix}1 & 1\end{pmatrix}\begin{pmatrix}\mathbb{V}\text{ar}\big[Y\big] & \text{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big)\\ \text{Cov}\big(Y, \mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big) & \mathbb{V}\text{ar}\big[\mathbb{1}_{\{Y \geq t-q_\varepsilon\}}\big]\end{pmatrix}\begin{pmatrix}1\\1\end{pmatrix}$$

$$= \langle \mathbf{1}_2, \Sigma\mathbf{1}_2\rangle$$

- $h_0(Y_k) - \mathbb{E}\big[h_0(Y_k)\big] = \Big(Y_k - \mathbb{E}\big[Y_k\big]\Big) + \Big(\mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{E}\big[\mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}}\big]\Big) = \langle \mathbf{1}_2, \begin{pmatrix} Y_k - \mathbb{E}\big[Y_k\big] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}\big[Y_k \geq t - q_\varepsilon\big]\end{pmatrix}\rangle$

D'où $\widehat{\delta}_n^{\text{cont}}(b) = \dfrac{1}{n}\displaystyle\sum_{k=1}^{n} 1 - F(t - Y_k) - \underbrace{\dfrac{\langle C, \mathbf{1}_2\rangle}{\langle \mathbf{1}_2, \Sigma\mathbf{1}_2\rangle}}_{\in\mathbb{R}}\langle \mathbf{1}_2, \begin{pmatrix} Y_k - \mathbb{E}\big[Y_k\big] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}\big[Y_k \geq t - q_\varepsilon\big]\end{pmatrix}\rangle$

$$= 1 - \frac{1}{n}\sum_{k=1}^{n} F(t - Y_k) - \langle \underbrace{\frac{\langle C, \mathbf{1}_2\rangle}{\langle \mathbf{1}_2, \Sigma\mathbf{1}_2\rangle}\mathbf{1}_2}_{=:\widetilde{\beta}}, \begin{pmatrix} Y_k - \mathbb{E}\big[Y_k\big] \\ \mathbb{1}_{\{Y_k \geq t-q_\varepsilon\}} - \mathbb{P}\big[Y_k \geq t - q_\varepsilon\big]\end{pmatrix}\rangle$$

Donc $\widehat{\delta}_n^{\text{cont}}(b) = \widehat{\delta}_n\big(\widetilde{\beta}\big)$

Or on a $\widetilde{\beta} = \dfrac{\langle C, \mathbf{1}_2\rangle}{\langle \mathbf{1}_2, \Sigma\mathbf{1}_2\rangle}\mathbf{1}_2 \neq \Sigma^{-1}C = \beta^\star$ et on a vu que $\beta^\star$ est l'unique argument qui minimise la variance $\widehat{\delta}_n(\beta)$.

Donc l'estimateur $\widehat{\delta}_n(\beta^\star)$ est préférable en terme de variance à un estimateur basé sur la méthode de la variable de contrôle utilisant la fonction $h_0 : y \longmapsto h_{0,1}(y) + h_{0,2}(y)$.

17. (♠) Implémenter un code R qui fournit, pour $n = 1000$, l'estimation $\widehat{\delta}_n(\beta^\star)$, l'erreur quadratique moyenne de $\widehat{\delta}_n(\beta^\star)$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```
                              #### Estimation de delta n°5 ####

# Espérance de h_0_1(Y) calculée théoriquement
esp_h_0_1 <- function(mu = -13,b=2){
  return(mu + b/6)
}

# Espérance de h_0_2(Y) calculée théoriquement
esp_h_0_2 <- function(mu = -13, b=2, epsilon=0.6, t=0, alpha=3, theta=2){
  q <- qgamma(epsilon, shape = alpha, scale = theta)
  return(exp(-(t - q - mu)/b)/2)
}

# Calcul du vecteur C
vecteur_C <- function(Y){
  # on calcule la covariance entre h_tilde(Y) et h_0_1(Y)
  Cov_1 <- cov(h_tilde(Y),h_0_1(Y))
  # on calcule la covariance entre h_tilde(Y) et h_0_2(Y)
  Cov_2 <- cov(h_tilde(Y),h_0_2(Y))
  return(c(Cov_1,Cov_2))# on renvoie le vecteur C
}

# Estimation de delta (méthode avec beta*)
delta_beta <- function(n, Sigma){
  Y <- rgen_g(n) # on simule n v.a. suivant g
  m1 <- esp_h_0_1() # on calcule m1, l'espérance de h_0_1(Y) (calculée théoriquement)
  m2 <- esp_h_0_2() # on calcule m1, l'espérance de h_0_2(Y) (calculée théoriquement)
  # on crée le vecteur du produit scalaire avec beta*
  vect <- matrix(c(h_0_1(Y) - m1, h_0_2(Y) - m2),ncol = 2)
  beta <- solve(Sigma)%*%vecteur_C(Y) # on calcule beta*
  h <- h_tilde(Y) - vect%*%beta # on calcule h : Y |--> 1 - F(t-Y) - < beta* , vect >
  return(h) # on renvoie h
}

#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(delta_beta(n,Sigma))

##
## Estimateur                                    0.0701352661
## Erreur quadratique moyenne                    0.0007711597
## Intervalle de confiance    0.06841411 0.0718564200
```

Cette nouvelle estimation de δ a encore une fois réduit la variance et la taille de l'intervalle de confiance. On va maintenant chercher un dernier estimateur de δ de variance encore plus petite.

▷ Méthode Monte Carlo n°4

18. Soit $h \in L^2\big([0,1]\big)$. Montrer que pour une suite $\big(U_k\big)_{k\geq 1}$ de variables aléatoires i.i.d. de loi uniforme sur $[0,1]$, on a :

$$\mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h\left(\frac{k-1+U_k}{n}\right)\right] \leq \mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h(U_k)\right]$$

Soient $h \in L^2\big([0,1]\big)$ et $\big(U_k\big)_{k\geq 1}$ une suite de variables alétoires i.i.d. de loi uniformes sur $[0,1]$,

on a $\mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h(U_k)\right] = \frac{1}{n^2}\mathbb{V}\mathrm{ar}\left[\sum_{k=1}^{n} h(U_k)\right] = \frac{1}{n^2}\sum_{k=1}^{n}\mathbb{V}\mathrm{ar}\big[h(U_k)\big]$ car les $\big(U_k\big)_{k\geq 1}$ sont i.i.d.

Posons $D_k = \left[\frac{k-1}{n};\frac{k}{n}\right]$, pour tout $k \in [\![1,n]\!]$ et soit $Z \sim \mathscr{U}\big([0,1]\big)$,

alors pour tout $k \in [\![1,n]\!]$, on a $\mathbb{V}\mathrm{ar}\big[h(U_k)\big] = \sum_{i=1}^{n}\mathbb{P}\big[Z\in D_i\big]\mathbb{E}\Big[h(U_k)^2\Big|Z\in D_i\Big] - \left(\sum_{i=1}^{n}\mathbb{P}\big[Z\in D_i\big]\mathbb{E}\Big[h(U_k)\Big|Z\in D_i\Big]\right)^2$

or en posant :

▷ $\sigma_i^2 = \mathbb{V}\mathrm{ar}\Big[h(U_k)\Big|Z\in D_i\Big]$ ▷ $\mu_i = \mathbb{E}\Big[h(U_k)\Big|Z\in D_i\Big]$ ▷ $p_i = \mathbb{P}\big[Z\in D_i\big]$

on a pour tout $k \in [\![1,n]\!]$, $\mathbb{V}\mathrm{ar}\big[h(U_k)\big] = \underbrace{\sum_{i=1}^{n} p_i\sigma_i^2}_{\geq\left(\sum_{i=1}^{n} p_i\sigma_i\right)^2} + \underbrace{\sum_{i=1}^{n} p_i\mu_i^2 - \left(\sum_{i=1}^{n} p_i\mu_i\right)^2}_{\geq 0 \text{ d'après Jensen}}$

Donc pour tout $k \in [\![1,n]\!]$, $\mathbb{V}\mathrm{ar}\big[h(U_k)\big] \geq \left(\sum_{i=1}^{n} p_i\sigma_i\right)^2$

$$\begin{aligned}\text{D'où } \mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h(U_k)\right] &= \frac{1}{n^2}\sum_{k=1}^{n}\mathbb{V}\mathrm{ar}\big[h(U_k)\big] \geq \frac{1}{n^2}\sum_{k=1}^{n}\left(\sum_{i=1}^{n} p_i\sigma_i\right)^2 \\ &= \frac{1}{n^2}\sum_{k=1}^{n}\left(\sum_{i=1}^{n}\mathbb{P}\big[Z\in D_i\big]\sqrt{\mathbb{V}\mathrm{ar}\big[h(U_k)|Z\in D_i\big]}\right)^2\end{aligned}$$

Et d'après la *proposition 3.7* du cours, on a bien $\big(p_1,\ldots,p_n\big)$ comme allocation optimale et donc on a :

$$\begin{aligned}\mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h(U_k)\right] &\geq \frac{1}{n^2}\sum_{k=1}^{n}\mathbb{V}\mathrm{ar}\left[h\left(\frac{k-1+U_k}{n}\right)\right] \quad \text{car } \left(\frac{k-1+U_k}{n}\right)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}\Big(\big(D_k\big)_{k\geq 1}\Big) \\ &= \mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h\left(\frac{k-1+U_k}{n}\right)\right] \quad \text{car les } \big(U_k\big)_{k\geq 1} \text{ sont i.i.d.}\end{aligned}$$

On trouve donc bien que pour toute fonction $h \in L^2\big([0,1]\big)$ et pour toute suite $\big(U_k\big)_{k\geq 1}$ de variables aléatoires i.i.d. uniformes sur $[0,1]$:

$$\boxed{\mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h\left(\frac{k-1+U_k}{n}\right)\right] \leq \mathbb{V}\mathrm{ar}\left[\frac{1}{n}\sum_{k=1}^{n} h(U_k)\right]}$$

19. En déduire un nouvel estimateur $\widehat{\Delta}_n$ de δ. Comment peut-on interpréter cet estimateur?

D'après ce qui précède et comme la fonction $\widetilde{h} : y \longmapsto 1 - F(t-y)$ est dans $L^2([0,1])$, on en déduit l'estimateur suivant :

$$\widehat{\Delta}_n = \sum_{k=1}^{n} \frac{\mathbb{P}[Z \in D_k]}{n_k} \sum_{i=1}^{n_k} \widetilde{h}\left(G^{\leftarrow}\left(U_i^{(k)}\right)\right) \text{ avec } \left(U_i^{(k)}\right)_{i \geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}([0,1]) | Z \in D_k$$

Or on a pour tout $k \in [\![1,n]\!]$:

$\triangleright$ $n_k = 1$ $\triangleright$ $\mathbb{P}[Z \in D_k] = \mathbb{P}\left[\frac{k-1}{n} \leq Z \leq \frac{k}{n}\right] = \frac{1}{n}$ $\triangleright$ $U^{(k)} = \frac{k-1+U_k}{n}$ avec $U_k \sim \mathscr{U}([0,1])$

Donc on a $\boxed{\widehat{\Delta}_n = \frac{1}{n}\sum_{k=1}^{n} 1 - F\left(t - G^{\leftarrow}\left(\frac{k-1+U_k}{n}\right)\right)}$ avec $(U_k)_{k\geq 1} \overset{\text{i.i.d.}}{\sim} \mathscr{U}([0,1])$.

On obtient donc un estimateur stratifié avec :

- $\triangleright$ Z comme variable de stratification qui suit une loi uniforme sur $[0,1]$.
- $\triangleright$ Les strates sont les $D_k = \left[\frac{k-1}{n}; \frac{k}{n}\right]$ pour tout $k \in [\![1,n]\!]$.
- $\triangleright$ Enfin, l'allocation (proportionnelle) optimale est $\underbrace{(1,\ldots,1)}_{n \text{ fois}}$ car pour tout $k \in [\![1,n]\!]$, $n_k = 1$.

20. (♠) Implémenter un code R qui fournit, pour $n = 1000$, l'estimation $\widehat{\Delta}_n$, l'erreur quadratique moyenne de $\widehat{\Delta}_n$ et un intervalle de confiance asymptotique bilatéral pour δ au niveau 95%.

```
                           #### Estimation de delta n°6 ####
# Estimation de delta (méthode de stratification)
Delta_strat <- function(n){
  S <- 0 # on initialise la somme à 0
  for (k in 1:n){ # Pour i variant de 1 à n
    Y <- G_fleche((k-1+runif(n))/n) # on simule n v.a. suivant G_fleche((k-1+U)/n)
    S <- S + h_tilde(Y) # on ajoute à S h_tilde(G_fleche((k-1+U)/n))
  }
  # on renvoie la somme divisée par n
  # (car Delta_n = 1/n*Somme(h_tilde(G_fleche((k-1+U)/n))))
  return(S/n)
}
#On veut une estimation pour n = 1000
n <- 1000
# On applique la fonction MC pour renvoyer le tableau regroupant l'estimateur,
# son erreur quadratique moyenne et un intervalle de confiance de delta au
# niveau 95%
MC(Delta_strat(n))

##
## Estimateur                              6.974913e-02
## Erreur quadratique moyenne              2.772812e-09
## Intervalle de confiance    0.06974586 6.975239e-02
```

Avec cette dernière estimation de δ via la méthode de stratification, on a drastriquement réduit la variance $\left(\text{passant de l'ordre de } 10^{-4} \text{ pour la méthode précédente à } 10^{-9}\right)$.
On va maintenant comparer l'efficacité relative de tous les estimateurs calculés.

21. (♠) Comparer l'efficacité relative des différents estimateurs de δ que vous avez proposés.

```
                    #### Comparaison de la vitesse d'exécution ####
# Import de la librairie Microbenchmark
library(microbenchmark)
# On va simuler n = 100 v.a. pour chaque estimation et comparer le temps
d'exécution
n <- 100
# On compare les temps de calculs
comparaison <- microbenchmark(p_n(n), delta_n(n), delta_A_n(n), delta_cont(n),
                              delta_beta(n,Sigma), Delta_strat(n))
print(comparaison, signif= 3)

## Unit: microseconds
##                  expr    min     lq   mean median     uq   max neval cld
##                p_n(n)   25.8   31.3   36.3   35.9   40.2    66   100   a
##            delta_n(n)   25.5   28.8   55.5   30.5   32.3  2430   100   a
##          delta_A_n(n)   40.2   44.6   87.0   47.0   49.4  4030   100   a
##         delta_cont(n)   78.3   85.8  143.0   93.0  111.0  4510   100   a
##   delta_beta(n, Sigma)  140.0  163.0  323.0  188.0  215.0 13400   100   a
##        Delta_strat(n) 2540.0 2910.0 3460.0 2980.0 3160.0 49600   100   b
```

On remarque (dans la colonne `mean`) que les 6 estimateurs mettent en moyenne tous à peu près le même temps pour s'exécuter (même si le temps augmente au fur et à mesure que l'estimateur se complexifie) sauf pour l'estimateur stratifiée qui est 100 fois plus long que l'estimateur de Monte Carlo classique $\widehat{p}_n$. De plus, pour chaque estimateur, on simule $n = 1000$ variables aléatoires hormis pour l'estimateur stratifié $\widehat{\Delta}_n$ pour qui on en simule 10^6.

Donc pour les 5 premiers estimateurs une comparaison en terme de variance suffit car nous allons voir que les variances ont une différence bien plus significative comparée aux coûts de calcul. Cependant, pour l'estimateur stratifié $\widehat{\Delta}_n$ il faut rajouter un facteur de coût d'environ :

- × 100 par rapport à $\widehat{p}_n$
- × 60 par rapport à $\widehat{\delta}_n$
- × 40 par rapport à $\widehat{\delta}_n^A$
- × 25 par rapport à $\widehat{\delta}_n^{\text{cont}}$
- × 10 par rapport à $\widehat{\delta}_n(\beta^\star)$

Regardons maintenant les rapports des variances des estimateurs :

```
                    #### Comparaison des variances deux à deux ####
# On va simuler n = 1000 v.a. pour chaque estimation et comparer les variances
n <- 1000
```

```
# Vecteur des variances des estimateurs
VAR <- c(var(p_n(n)), var(delta_n(n)),var(delta_A_n(n)),var(delta_cont(n)),
          var(delta_beta(n,Sigma)),var(Delta_strat(n)))
# on crée une matrice (ou data.frame) pour comparer les matrices deux à deux
# cette matrice fait le rapport des variances deux à deux
Tb_var <- data.frame(outer(VAR,VAR,"/"))
# on renomme les colonnes du data.frame
    names(Tb_var) <- c("p_n","delta_n", "delta_A_n", "delta_cont", "delta_beta",
                        "Delta_strat")
# Affichage du tableau
Tb_var

##              p_n      delta_n    delta_A_n   delta_cont   delta_beta Delta_strat
## 1 1.000000e+00 1.033212e+01 2.585571e+01 3.627843e+01 8.543960e+01  33015167.7
## 2 9.678560e-02 1.000000e+00 2.502460e+00 3.511230e+00 8.269323e+00   3195392.8
## 3 3.867618e-02 3.996067e-01 1.000000e+00 1.403111e+00 3.304477e+00   1276900.5
## 4 2.756459e-02 2.848005e-01 7.127019e-01 1.000000e+00 2.355107e+00    910049.5
## 5 1.170417e-02 1.209289e-01 3.026197e-01 4.246091e-01 1.000000e+00    386415.3
## 6 3.028911e-08 3.129506e-07 7.831464e-07 1.098841e-06 2.587889e-06         1.0
```

Comme on peut le voir en regardant la partie triangulaire inférieure de la matrice, les variances des estimateurs de δ ont sans cesse diminué au fur et à mesure que l'on a créé de nouveaux estimateurs. Ce qui avait déjà été prouvé en théorie est donc bien vérifié :

$$\mathbb{V}\mathrm{ar}[\widehat{p}_n] \geq \mathbb{V}\mathrm{ar}[\widehat{\delta}_n] \geq \mathbb{V}\mathrm{ar}[\widehat{\delta}_n^A] \geq \mathbb{V}\mathrm{ar}[\widehat{\delta}_n^{\mathrm{cont}}] \geq \mathbb{V}\mathrm{ar}[\widehat{\delta}_n(\beta^\star)] \geq \mathbb{V}\mathrm{ar}[\widehat{\Delta}_n]$$

Le meilleur estimateur en terme de variance est l'estimateur stratifié $\widehat{\Delta}_n$ mais celui-ci a un coût de calcul 100 fois plus important que celui de l'estimateur Monte Carlo classique $\widehat{p}_n$.
Cependant, si on calcule l'efficacité relative de $\widehat{\Delta}_n$ par rapport à n'importe quel autre estimateur, on trouve que :
$\rho(\widehat{\Delta}_n, \widetilde{\delta}_n) = \dfrac{C \times \sigma^2_{\widehat{\Delta}_n}}{\sigma^2_{\widetilde{\delta}_n}} < 1$ avec $\widetilde{\delta}_n$ un estimateur de δ parmi $\widehat{p}_n$, $\widehat{\delta}_n$, $\widehat{\delta}_n^A$, $\widehat{\delta}_n^{\mathrm{cont}}$, et $\widehat{\delta}_n(\beta^\star)$ et C le facteur de coût.

En effet, il suffit de multiplié par les facteurs de coût la dernière ligne du tableau de rapport des variances et on trouve bien le résultat ci-dessus :

```
# Efficacités relatives de Delta_n par rapport aux autres estimateurs
facteurs_cout <- c(100,60,40,25,10)#facteurs de coûts par rapport aux autres estimateurs
Tb_var[6,1:5]*facteurs_cout # dernière ligne multipliée par les facteurs de coût

##              p_n      delta_n    delta_A_n   delta_cont   delta_beta
## 6 3.028911e-06 1.877703e-05 3.132585e-05 2.747103e-05 2.587889e-05
```

Donc même si le coût de calcul de $\widehat{\Delta}_n$ est plus important, l'efficacité relative de cet estimateur est meilleure que celle des autres.

☐ Exercice 2

Soient $m \in \mathbb{N}^*$ et $k \in [\![2, m-1]\!]$. Avec la densité g définie à l'Exercice 1 et G la fonction de répartition associée à g, on définit :

$$\Psi(x) = \frac{m!}{(k-1)!(m-k)!} G(x)^{k-1}\big(1-G(x)\big)^{m-k} g(x)$$

1. (♠) Implémenter un algorithme du rejet qui simule n réalisations suivant la densité Ψ.

On choisit naturellement g comme densité instrumentale.

On a pour tout $x > \mu - 2b$, $\dfrac{\Psi(x)}{g(x)} = \dfrac{m!}{(k-1)!(m-k)!} G(x)^{k-1}\big(1-G(x)\big)^{m-k}$

Or $]\mu-2b, +\infty[$ est ouvert dans $\mathbb{R}$ donc la condition nécessaire d'optimimalité est : $\dfrac{\partial}{\partial x}\left(\dfrac{\Psi(x)}{g(x)}\right) = 0$

$$\iff \frac{m!}{(k-1)!(m-k)!}\left[(k-1)g(x)G(x)^{k-2} \times \big(1-G(x)\big)^{m-k} - G(x)^{k-1} \times (m-k)g(x)\big(1-G(x)\big)^{m-k-1}\right] = 0$$

$$\iff g(x)G(x)^{k-1}\big(1-G(x)\big)^{m-k}\left[\frac{(k-1)}{G(x)} - \frac{(m-k)}{1-G(x)}\right] = 0$$

$$\iff \frac{(k-1)}{G(x)} = \frac{(m-k)}{1-G(x)} \quad \text{car pour tout } x > \mu-2b,\ 0 < G(x) < 1 \text{ et } g(x) \neq 0$$

$$\iff \frac{G(x)(m-k)}{(k-1)(m-k)} - \frac{\big(1-G(x)\big)(k-1)}{(m-k)(k-1)} = 0$$

$$\iff G(x) = \frac{k-1}{m-1}$$

$$\iff \boxed{x^* = G^{\leftarrow}\left(\frac{k-1}{m-1}\right)}$$

Donc pour tout $x > \mu - 2b$, $\dfrac{\Psi(x)}{g(x)} \leq \dfrac{m!}{(k-1)!(m-k)!} G\left(G^{\leftarrow}\left(\frac{k-1}{m-1}\right)\right)^{k-1}\left(1 - G\left(G^{\leftarrow}\left(\frac{k-1}{m-1}\right)\right)\right)^{m-k}$

i.e. pour tout $x \in]0,1[$, $\boxed{\dfrac{\Psi(x)}{g(x)} \leq \dfrac{m!}{(k-1)!(m-k)!}\left(\dfrac{k-1}{m-1}\right)^{k-1}\left(\dfrac{m-k}{m-1}\right)^{m-k} =: M(k,m)}$ car $G^{\leftarrow} = G^{-1}$.

Algorithme du rejet :

Pour obtenir n variables aléatoires suivant la densité Ψ, on répète n fois le schéma suivant :

- ⊳ on simule $Y \sim g$ (g étant la densité instrumentale)
- ⊳ on simule $U \sim \mathcal{U}([0,1])$

↪ jusqu'à ce que $U < \dfrac{\Psi(Y)}{M(k,m)g(Y)} = \left(\dfrac{m-1}{k-1}G(Y)\right)^{k-1}\left(\dfrac{m-1}{m-k}\big(1-G(Y)\big)\right)^{m-k}$

```r
                              #### Algorithme du rejet ####

# Densité Psi
densite_psi <- function(x,m,k){
  return(factorial(m)/(factorial(k-1)*factorial(m-k))
         *fdr_G(x)^(k-1)*(1-fdr_G(x))^(m-k)*densite_g(x))
}

# Probabilité d'acceptation 1/M calculée théoriquement
p_accept <- function(m, k){
  M <- (factorial(m)/(factorial(k-1)*factorial(m-k))
        *((k-1)/(m-1))^(k-1)*((m-k)/(m-1))^(m-k))
  return(1/M)
}

# Algorithme du rejet pour simuler n variables aléatoire suivant Psi
rejet_Psi <- function(n = 1000, m, k){
  res <- c() # vecteur résultat à remplir
  p <- p_accept(m,k) #probabilité d'acceptation calculée précédemment
  compteur <- 0 # compteur du nombre de variables aléatoires simulées
  while (length(res) < n){# tant que le vecteur résultat n'est pas de taille n,
                          # on réitère le schéma suivant :

    # on rajoute au compteur le nombre de v.a. simulées dans la boucle
    compteur <- compteur+(n - length(res))%/%p +1

    # on simule (n - length(res)) %/% p + 1 v.a. suivant la densité g
    Y <- rgen_g((n - length(res)) %/% p +1)

    # on simule (n - length(res)) %/% p + 1 v.a. uniformes sur [0,1]
    U <- runif((n - length(res))%/%p +1)

    # On récupère uniquement les éléments du vecteur Y qui vérifie U < Psi(Y)/(M*g(Y))
    res <- c(res,Y[U < p* densite_psi(Y,m,k)/(densite_g(Y))])
  }
  # On retourne le résultat
  return(res[1:n])
# NB : pour renvoyer la probabilité d'acceptation de l'algorithme, on return(n/compteur)
}
```

2. (♠) Vérifier graphiquement que votre code est valide pour $m = 15$ et $k = 7$.

```
#### Vérification graphique ####
# Package de couleurs
library(viridisLite)
c_ <- viridis(5)

# On va simuler 10000 v.a. suivant Psi
n <- 10000
# Paramètres m et k données
m <- 15
k <- 7
# Application de l'algorithme de rejet
x <- rejet_Psi(n,m,k)

# Histogramme de l'échantillon
hist(x, freq = FALSE,main = "Distribution de l'échantillon", ylab = "Fréquences",
     ylim = c(0,1), breaks = 30, col = "grey70", border = "grey40")
# Supperposition de la courbe de la densité Psi
lines(y <- seq(min(x),max(x),0.001), densite_psi(y,m,k), col = c_[4], lwd = 4)
# Légende
legend("topright", "Densité Psi", col = c_[4], lwd = 4, box.lty = 1, bg = "Grey95",
       inset = 0.005, box.col = "grey50")
```

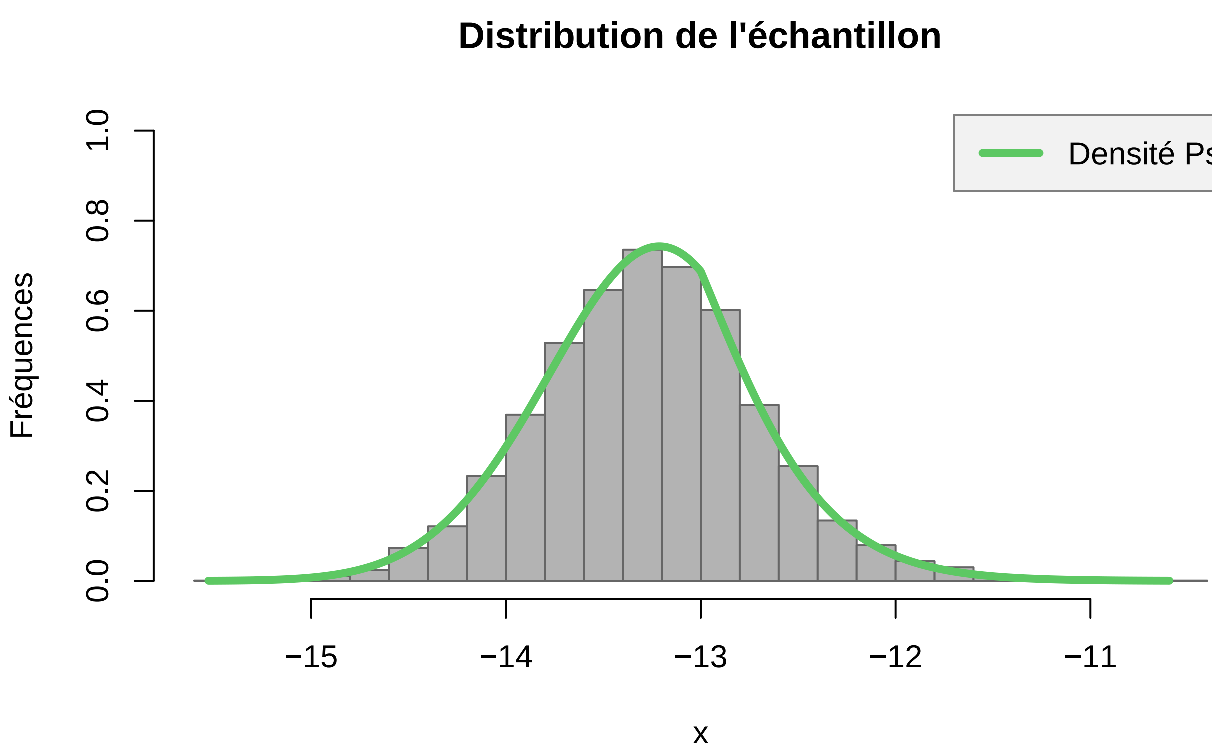

FIGURE 3 – Vérification graphique de l'algorithme du rejet

La distribution de l'échantillon et la courbe de la densité Ψ se superposent parfaitement, on a bien simulé la densité Ψ grâce à l'algorithme du rejet.

Probabilité d'acceptation théorique : $\dfrac{(k-1)!(m-k)!}{m!} \times \left(\dfrac{m-1}{k-1}\right)^{k-1} \times \left(\dfrac{m-1}{m-k}\right)^{m-k} \simeq 0,3151509$

Nombre total de variables aléatoires simulées : $\simeq 317000$

Probabilité d'acceptation de votre algorithme : $\simeq 0,31514$

Renseigner dans la grille suivante le nombre de fois où les différents éléments cités sont utilisés dans votre code.

`c()`	`for`	`while`	`if-else`	`Vectorize`	`apply`
1	0	1	0	0	0