

# Computational statistics

## Week 3

Léa Bresson

### Exercices:

- Exercice 7.1 - R
- Exercice 8.8 - R
- Exercice 9.2 - R
- Exercice 9.18
- Exercice 10.25
- Exercice 6.51

## 1 Exercice 7.1 - R

Calculate the mean of a  $\text{Gamma}(4.3, 6.2)$  random variable using a), b), c) or d). In each case monitor the convergence.

a) Accept-Reject with a  $\text{Gamma}(4, 7)$  candidate

### Solution:

Dans cet exercice, on note  $f$  la fonction de densité de la loi  $\Gamma(4.3, 6.2)$  et  $g$  la fonction de densité candidate. Par ailleurs, pour chaque question, nous effectuerons 10.000 simulations.

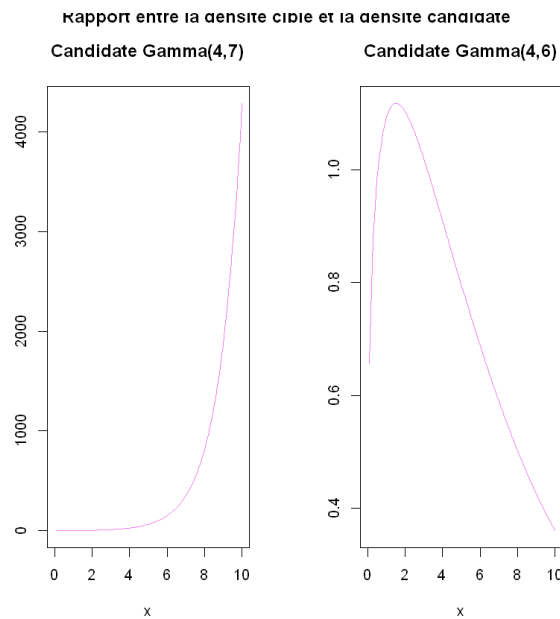
Nous ne pouvons pas choisir comme densité candidate la loi  $\Gamma(4, 7)$  pour générer des réalisations de la loi  $\Gamma(4.3, 6.2)$  car le rapport entre les densités de ces deux lois ( $\frac{f}{g}$ ) n'est pas borné. Par conséquent, nous choisissons comme densité candidate la loi  $\Gamma(4, 6)$  (cf. dropbox, "fichier erratum"). Dans ce cas de figure, le rapport entre la densité cible et la densité candidate est borné ce qui nous autorise à implémenter l'algorithme AR.

Les figures ci-dessous représentent le rapport  $\frac{f}{g}$  et la nécessité de choisir une autre distribution que la  $\Gamma(4, 7)$ .

```

In [1]: rapport1 = function(x){
          dgamma(x,4.3,6.2)/dgamma(x,4,7)
        }
        rapport = function(x){
          dgamma(x,4.3,6.2)/dgamma(x,4,6)
        }
        par(mfrow = c(1, 2))
        curve(rapport1(x), col = "violet", xlim = c(0, 10), main = "Candidate Gamma(4,7)",
              ylab = "")
        title(main="Rapport entre la densité cible et la densité candidate", outer=T, line=-0.5)
        curve(rapport(x), col = "violet", xlim = c(0, 10), main = "Candidate Gamma(4,6)",
              ylab = "")

```



Dans un premier temps, il convient de déterminer la valeur  $\sup_g \frac{f}{g}$ . Pour ce faire nous utilisons la fonction R "Optimize". Par la suite, nous implémentons l'algorithme AR.

```

In [2]: M = optimize(rapport, interval=c(0,100), maximum=T)$objective
        M

```

1.11728544570492

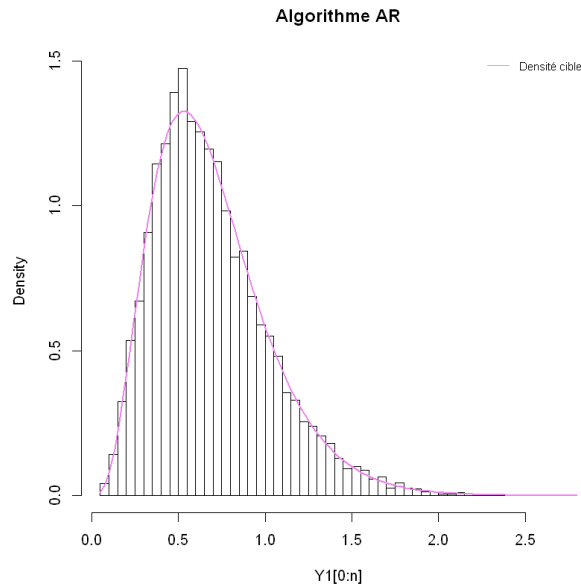
```

In [3]: # Initialisation
        n = 10000

        # Algorithme accept-Reject
        g = rgamma(n,4,6)
        u = runif(n, max=M*dgamma(g,4,6))
        Y1 = g[u < dgamma(g,4.3,6.2)]

```

```
In [4]: hist(Y1[0:n], main = "Algorithme AR", freq = F, breaks = 50)
        curve(dgamma(x,4.3,6.2), col = "violet", add = T, lwd = 2)
        legend("topright", legend = c("Densité cible"), lty=1, col=c("violet"), bty = "n", cex=.75)
```



L'histogramme ci-dessus indique que l'algorithme AR implémenté est correct. Nous nous intéressons donc désormais à la convergence de la moyenne empirique.

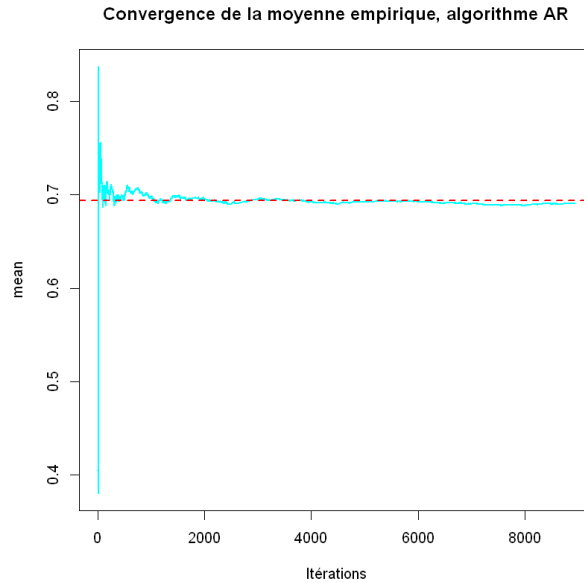
Dans un premier temps, soulignons que la moyenne empirique (0.6906) est très proche de l'espérance de la loi cible (0.6935) et donc de la vraie valeur de la moyenne (via la loi forte des grands nombres).

```
In [5]: mean(Y1)
        m_cible = 4.3/6.2
        m_cible
```

0.690648056775266

0.693548387096774

```
In [6]: plot(cumsum(Y1)/(1:length(Y1)), col="cyan", type="l", lwd=2,xlab="Itérations", ylab="mean",
            main="Convergence de la moyenne empirique, algorithme AR")
        abline(h=m_cible, lty=2, lwd=2, col=2)
```



Pour conclure, l'algorithme AR donne de bons résultats: la moyenne estimée converge rapidement vers la moyenne observée.

#### b) Metropolis–Hastings with a Gamma(4, 7) candidate

Nous implémentons désormais l'algorithme de Metropolis-Hastings. La distribution candidate est toujours la loi  $\Gamma(4, 7)$ .

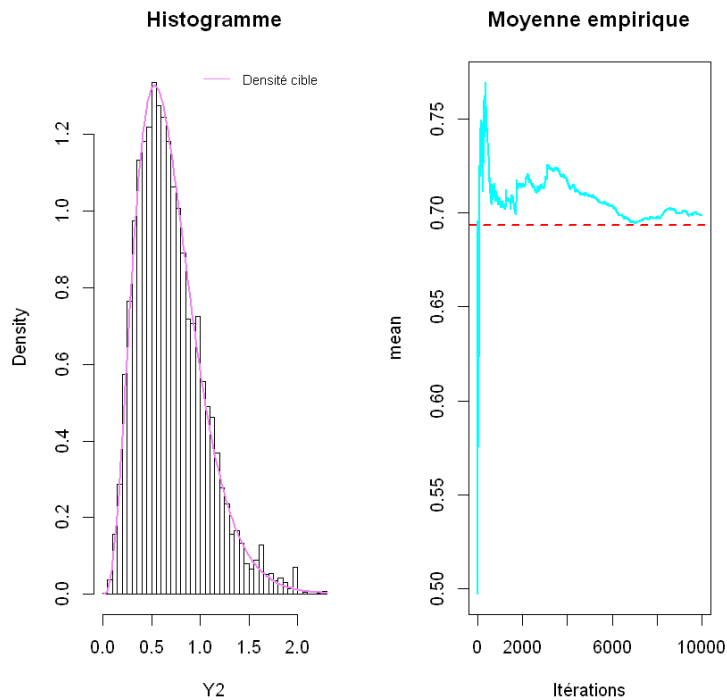
```
In [7]: Y2 = c()
        Y2[1] = rgamma(1, 4.3, 6.2)
        g = rgamma(n, 4, 7)
        for(i in 2:n){
            rho = min(dgamma(g[i-1], 4.3, 6.2) * dgamma(Y2[i-1], 4, 7) / (
                dgamma(Y2[i-1], 4.3, 6.2) * dgamma(g[i-1], 4, 7)), 1)
            if(runif(1)<rho) Y2[i] = g[i-1] else Y2[i] = Y2[i-1]
        }

        mean(Y2)
        m_cible
```

0.69894507939276

0.693548387096774

```
In [9]: par(mfrow = c(1, 2))
        hist(Y2, main = "Histogramme", freq = F, breaks = 50)
        curve(dgamma(x,4.3,6.2), col = "violet", add = T, lwd = 2)
        legend("topright", legend = c("Densité cible"), lty=1, col=c("violet"), bty = "n", cex=.75)
        plot(cumsum(Y2)/(1:length(Y2)), col="cyan", type="l", lwd=2, xlab="Itérations", ylab="mean",
            main="Moyenne empirique")
        abline(h=m_cible, lty=2, lwd=2, col=2)
```



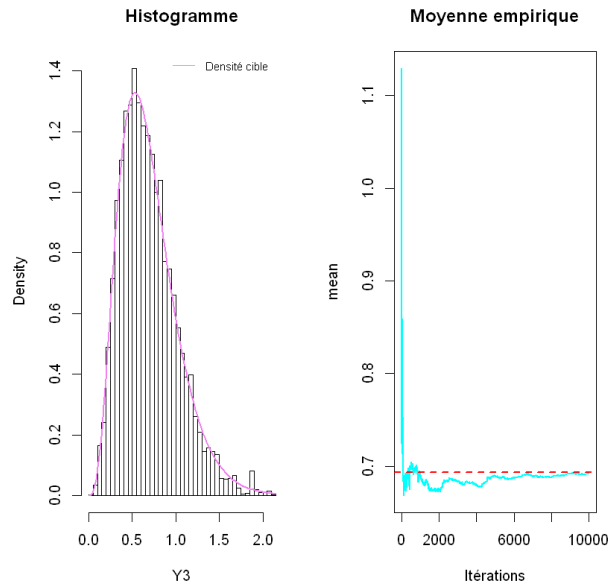
Il semble que la densité cible soit légèrement moins bien approchée que dans le cas précédent (cf. histogramme ci-dessus); néanmoins la performance de l'algorithme reste très bonne. La moyenne empirique est très proche de la "vraie" moyenne (0.6989 contre 0.6935) et converge vers celle-ci. Notons que la vitesse de convergence est plus faible qu'avec l'algorithme AR.

### c) Metropolis–Hastings with a Gamma(5, 6) candidate

Nous utilisons cette fois-ci la distribution  $\Gamma(5,6)$  comme distribution candidate.

```
In [10]: Y3 = c()
         Y3[1] = rgamma(1, 4.3, 6.2)
         g = rgamma(n, 4, 7)
         for(i in 2:n){
             rho = min(dgamma(g[i-1], 4.3, 6.2) * dgamma(Y3[i-1], 4, 7) / (dgamma(Y3[i-1], 4.3, 6.2) * dgamma(g[i-1], 4, 7)))
             if(runif(1)<rho) Y3[i] = g[i-1] else Y3[i] = Y3[i-1]
         }
```

```
In [11]: par(mfrow = c(1, 2))
         hist(Y3, main = "Histogramme", freq = F, breaks = 50)
         curve(dgamma(x,4.3,6.2), col = "violet", add = T, lwd = 2)
         legend("topright", legend = c("Densité cible"), lty=1, col=c("violet"), bty = "n", cex=.75)
         plot(cumsum(Y3)/(1:length(Y3)), col="cyan", type="l", lwd=2, xlab="Itérations",
              ylab="mean", main="Moyenne empirique")
         abline(h=m_cible, lty=2, lwd=2, col=2)
```



```
In [12]: mean(Y3)
         m_cible
```

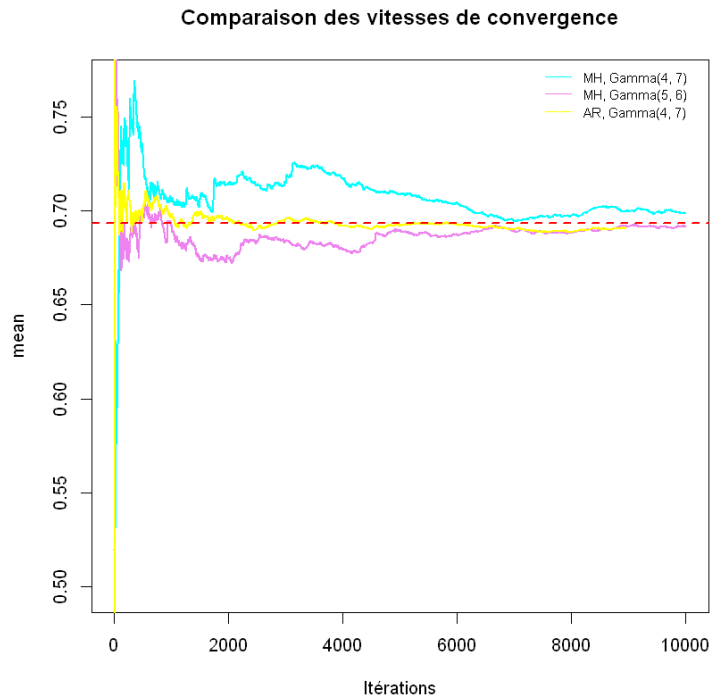
0.691608565520768

0.693548387096774

Nous pouvons dresser les mêmes conclusions que précédemment.

Comparaison des 3 algorithmes:

```
In [13]: plot(cumsum(Y2)/(1:length(Y2)), col="cyan", type="l", lwd=2, xlab="Itérations",
             ylab="mean", main="Comparaison des vitesses de convergence")
lines(cumsum(Y3)/(1:length(Y3)), col="violet", type="l", lwd=2, xlab="Itérations", ylab="mean")
lines(cumsum(Y1)/(1:length(Y1)), col="yellow", type="l", lwd=2, xlab="Itérations", ylab="mean")
abline(h=m_cible, lty=2, lwd=2, col=2)
legend("topright", legend = c("MH, Gamma(4, 7)", "MH, Gamma(5, 6)", "AR, Gamma(4, 7)"),
       lty=1, col=c("cyan", "violet", "yellow"), bty = "n", cex=.75)
```



```
In [14]: precision = c(abs(mean(Y1)-m_cible),abs(mean(Y2)-m_cible), abs(mean(Y3)-m_cible))
precision
```

1. 0.00290033032150772
2. 0.00539669229598538
3. 0.00193982157600636

Pour conclure, l'algorithme AR est celui dont la moyenne empirique converge le plus rapidement.  
Les trois algorithmes fournissent de bonnes estimations de la moyenne.

## 2 Exercice 8.8 (Neal 2003) - R

Consider the following hierarchical model, ( $i=1,\dots,10$ ):

$$x_i | \nu \sim N(0, e^\nu), \nu \sim N(0, 3^2)$$

a) When simulating this distribution from an independent Metropolis-Hastings algorithm with  $N(0,1)$  proposals on all variables, show that the algorithm fails to recover the proper distribution for  $\nu$  by looking at the smaller values of  $\nu$ .

**Solution:**

Pour chaque question, nous effectuons 10.000 simulations et retenons 1000 simulations en tant que "burn in" (cela nous permet, notamment, de minimiser l'effet du choix des valeurs initiales).

Dans cet exercice, la distribution cible est:

$$f(\nu, x_1, \dots, x_{10}) = \prod_{i=1}^{10} f(x_i|\nu) f(\nu) = \left( \frac{1}{\sqrt{2\pi e^\nu}} \right)^{10} e^{-\frac{1}{2e^\nu} \sum_{i=1}^{10} x_i^2} \times \frac{1}{3\sqrt{2\pi}} e^{-\frac{\nu^2}{2 \times 3^2}}$$

La distribution candidate considérée s'écrit:

$$q(\nu, x_1, \dots, x_{10}) = \prod_{i=1}^{10} N_{x_i}(0, 1) N_\nu(0, 1)$$

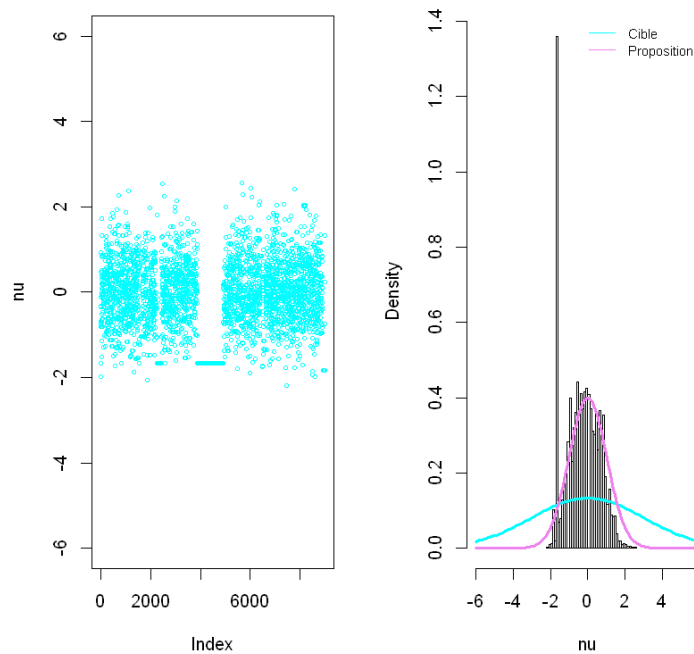
**Implémentation de l'algorithme:**

```
In [59]: library(mixtools) # pour utiliser dmnorm
# cible
cible = function(nu, para){
  dnorm(nu, 0, 3) * prod(dnorm(para, 0, sqrt(exp(nu))))
}
# proposition
proposition = function(nu, para){
  dnorm(nu) * dmnorm(para)
}
n = 10000
burn_in = 1000
nu_sim = rep(0, n + 1)
para_sim = matrix(rep(1, 10 * (n + 1)), nrow = n + 1, ncol = 10, byrow = F)
nu = c()
for (i in 1:n){
  nu[i] = rnorm(1)
  para = replicate(10, rnorm(1))
  rho[i] = min( cible(nu[i], para) * proposition(nu_sim[i], para_sim[i, ])
               / (cible(nu_sim[i], para_sim[i, ]) * proposition(nu[i], para)) , 1)
  if(runif(1)<rho[i]) nu_sim[i + 1] = nu[i] else nu_sim[i + 1] = nu_sim[i]
  if(runif(1)<rho[i]) para_sim[i + 1, ] = para else para_sim[i + 1, ] = para_sim[i, ]
}
```

```
In [62]: par(mfrow = c(1, 2))
plot(nu_sim[burn_in:n], type = "p", main= "", col = "cyan", ylim = c(-6, 6), ylab = "nu", cex=0.5)
title(main="Algorithme MH - N(0,1)", outer=TRUE, line=-2)
hist(nu_sim[burn_in:n], main = "", freq = F, xlim = c(-6, 6), breaks = 50, xlab = "nu")
curve(dnorm(x, sd = 3), col = "cyan", add = T, lwd = 3)
curve(dnorm(x), col = "violet", add = T, lwd = 3)
legend("topright", legend = c("Cible", "Proposition"), lty=1, col=c("cyan", "violet"), bty = "n", cex=
```



Algorithme MH -  $N(0,1)$



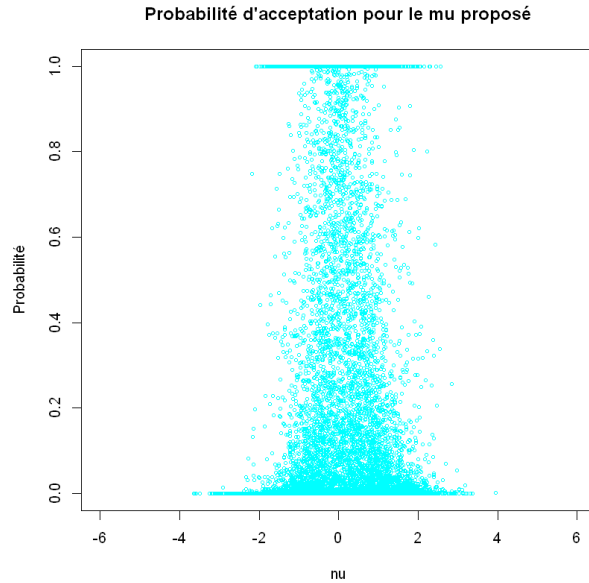
Les graphiques ci-dessus indiquent que les valeurs simulées de  $\nu$  se situent principalement dans l'intervalle  $[-2,2]$ ; ceci s'explique par le choix d'une loi normale centrée réduite comme densité proposée. L'histogramme souligne le fait que l'algorithme ne simule pas correctement les valeurs de  $\nu$  en particulier lorsqu'elles sont faibles.

Plus précisément, les valeurs comprises dans l'intervalle  $[-2,-4]$  sont simulées mais rejetées, tandis que les valeurs plus extrêmes ne sont même pas simulées (cf. graphe de la question suivante).

**b) Explain why this poor behavior occurs by considering the acceptance probability when  $\nu$  is smaller than -5.**

Le graphe ci-dessous renseigne la probabilité d'acceptation en fonction des valeurs simulées de  $\nu$ .

```
In [61]: plot(nu, rho, main= "Probabilité d'acceptation pour le mu proposé", col = "cyan",
             xlim = c(-6, 6), ylab = "Probabilité", xlab = "nu" , cex=0.5)
```



Nous constatons que la probabilité d'acceptation est extrêmement faible, voire nulle, pour les valeurs de  $\nu$  comprises dans les intervalles  $[-4, -2]$  et  $[2, 4]$ . De plus, les valeurs qui ne sont pas dans l'intervalle  $[-4, 4]$  ne sont pas simulées (ceci est lié au choix d'une  $N(0,1)$  comme densité de proposition).

L'écart type de la densité candidate est trop faible: pour pallier ce problème, nous considérons désormais des lois normales de variance 1 mais centrées sur la valeur observée.

- c. **Evaluate the performance of the corresponding Metropolis-Hastings which updates one component at a time with a Gaussian proposal centred at the current value and variance equal to one. Explain why the problems now occurs for the larger values of  $\nu$ .**

```

In [63]: # MH algo - N(current value, 1)
nu_sim = rep(0, n + 1)
para_sim = matrix(rep(1, 10 * (n + 1)), nrow=n + 1, ncol=10, byrow=F)
for (i in 1:n) {
  nu = rnorm(1, mean = nu_sim[i])
  rho = min(cible(nu, para_sim[i, ]) * dnorm(nu_sim[i], mean = nu)
            / (cible(nu_sim[i], para_sim[i,]) * dnorm(nu, mean = nu_sim[i])) , 1)
  if (runif(1) < rho) nu_sim[i + 1] = nu else nu_sim[i + 1] = nu_sim[i]
  for (j in 1:9) {
    y = para_sim[i, j]
    y[j] = rnorm(1, mean = para_sim[i, j])
    rho = min(cible(nu_sim[i + 1], y) * dnorm(para_sim[i, j], mean = y[j])
              / (cible(nu_sim[i + 1], para_sim[i,]) * dnorm(y[j], mean = para_sim[i, j])) , 1)
    if (runif(1) < rho) para_sim[i, j] = y else para_sim[i, j] = para_sim[i, j]
  }
  y = para_sim[i,]
  y[10] = rnorm(1, mean = para_sim[i, 10])
  rho = min(cible(nu_sim[i + 1], y) * dnorm(para_sim[i, 10], mean = y[10])
            / (cible(nu_sim[i + 1], para_sim[i,]) * dnorm(y[10], mean = para_sim[i, 10])) , 1)
  if (runif(1) < rho) para_sim[i + 1, ] = y else para_sim[i + 1, ] = para_sim[i, ]
}

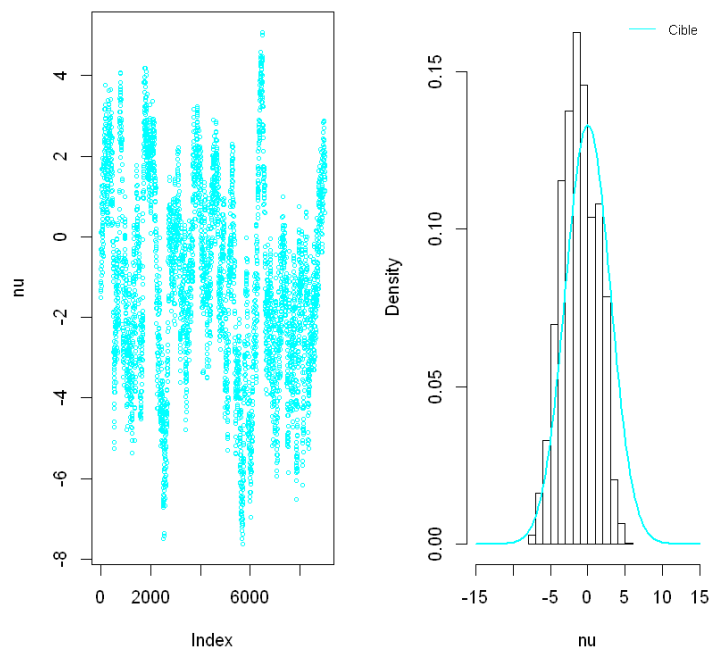
```

```

In [67]: par(mfrow = c(1, 2))
plot(nu_sim[burn_in:n], type = "p", main= "", col = "cyan", ylab = "nu", cex=0.5)
title(main="Algorithme MH, N(current value, 1)", outer=TRUE, line=-2)
hist(nu_sim[burn_in:n], main = "", freq = F, xlab = "nu", xlim = c(-15,15))
curve(dnorm(x, sd = 3), col = "cyan", add = T, lwd = 2)
legend("topright", legend = c("Cible"), lty=1, col=c("cyan"), bty = "n", cex=.75)

```

Algorithme MH, N(current value, 1)



Avec cette méthode les estimations sont plus proches de la densité cible.

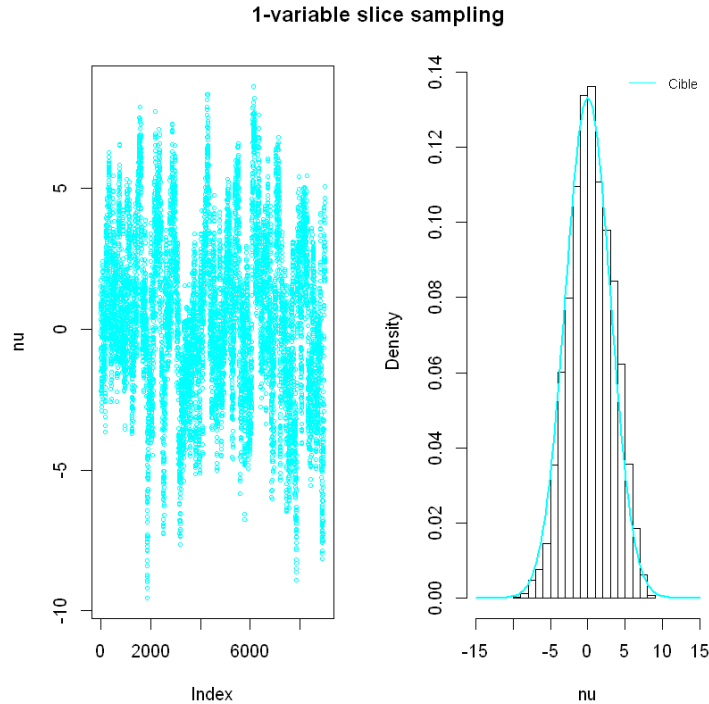
Par ailleurs, les valeurs de  $\nu$  ne sont plus restreintes à l'intervalle  $[-4;4]$ . En particulier, de faibles valeurs de  $\nu$  (i.e. inférieures à -5) sont désormais simulées. Ce n'est pas le cas pour les fortes valeurs de  $\nu$  (i.e. supérieures à 5). Ceci est liée à une mauvais ajustement de la queue droite de la distribution (cf. histogramme).

Pour conclure cette méthode pallie le problème d'un trop faible taux d'acceptation mais bute sur l'estimation de la queue droite de la distribution.

- d. Compare with a single-variable slice sampling, that is, slice sampling applied to the eleven full conditional distributions of  $\nu$  given the  $\xi_i$ 's and of the  $\xi_i$ 's given the  $\xi_j$ 's and  $\nu$ .**

```
In [69]: full.nu = function(nu, para) {
  dnorm(nu, 0, 3) * prod(dnorm(para, 0, sqrt(exp(nu))))
}
full.x = function(x, nu) {
  exp(-0.5/exp(nu) * x^2)
}
full.x = Vectorize(full.x, "x")
A0 = function(u, nu_sim, para) {
  c(uniroot(function(x) full.nu(x, para = para) - u, c(-40, nu_sim))$root,
    uniroot(function(x) full.nu(x, para = para) - u, c(nu_sim, 40))$root)
}
para_sim = matrix(1, ncol = 10, nrow = n + 1)
a_sim = matrix(0.5, ncol = 10, nrow = n + 1)
nu_sim = rep(1, n + 1)
u_sim = rep(0.1, n + 1)
for (i in 1:n) {
  u_sim[i + 1] = runif(1, min = 0, max = full.nu(nu_sim[i], para_sim[i,]))
  A1 = A0(u_sim[i + 1], nu_sim[i], para_sim[i, ])
  nu_sim[i + 1] = runif(1, min = A1[1], max = A1[2])
  for (j in 1:10) {
    a_sim[i + 1, j] = runif(1, min = 0, max = full.x(para_sim[i, j], nu_sim[i + 1]))
    para_sim[i + 1, j] = runif(1, min = -sqrt(-2 * exp(nu_sim[i + 1]) *
      log(a_sim[i + 1, j])), max = sqrt(-2 * exp(nu_sim[i + 1]) * log(a_sim[i + 1, j])))
  }
}
```

```
In [73]: par(mfrow = c(1, 2))
plot(nu_sim[burn_in:n], type = "p", main = "", col = "cyan", ylab = "nu", cex=0.5)
title(main="1-variable slice sampling", outer=TRUE, line=-2)
hist(nu_sim[burn_in:n], main = "", freq = F, xlab = "nu", xlim = c(-15,15))
curve(dnorm(x, sd = 3), col = "cyan", add = T, lwd = 2)
legend("topright", legend = c("Cible"), lty=1, col=c("cyan"), bty = "n", cex=.75)
```



Ce dernier algorithme offre les meilleurs résultats: l'approximation de la queue droite de distribution est meilleure. Les valeurs "extrêmes" de  $\nu$  sont désormais simulées.

### 3 Exercice 9.2 - R

Write a Gibbs sampler to generate standard bivariate normal random variables (with mean 0, variance 1 and correlation  $\rho$ ). (Recall that if  $(X, Y)$  is standard bivariate normal, the conditional density of  $X|Y = y$  is  $N(\rho y, (1 - \rho^2))$ ). For  $\rho = .3$ , use the generated random variables to estimate the density of  $X^2 + Y^2$  and calculate  $P(X^2 + Y^2 > 2)$ .

#### Solution:

L'algorithme Gibbs sampler consiste à simuler les lois conditionnelles de la loi jointe  $N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$ .

Pour  $t = 1, \dots, T$ , nous simulons donc:

$$Y_t | X_{t-1} \sim N(\rho X_{t-1}, 1 - \rho^2), \text{ et}$$

$$X_t | Y_t \sim N(\rho Y_t, 1 - \rho^2).$$

Dans cet exercice, nous effectuons 10.000 simulations avec un "burn in" égal à 1000.

#### Implémentation:

```

In [74]: # Gibbs sampler
n = 10000
rho = 0.3
burn_in = 1000
# initialisation
x = c()
y = c()
x[1] = 1
y[1] = 1
for (t in 1:n) {
  y[t + 1] = rnorm(1, mean=rho * x[t], sd = sqrt(1 - rho^2))
  x[t + 1] = rnorm(1, mean=rho * y[t+1], sd = sqrt(1 - rho^2))
}

```

Dans un premier nous nous assurons que les estimations sont correctes.

```

In [31]: cor(x, y)

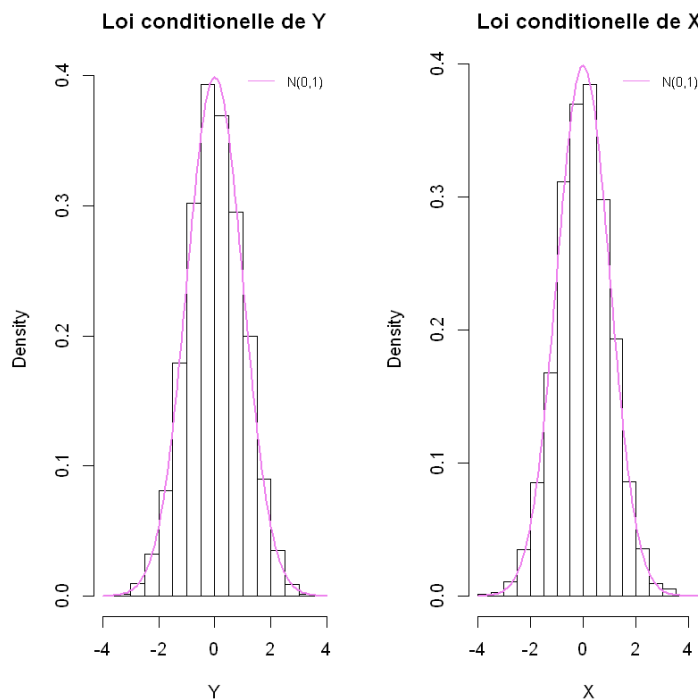
```

0.303223476970593

```

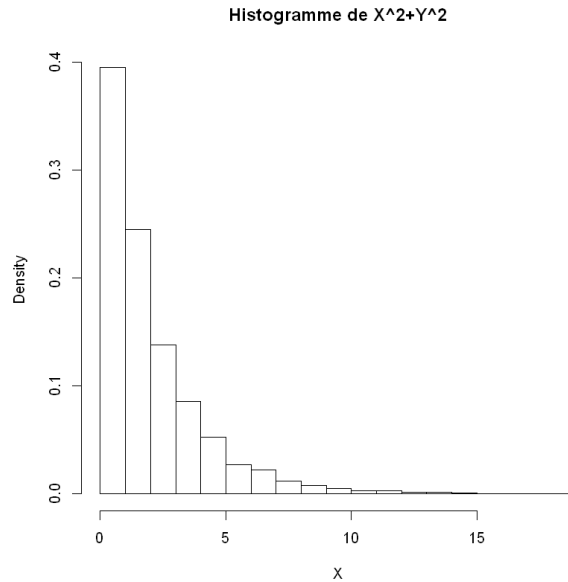
In [78]: par(mfrow = c(1, 2))
hist(y[burn_in:n], freq= F, main = "Loi conditionnelle de Y", xlab= "Y")
curve(dnorm(x, mean=0, sd = 1), col = "violet", add = T, lwd = 2)
legend("topright", legend = c("N(0,1)"), lty=1, col=c("violet"), bty = "n", cex=.75)
hist(x[burn_in:n], freq= F, main = "Loi conditionnelle de X", xlab= "X")
curve(dnorm(x, mean=0, sd = 1), col = "violet", add = T, lwd = 2)
legend("topright", legend = c("N(0,1)"), lty=1, col=c("violet"), bty = "n", cex=.75)

```



Nous nous intéressons désormais à la densité  $X^2 + Y^2$ .

```
In [79]: z = x^2 + y^2
hist(z[burn_in:n], freq= F, main = "Histogramme de X^2+Y^2", xlab= "X")
```



```
In [34]: mean(z>2)
```

0.357664233576642

La probabilité que  $X^2 + Y^2$  soit supérieure à 2 est environ égale à 0.36.

## 4 Exercice 9.18

For the Gibbs sampler

$$X|y \sim N(\rho y, 1 - \rho^2),$$

$$Y|x \sim N(\rho x, 1 - \rho^2),$$

of Example 9.1: a) Show that for the  $X$  chain, the transition kernel is

$$K(x^*, x) = \frac{1}{2\pi(1 - \rho^2)} \int e^{-\frac{1}{2(1-\rho^2)}(x-\rho y)^2} e^{-\frac{1}{2(1-\rho^2)}(y-\rho x^*)^2} dy.$$

**Solution:**

Le noyau de transition de la chaîne de Markov  $X$  est défini par :

$$K(x^*, x) = \int f_{Y|X}(y|x^*) f_{X|Y}(x|y) dy.$$

Or, l'énoncé nous indique que  $X|y \sim N(\rho y, 1 - \rho^2)$ , ainsi on a  $f_{X|Y}(x|y) = \frac{e^{-\frac{(x-\rho y)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}}$ .

De même, on déduit de  $Y|x \sim N(\rho x, 1 - \rho^2)$  que  $f_{Y|X}(y|x) = \frac{e^{-\frac{(y-\rho x)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}}$ .

On obtient donc :

$$K(x^*, x) = \int \frac{e^{-\frac{(x-\rho y)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}} \frac{e^{-\frac{(y-\rho x^*)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}} dy$$

$$\leftrightarrow K(x^*, x) = \frac{1}{2\pi(1-\rho^2)} \int e^{-\frac{1}{2(1-\rho^2)}(x-\rho y)^2} e^{-\frac{1}{2(1-\rho^2)}(y-\rho x^*)^2} dy.$$

**b) Show that  $X \sim N(0, 1)$  is the invariant distribution of the X chain.**

**Solution:**

La distribution stationnaire de la chaîne de Markov X est la densité partielle  $f_X$ .

$$f_X(x) = \frac{1}{\int \frac{f_{Y|X}(y|x)}{f_{X|Y}(x|y)} dy}$$

En développant le dénominateur:

$$\int \frac{f_{Y|X}(y|x)}{f_{X|Y}(x|y)} dy = \int \frac{\frac{1}{\sqrt{2\pi(1-\rho^2)}} e^{-\frac{(y-\rho x)^2}{2(1-\rho^2)}}}{\frac{1}{\sqrt{2\pi(1-\rho^2)}} e^{-\frac{(x-\rho y)^2}{2(1-\rho^2)}}} dy = \int e^{\frac{(x-\rho y)^2 - (y-\rho x)^2}{2(1-\rho^2)}} dy = e^{\frac{x^2}{2}} \int \frac{e^{-y^2}}{2} dy = \sqrt{2\pi} e^{\frac{x^2}{2}}.$$

On obtient donc  $f_X(x) = \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}$  ce qui correspond bien à la densité d'une  $N(0,1)$

**c) Show that  $X|x^* \sim N(\rho^2 x^*, 1 - \rho^4)$ . (Hint: Complete the square in the exponent of part a).)**

**Solution:**

Nous avons:

$$K(x^*, x) = \int \frac{e^{-\frac{(x-\rho y)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}} \frac{e^{-\frac{(y-\rho x^*)^2}{2(1-\rho^2)}}}{\sqrt{2\pi(1-\rho^2)}} dy$$

Cette expression peut être simplifiée. Nous nous concentrons sur le numérateur de l'exponentielle:

$$\begin{aligned} (x - \rho y)^2 + (y - \rho x^*)^2 &= x^2 - x\rho xy + \rho^2 y^2 + y^2 - 2\rho x^* y + \rho^2 x^{*2} \\ &= (1 + \rho^2)y^2 - 2y(\rho x + \rho x^*) + x^2 + \rho^2 x^{*2} \\ &= (1 + \rho^2) \left( y^2 - 2y \frac{\rho x + \rho x^*}{1 + \rho^2} + \left( \frac{\rho x + \rho x^*}{1 + \rho^2} \right)^2 \right) - \frac{(\rho x + \rho x^*)^2 + (x^2 + \rho^2 x^{*2})(1 + \rho^2)}{1 + \rho^2} \end{aligned}$$



En développant encore et en factorisant, nous obtenons:  $= (1 + \rho^2) \left( y - \frac{\rho x + \rho x^*}{1 + \rho^2} \right)^2 + \frac{(x - \rho^2 x^*)^2}{1 + \rho^2}$

Ainsi,

$$\begin{aligned} K(x^*, x) &= \frac{1}{2\pi(1 - \rho^2)} \int e^{-\frac{1}{2(2 - \rho^2)}(1 + \rho^2) \left( y - \frac{\rho x + \rho x^*}{1 + \rho^2} \right)^2} e^{-\frac{1}{2(1 - \rho^2)} \frac{(x - \rho^2 x^*)^2}{1 + \rho^2}} dy \\ &= \frac{1}{\sqrt{2\pi(1 - \rho^4)}} e^{-\frac{1}{2} \frac{(x - \rho^2 x^*)^2}{1 - \rho^4}} \int \sqrt{\frac{1 + \rho^2}{2\pi(1 - \rho^2)}} e^{-\frac{1}{2(1 - \rho^2)}(1 + \rho^2) \left( y - \frac{\rho x + \rho x^*}{1 + \rho^2} \right)^2} \\ &= \frac{1}{\sqrt{2\pi(1 - \rho^4)}} e^{-\frac{1}{2} \frac{(x - \rho^2 x^*)^2}{1 - \rho^4}}, \text{ ce qui est bien la densité d'une } N(\rho^2 x^*, 1 - \rho^4) \end{aligned}$$

Cela montre bien que  $X|x^* \sim N(\rho^2 x^*, 1 - \rho^4)$

**d) Show that we can write  $X_k = \rho^2 X_{k-1} + U_k$ ,  $k=1,2,\dots$ , where the  $U_k$  are iid  $N(0, 1 - \rho^4)$  and that  $cov(X_0, X_k) = \rho^{2k}$ , for all  $k$ . Deduce that the covariances go to zero.**

**Solution:**

D'après la question précédente,  $\forall k \in \mathbb{N}^*$ ,  $X_k | X_{k-1} \sim N(\rho^2 X_{k-1}, 1 - \rho^4)$  donc  $X_k - \rho^2 X_{k-1} \sim N(0, 1 - \rho^4)$ .

En posant  $\forall k \in \mathbb{N}^* : U_k = X_k - \rho^2 X_{k-1}$ , on déduit:

$$\begin{aligned} X_k &= U_k + \rho^2 X_{k-1} \\ &= U_k + \rho^2 (\rho^2 X_{k-2} + U_{k-1}) \\ &= \dots \\ &= \sum_{i=0}^{k-1} \rho^{2i} U_{k-i} + \rho^{2k} X_0 \end{aligned}$$

Ainsi la covariance s'écrit :

$$\begin{aligned} Cov(X_0, X_k) &= cov(X_0, \sum_{i=0}^{k-1} \rho^{2i} U_{k-i} + \rho^{2k} X_0) \\ &= \sum_{i=0}^{k-1} \rho^{2i} cov(X_0, U_{k-i}) + \rho^{2k} var(X_0) \\ &= Cov(X_0, X_k) = \rho^{2k} \end{aligned}$$

Car

- $var(X_0) = 1$  puisque  $X_0 \sim N(0, 1)$
- $Cov(X_0, U_{k-i}) = 0$  puisque  $X_0$  et  $U_i$  sont indépendants.

Par ailleurs, puisque  $1 - \rho^2 = var(X|y) > 0$ ,  $\rho \in [-1; 1]$ . Ainsi,  $\lim_{k \rightarrow +\infty} \rho^{2k} = 0$  et par conséquent la séquence  $Cov(X_0, X_k)$  converge vers 0.

## 5 Exercice 10.25

a) For the situation of Example 10.31, show that the full posterior distribution of  $(\beta, \sigma^2, \tau^2)$  is

$$\pi(\beta, \sigma^2, \tau^2 | y) \propto \sigma^{-2-I} \tau^{-2-IJ} \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2\right) \times \exp\left(-\frac{J \sum_i (\bar{y}_i - \beta)^2}{2(\tau^2 + J\sigma^2)}\right) (J\tau^{-2} + \sigma^{-2})^{-\frac{I}{2}}.$$

**Solution:**

L'exemple 10.31 est un modèle à effets aléatoires:

$$Y_{i,j} = \beta + U_i + \epsilon_{ij}$$

avec  $U_i \sim N(0, \sigma^2)$  et  $\epsilon_{ij} \sim N(0, \tau^2)$ ,  $i = 1, \dots, I$  et  $j = 1, \dots, J$ .

Nous avons donc:

$$\begin{aligned} \pi(\beta, u, \sigma^2, \tau^2 | y) &\propto \pi(y | \beta, u, \sigma^2, \tau^2) \pi(\beta, u, \sigma^2, \tau^2) \\ &\propto \pi(y | \beta, u, \sigma^2, \tau^2) \pi(\beta, \sigma^2, \tau^2) \pi(u) \text{ par indépendance} \\ &\propto \prod_{i,j=1}^{IJ} \left[ \tau^{-1} \exp\left(-\frac{(y_{i,j} - (\beta + U_i))^2}{2\tau^2}\right) \right] \sigma^{-2} \tau^{-2} \prod_{i=1}^I \left( \sigma^{-1} \exp\left(-\frac{U_i^2}{2\sigma^2}\right) \right) \\ &\propto \sigma^{-2-I} \tau^{-IJ-2} \exp\left(\sum_{i,j=1}^{IJ} \frac{-(y_{i,j} - \beta - U_i)^2}{2\tau^2} + \sum_{i=1}^I \frac{-U_i^2}{2\sigma^2}\right) = A_{i,j} \end{aligned}$$

Simplifions l'expression ci-dessus:

$$Z_i = \sum_{j=1}^J \frac{-(y_{i,j} - \beta - U_i)^2}{2\tau^2} + \frac{-U_i^2}{2\sigma^2} = \sum_{j=1}^J -\frac{(y_{i,j} - \beta)^2 - 2U_i(y_{i,j} - \beta) + U_i^2}{2\tau^2} + \frac{-U_i^2}{2\sigma^2}$$

En développant cette expression, nous trouvons :

$$Z_i = \sum_{j=1}^J -\frac{(y_{i,j} - \beta)^2}{2\tau^2} - \left(\frac{J}{2\tau^2} + \frac{1}{2\sigma^2}\right) \left[ \left(U_i - \frac{\sum_j (y_{i,j} - \beta)}{J + \tau^2 \sigma^{-2}}\right)^2 - \left(\frac{\sum_j (y_{i,j} - \beta)}{J + \tau^2 \sigma^{-2}}\right)^2 \right]$$

$$\text{Donc } U_i \sim N\left(\frac{\sum_j (y_{i,j} - \beta)}{J + \tau^2 \sigma^{-2}}, \frac{1}{J\tau^{-2} + \sigma^{-2}}\right).$$

En intégrant  $A_{i,j}$  par rapport à  $u$ , nous obtenons :

$$\pi(\beta, \sigma^2, \tau^2 | y) \propto \sigma^{-2-I} \tau^{-2-IJ} \prod_i \sqrt{\frac{1}{J\tau^{-2} + \sigma^{-2}}} \exp\left(\sum_{i,j} -\frac{(y_{i,j} - \beta)^2}{2\tau^2} + \left(\frac{J}{2\tau^2} + \frac{1}{2\sigma^2}\right) \left(\frac{\sum_j (y_{i,j} - \beta)}{J + \tau^2 \sigma^{-2}}\right)^2\right)$$

$$\text{En développant cette expression nous trouvons bien : } \pi(\beta, \sigma^2, \tau^2 | y) \propto \sigma^{-2-I} \tau^{-2-IJ} \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2\right) \times \exp\left(-\frac{J \sum_i (\bar{y}_i - \beta)^2}{2(\tau^2 + J\sigma^2)}\right) (J\tau^{-2} + \sigma^{-2})^{-\frac{I}{2}}.$$

b) Integrate out  $\beta$  and show that the marginal posterior density of  $(\sigma^2, \tau^2)$  is

$$\pi(\sigma^2, \tau^2 | y) \propto \frac{\sigma^{-2-I} \tau^{-2-IJ}}{(J\tau^{-2} + \sigma^{-2})^{\frac{I}{2}}} (\tau^2 + J\sigma^2)^{\frac{I}{2}} \times \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2 - \frac{J}{2(\tau^2 + J\sigma^2)} \sum_i (\bar{y}_i - \bar{y})^2\right).$$

**Solution:**

On a  $\pi(\beta, \sigma^2, \tau^2 | y) \propto \sigma^{-2-I} \tau^{-2-IJ} \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2\right) \times \exp\left(-\frac{I \sum_i (\bar{y}_i - \beta)^2}{2(\tau^2 + J\sigma^2)}\right) (J\tau^{-2} + \sigma^{-2})^{-\frac{I}{2}}$  (question a).

$$\text{Or } \sum_i (\bar{y}_i - \beta)^2 = \sum_i \bar{y}_i^2 - 2\beta \sum_i \bar{y}_i + I\beta^2 = I(\beta^2 - \frac{2\beta \sum_i \bar{y}_i}{I}) + \sum_i \bar{y}_i^2 = I(\beta - \frac{\sum_i \bar{y}_i}{I})^2 - \frac{(\sum_i \bar{y}_i)^2}{I} + \sum_i \bar{y}_i^2$$

On remarque donc que  $\beta | y \sim N(\frac{\sum_i \bar{y}_i}{I}, \frac{\tau^2 + J\sigma^2}{IJ})$ .

En intégrant sur  $\beta$ , on a :

$$\begin{aligned} \pi(\sigma^2, \tau^2 | y) &\propto \sigma^{-2-I} \tau^{-2-IJ} (J\tau^{-2} + \sigma^{-2})^{-\frac{I}{2}} (J\sigma^2 + \tau^2)^{\frac{1}{2}} \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2\right) \exp\left(-J \frac{(\sum_i \bar{y}_i)^2 I^{-1} + \sum_i \bar{y}_i^2}{2(\tau^2 + J\sigma^2)}\right) \\ &\propto \frac{\sigma^{-2-I} \tau^{-2-IJ}}{(J\tau^{-2} + \sigma^{-2})^{\frac{I}{2}}} (\tau^2 + J\sigma^2)^{\frac{1}{2}} \times \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2 - \frac{J}{2(\tau^2 + J\sigma^2)} \sum_i (\bar{y}_i - \bar{y})^2\right). \end{aligned}$$

$$\text{car } -(\sum_i \bar{y}_i)^2 I^{-1} + \sum_i \bar{y}_i^2 = \sum_i (\bar{y}_i - \bar{y})^2$$

**c) Show that the full posterior is not integrable since, for  $\tau \neq 0$ ,  $\pi(\sigma^2, \tau^2 | y)$  behaves like  $\sigma^{-2}$  in a neighborhood of 0.**

**Solution:**

Lorsque  $\sigma^2$  est proche de 0:

$$\begin{aligned} \pi(\sigma^2, \tau^2 | y) &\approx \sigma^{-2-I} \tau^{-2-IJ} \sigma^I |\tau| \exp\left(-\frac{\sum_{i,j} (\bar{y}_{i,j} - \bar{y})^2}{2\tau^2}\right) \\ &\approx \sigma^{-2} \tau^{-2-IJ} |\tau| \exp\left(-\frac{\sum_{i,j} (\bar{y}_{i,j} - \bar{y})^2}{2\tau^2}\right) \end{aligned}$$

car:

$$- \lim_{\sigma^2 \rightarrow 0} \exp\left(-\frac{1}{2\tau^2} \sum_{i,j} (y_{i,j} - \bar{y}_i)^2 - \frac{J}{2(\tau^2 + J\sigma^2)} \sum_i (\bar{y}_i - \bar{y})^2\right) = \exp\left(-\frac{\sum_{i,j} (\bar{y}_{i,j} - \bar{y})^2}{2\tau^2}\right), \text{ et lorsque } \sigma^2 \text{ est proche de 0, on a:}$$

$$- \frac{\sigma^{-2-I} \tau^{-2-IJ}}{(J\tau^{-2} + \sigma^{-2})^{\frac{I}{2}}} \approx \sigma^{-2-I} \tau^{-2-IJ} |\tau|$$

$$- (J\tau^{-2} + \sigma^{-2})^{\frac{1}{2}} \approx \sigma^I$$

On voit donc que  $\pi(\sigma^2, \tau^2)$  n'est pas intégrable.

## 6 Exercice 6.51

For the two-state transition matrix  $P = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$ , show that

a) the stationary distribution is  $\pi = (\beta/(\alpha + \beta), \alpha/(\alpha + \beta))$

**Solution:**

La distribution stationnaire est telle que  $\pi P = \pi$ , ce qui est équivalent à:

$$\begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix} \begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix} = \begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix}$$
$$\iff \begin{cases} (1-\alpha)\pi_1 + \beta\pi_2 = \pi_1 \\ \alpha\pi_1 + (1-\beta)\pi_2 = \pi_2 \end{cases}$$

Or  $\pi$  est une probabilité, donc  $\pi_1 + \pi_2 = 1$ . La solution du système ci-dessus est donc :

$$\begin{cases} \pi_1 = \frac{\beta}{\alpha+\beta} \\ \pi_2 = \frac{\alpha}{\alpha+\beta} \end{cases}$$

b) the mean first passage matrix is  $M = \begin{pmatrix} (\alpha + \beta)/\beta & 1/\alpha \\ 1/\beta & \alpha + \beta/\alpha \end{pmatrix}$

**Solution:**

Soit  $m_{i,j}$  un élément de la matrice de premier passage moyen  $M$  et  $\lambda_i$  le nombre d'étapes avant d'entrer dans le  $i^{\text{eme}}$  état.

Deux situations se présentent :

- si nous commençons à partir du 1er état, on a  $P(\lambda_1 = 1) = 1 - \alpha$  et  $P(\lambda_1 = k) = \alpha\beta(1 - \beta)^{k-2}$
- si nous commençons à partir du second état, on a  $P(\lambda_1 = k) = \beta(1 - \beta)^{k-1}$ .

Ainsi:

$$\begin{aligned}
m_{1,1} &= \sum_{k=1}^{\infty} kP(\lambda_1 = k) \\
&= 1 - \alpha + \sum_{k=1}^{\infty} k\alpha\beta(1 - \beta)^{k-2} \\
&= 1 - \alpha + \alpha\beta \sum_{k=2}^{\infty} k(1 - \beta)^{k-2} \\
&= 1 - \alpha + \alpha \frac{\beta}{1 - \beta} \frac{\partial}{\partial \beta} \left( \sum_{k=2}^{\infty} -k(1 - \beta)^k \right) \\
&= 1 - \alpha + \alpha \frac{\beta}{1 - \beta} \frac{\partial}{\partial \beta} \left( \sum_{k=0}^{\infty} -k(1 - \beta)^k + 2 - \beta \right) \\
&= 1 - \alpha + \alpha \frac{\beta}{1 - \beta} \frac{\partial}{\partial \beta} \left( -\frac{1}{\beta} + 2 - \beta \right) \\
&= \frac{\alpha + \beta}{\beta}
\end{aligned}$$

De même,

$$\begin{aligned}
m_{2,1} &= \sum_{k=1}^{\infty} k\beta(1 - \beta)^{k-1} \\
&= \beta \frac{\partial}{\partial \beta} \left( -\sum_{k=0}^{\infty} (1 - \beta)^k \right) \\
&= \beta \frac{\partial}{\partial \beta} \left( -\frac{1}{1 - (1 + (1 - \beta))} \right) \\
&= \beta \frac{\partial}{\partial \beta} \left( \frac{1}{\beta} \right) \\
&= \frac{1}{\beta}
\end{aligned}$$

Par symétrie, on déduit:  $m_{2,2} = \frac{\alpha + \beta}{\beta}$  et  $m_{1,2} = \frac{1}{\beta}$

Pour conclure la matrice de premier passage moyen est  $M = \begin{pmatrix} (\alpha + \beta)/\beta & 1/\alpha \\ 1/\beta & (\alpha + \beta)/\alpha \end{pmatrix}$