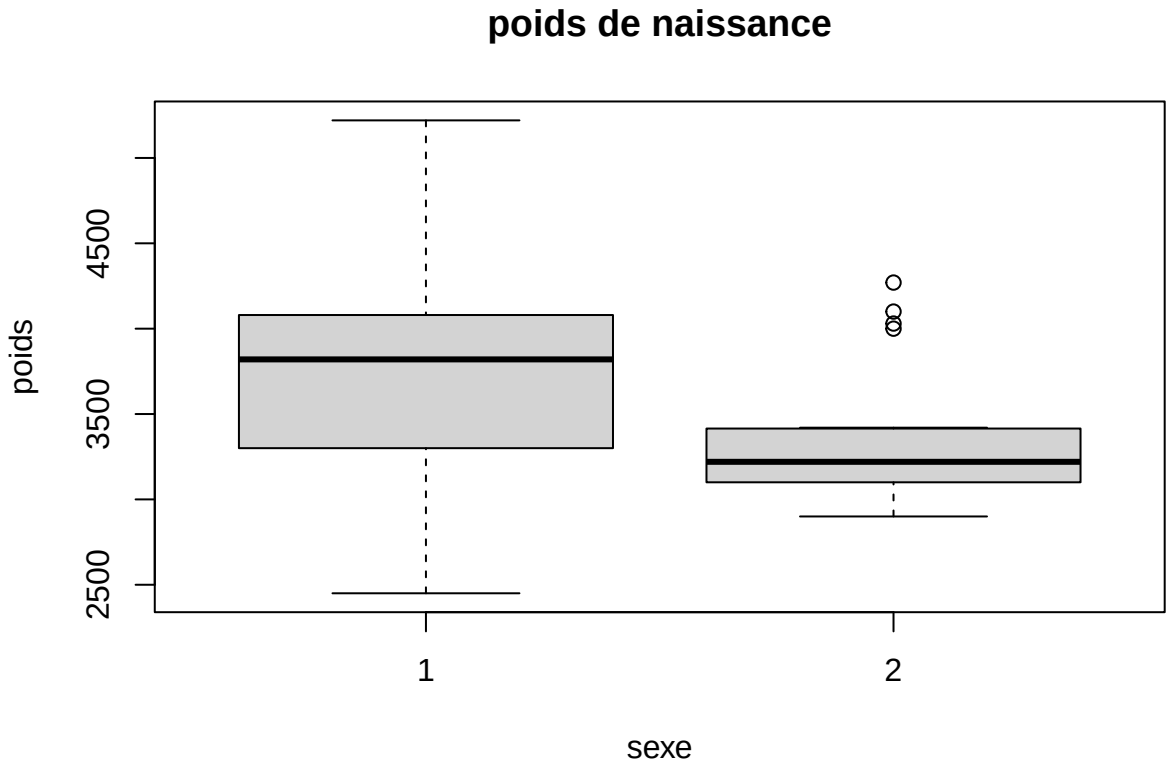
Statistique bayésienne avec R

Exercice sur les poids de naissance

Julien JACQUES

```
data=read.table('Rcode/poidsnaissance.txt',header = T,sep=',',row.names = 1)
data$OBS=NULL
sexe=data$SEXE+1
poids=data$POIDNAIS
boxplot(poids~sexe,main="poids de naissance")
```



Modélisation hierarchique du poids en fonction du sexe

On commence par définir les données

```
dat <- list(poids = poids, sexe = sexe, N = length(poids))
```

Puis 3 initialisations différentes

```
inits <- list( list(moyennes = c(2600, 4000), sigma = 500),list(moyennes = c(4500, 2700), sigma = 700),l
```

On définit le modèle

```
library(rjags)
```

```
## Loading required package: coda
```

```
## Linked to JAGS 4.3.0
```

```
## Loaded modules: basemod,bugs
```

```
m1 <- jags.model('Rcode/modelepoidsnaissance.txt', data = dat, inits = inits, n.chains = 3, quiet=TRUE)
```

Puis on lance les itérations MCMC

```
update(m1, 3000,progress.bar="none")
mcmc1 <- coda.samples(m1, variable.names = c("moyennes", "sigma"), n.iter = 2000,progress.bar="none")
```

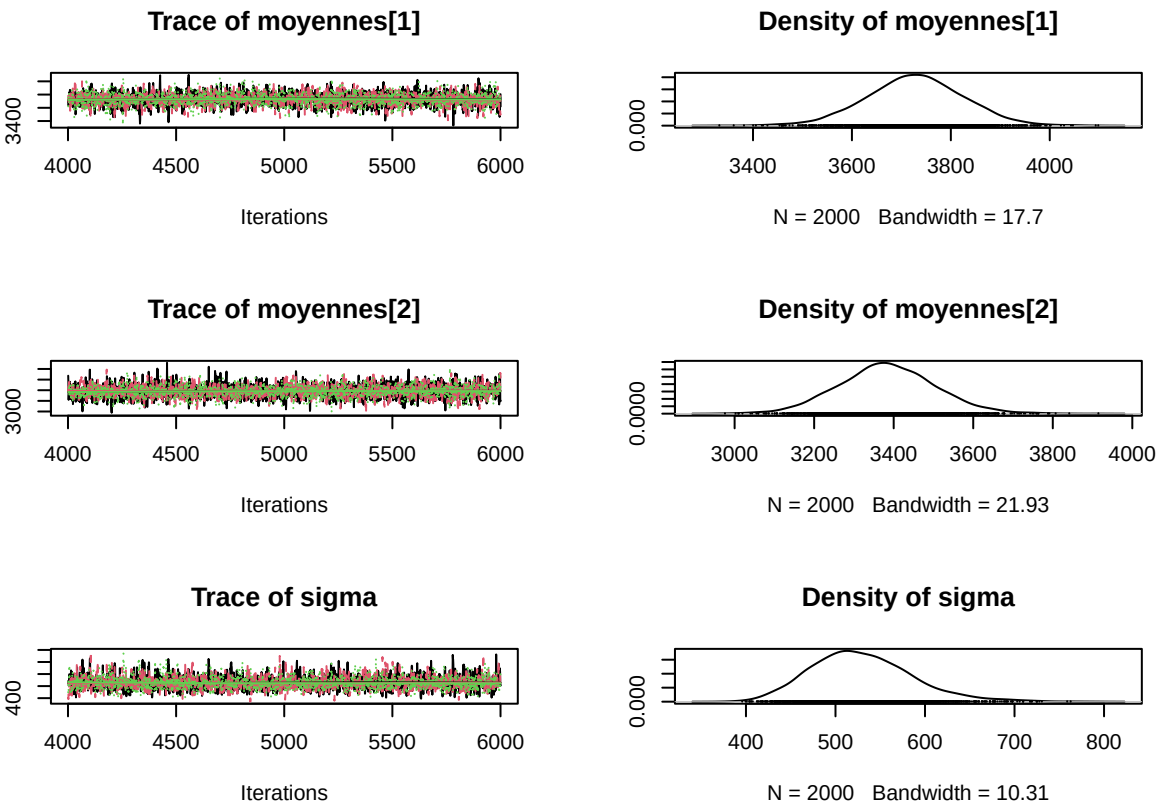
Comme résultat on peut regarder par exemple la moyenne du poids moyen des garçons :

```
mean(mcmc1[[1]][, "moyennes[1]"])
```

```
## [1] 3730.914
```

On peut représenter les chaînes MCMC

```
plot(mcmc1)
```



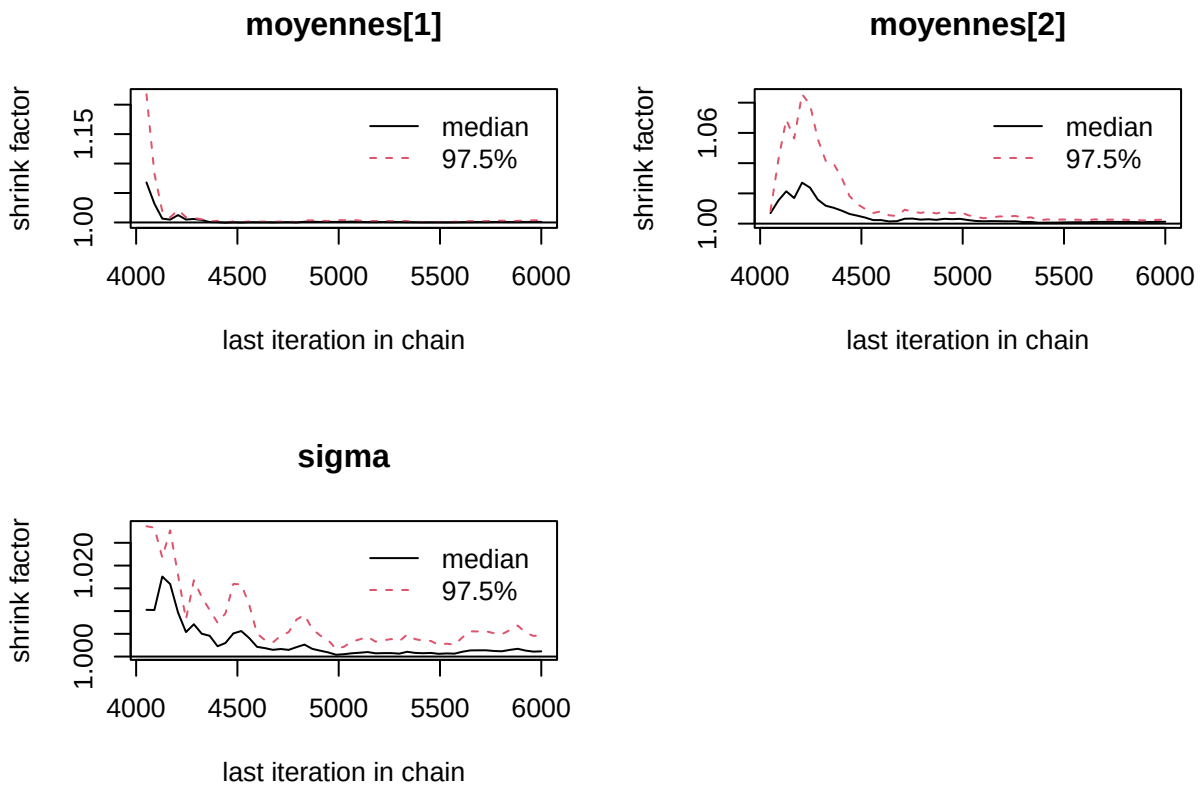
Les diagnostiques de convergence permettent de vérifier que la période de chauffe était suffisament longue

```
gelman.diag(mcmc1)
```

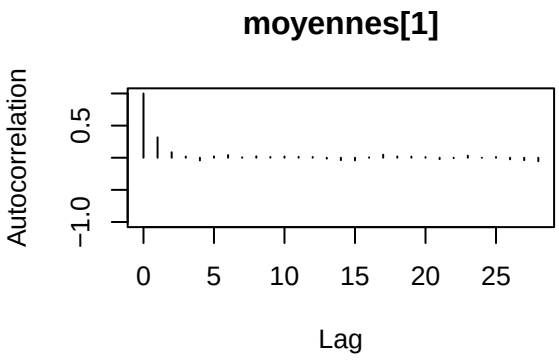
```
## Potential scale reduction factors:
##
##              Point est. Upper C.I.
## moyennes[1]           1          1
## moyennes[2]           1          1
```

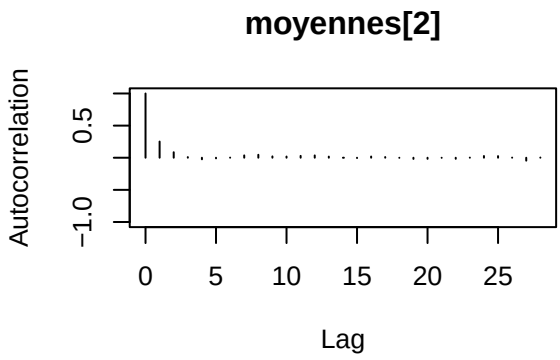
```
## sigma                1          1
##
## Multivariate psrf
##
## 1
```

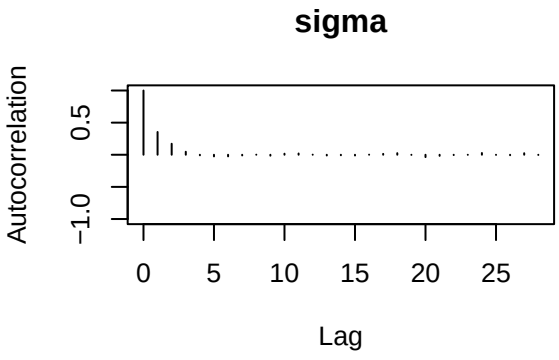
```
gelman.plot(mcmc1)
```




```
autocorr.plot(mcmc1[[1]])
```







On peut finalement examiner les résultats (loi a posteriori des paramètres) :

```
summary(mcmc1)
```

```
##
## Iterations = 4001:6000
## Thinning interval = 1
## Number of chains = 3
## Sample size per chain = 2000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##               Mean     SD Naive SE Time-series SE
## moyennes[1] 3727.8  98.18   1.2676          1.621
## moyennes[2] 3383.1 121.22   1.5649          1.975
## sigma        529.5  56.63   0.7311          1.022
##
## 2. Quantiles for each variable:
##
##               2.5%    25%    50%    75%  97.5%
## moyennes[1] 3534.4 3664.5 3727.7 3792.0 3918.2
## moyennes[2] 3146.5 3303.3 3382.2 3461.2 3629.3
## sigma        432.3  489.9  524.7  564.1  656.2
```

On peut aussi calculer le critère DIC :

```
dic.samples(m1,n.iter=1000)
```

```
## Mean deviance:  736.9
## penalty 3.124
```

```
## Penalized deviance: 740
```

Comment répondre à la question de l'influence du sexe sur le poids de naissance ?

Solution 1 :

On compare avec un modèle sans la variable sexe

On commence par définir les données

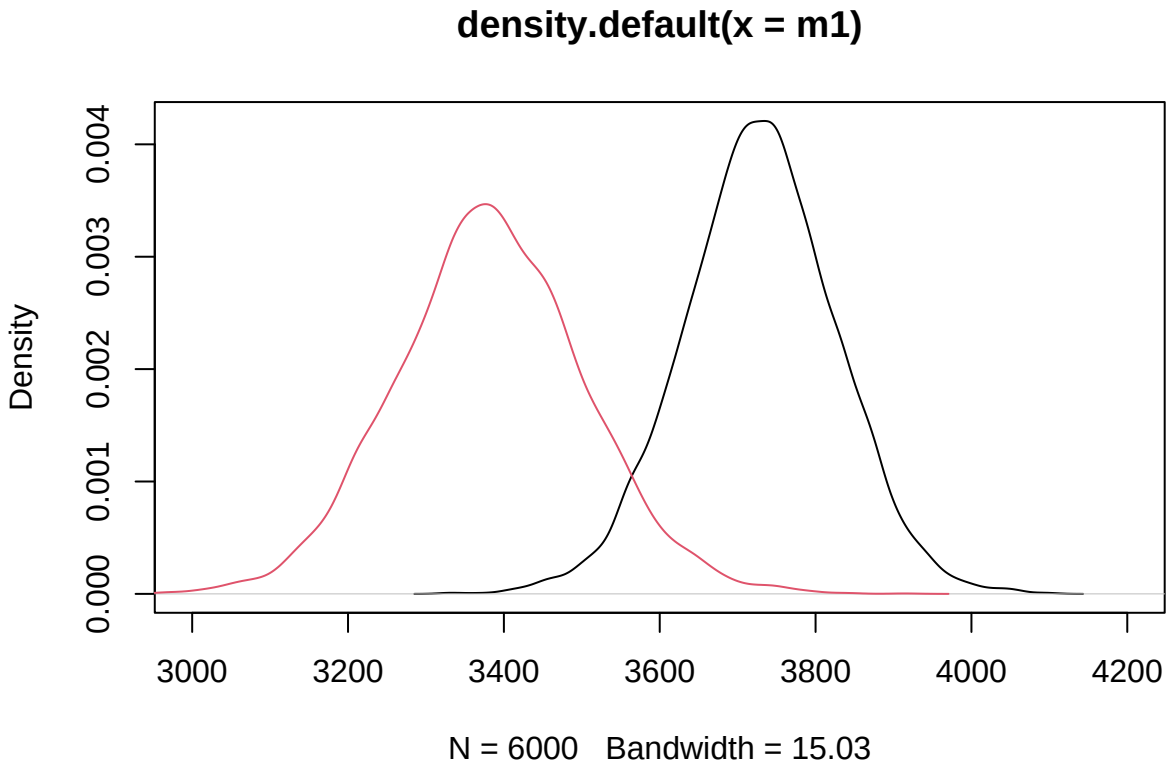
```
dat <- list(poids = poids, N = length(poids))
inits <- list( list(mu = c(2600), sigma = 500),list(mu = c(4500), sigma = 700),list(mu = c(4000), sigma
m0 <- jags.model('Rcode/modelepoidsnaissance0.txt', data = dat, inits = inits, n.chains = 3, quiet=TRUE)
update(m0, 3000,progress.bar="none")
dic.samples(m0,n.iter=1000)
```

```
## Mean deviance:  741.2
## penalty 2.225
## Penalized deviance: 743.4
```

Solution 2 :

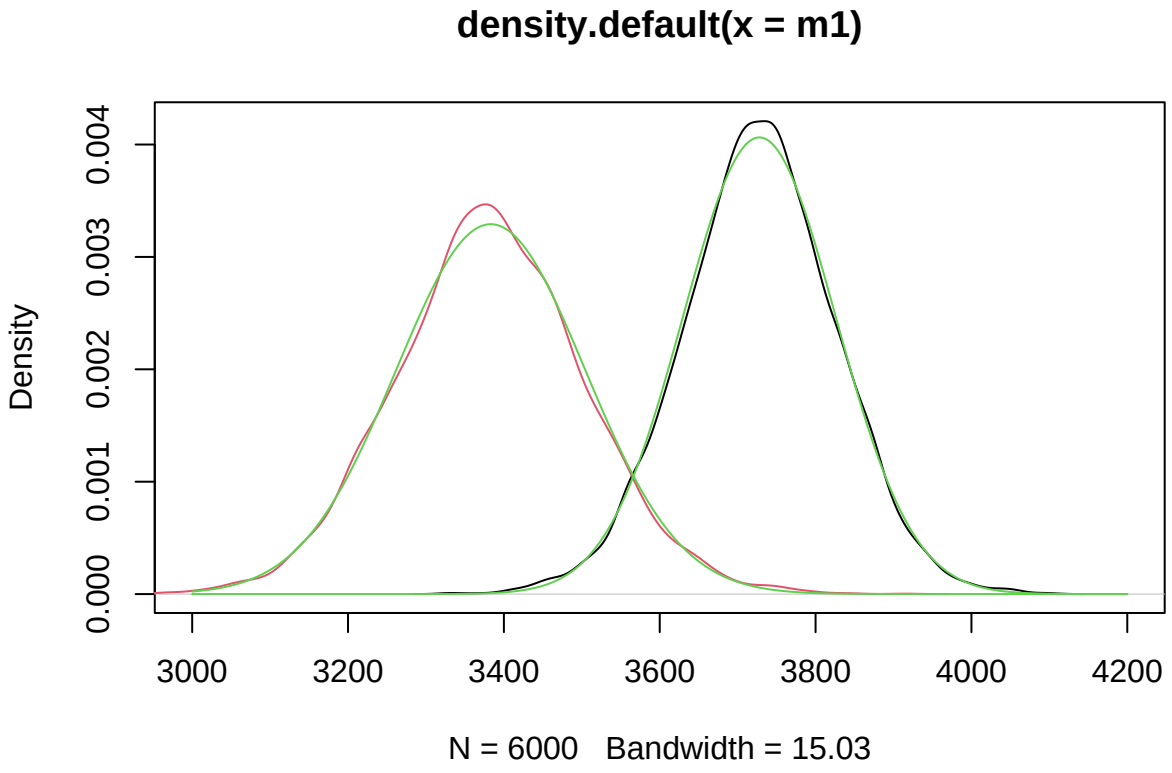
On peut utuliser les lois a posteriori des poids moyens des filles et des garçons.

```
m1=c(mcmc1[[1]][,1],mcmc1[[2]][,1],mcmc1[[3]][,1])
m2=c(mcmc1[[1]][,2],mcmc1[[2]][,2],mcmc1[[3]][,2])
plot(density(m1),xlim=c(3000,4200))
lines(density(m2),col=2)
```



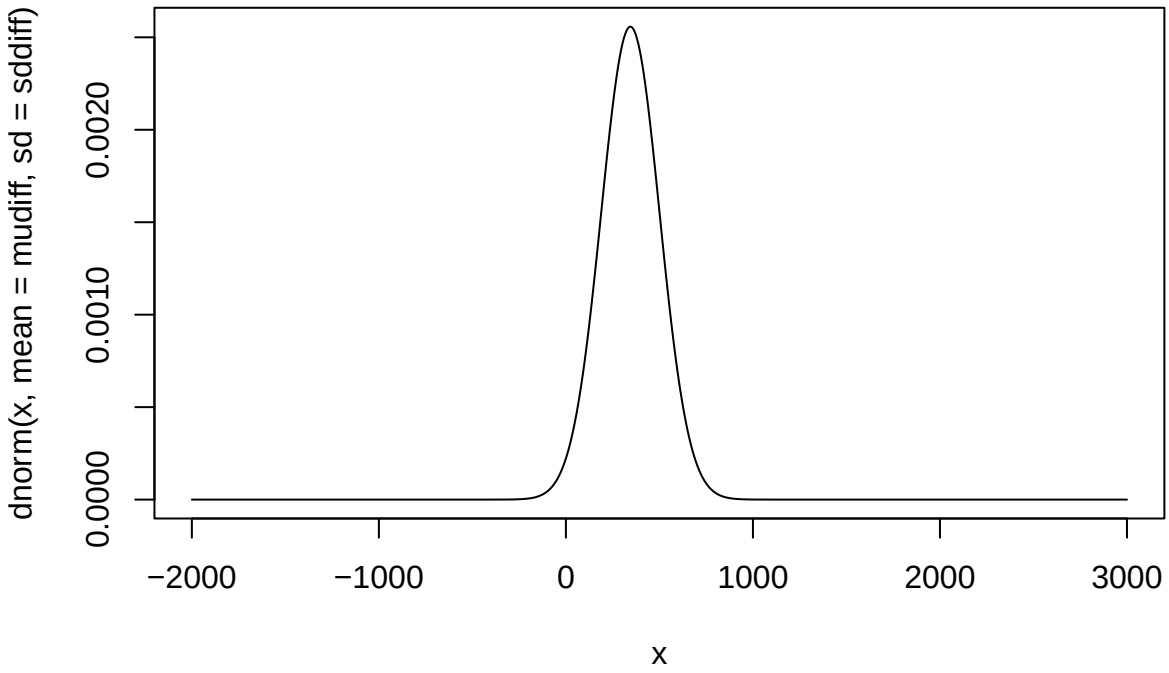
Ces densités peuvent être vraisemblablement approchées par des lois gaussiennes

```r
plot(density(m1),xlim=c(3000,4200))
lines(density(m2),col=2)
x=seq(3000,4200,1)
lines(x,dnorm(x,mean=mean(m1),sd=sd(m1)),col=3)
lines(x,dnorm(x,mean=mean(m2),sd=sd(m2)),col=3)
```



Sous l'hypothèse gaussienne, la loi de la différence entre les poids moyens des garçons et des filles est une loi normale dont l'espérance est la différences des espérances et la variance et la somme des variances :

```r
mudiff=mean(m1)-mean(m2)
sddiff=sqrt(sd(m1)^2+sd(m2)^2)
x=seq(-2000,3000,1)
plot(x,dnorm(x,mean=mudiff,sd =sddiff),type='l')
```



Il suffit alors de calculer la probabilité que cette différence soit positive :

```
print(1-pnorm(0,mean=mudiff,sd =sddiff))
```

```
## [1] 0.9864429
```

Cette probabilité correspond à la probabilité, que le poids moyen des garçons soit plus grand que le poids moyen des filles.

Modélisation hierarchique du poids en fonction du sexe et de l'âge gestiationnel

On commence par définir les données

```
dat <- list(poids = poids, sexe = sexe, N = length(poids), nbsemaines = data$AGEGEST)
```

Puis 3 initialisations différentes

```
inits <- list( list(moyennes = c(2600, 4000), sigma = 500, b=0),list(moyennes = c(4500, 2700), sigma =
```

On définit le modèle

```
library(rjags)
m2 <- jags.model('Rcode/modelepoidsnaissance3.txt', data = dat, inits = inits, n.chains = 3, quiet=TRUE
```

Puis on lance les itérations MCMC

```
update(m2, 3000,progress.bar="none")
mcmc2 <- coda.samples(m2, variable.names = c("moyennes", "sigma","b"), n.iter = 2000,progress.bar="none
```

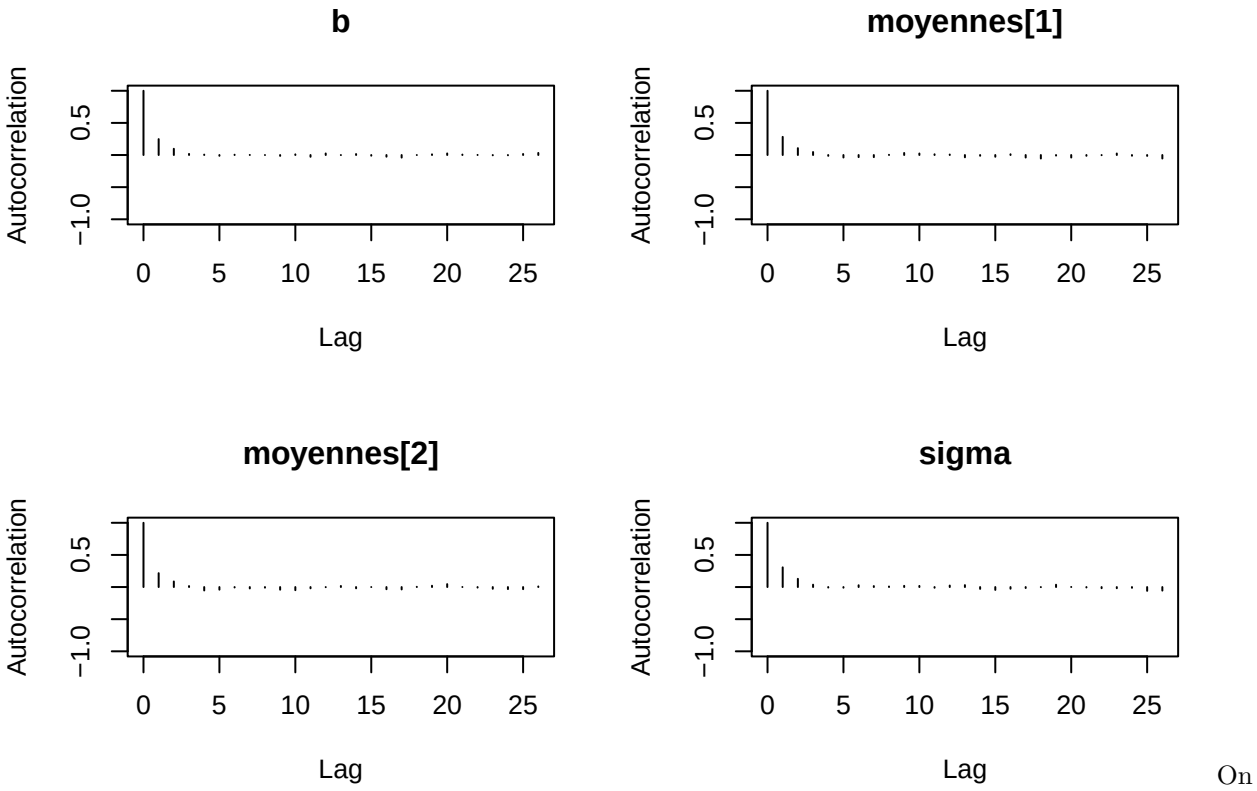
Les diagnostiques de convergence permettent de vérifier que la période de chauffe était suffisament longue

```
gelman.diag(mcmc2)
```

```
## Potential scale reduction factors:
##
```

```
##             Point est. Upper C.I.
## b                    1          1
## moyennes[1]          1          1
## moyennes[2]          1          1
## sigma                1          1
##
## Multivariate psrf
##
## 1
```

```
autocorr.plot(mcmc2[[1]])
```



On peut finalement examiner les résultats (loi a posteriori des paramètres) :

```
summary(mcmc2)
```

```
##
## Iterations = 4001:6000
## Thinning interval = 1
## Number of chains = 3
## Sample size per chain = 2000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##               Mean     SD Naive SE Time-series SE
## b            152.8  60.73   0.7841         1.0343
## moyennes[1] 3743.7  93.12   1.2021         1.5269
## moyennes[2] 3352.1 110.82   1.4307         1.8427
## sigma        500.9  53.72   0.6936         0.9778
##
```

```
## 2. Quantiles for each variable:
##
##                  2.5%    25%    50%    75%  97.5%
## b              36.88  112.6  153.4  193.6  269.9
## moyennes[1] 3562.53 3680.9 3744.5 3805.8 3928.0
## moyennes[2] 3134.28 3278.3 3352.9 3424.7 3569.6
## sigma        409.29  463.0  496.7  533.3  618.9
```

On peut aussi calculer le critère DIC, qui est meilleur, ce qui signifie que l'apport de la nouvelle variable est significatif :

```
dic.samples(m2,n.iter=1000)
```

```
## Mean deviance:  731.6
## penalty 4.244
## Penalized deviance: 735.9
```