

Estimation de la population de phoques sur l'île Marion

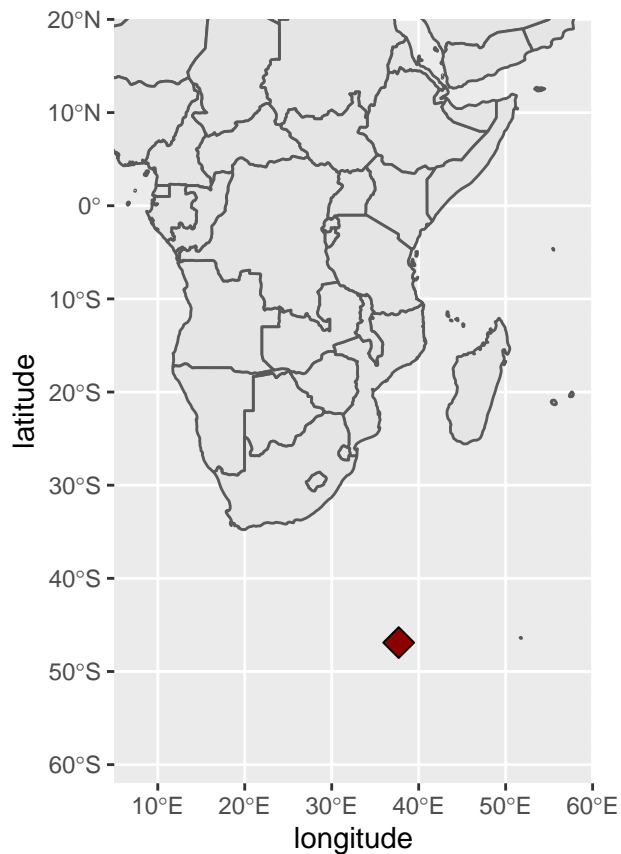
MP Etienne

11/14/2019

Contexte

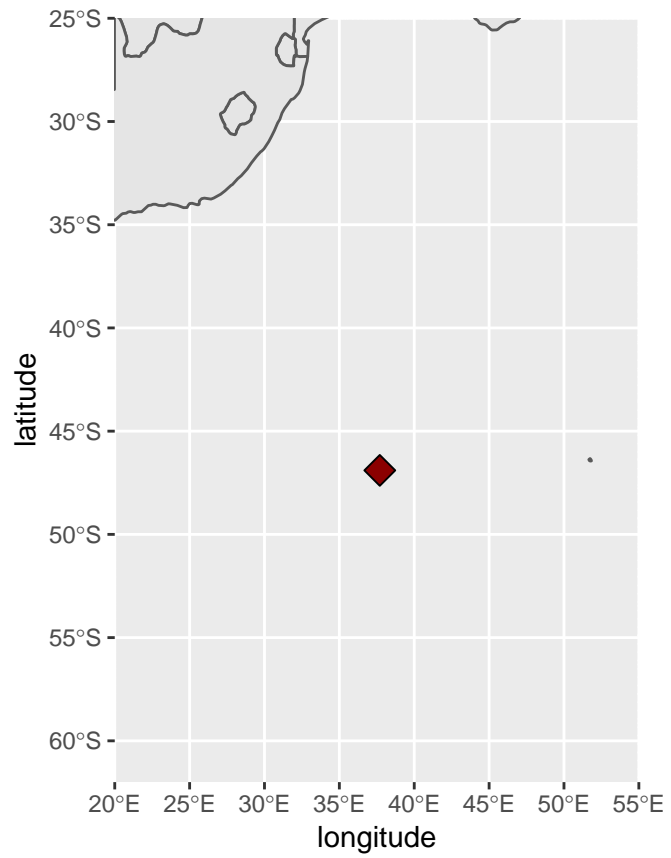
Dans une étude publiée en 2016, (un extrait est disponible ici, les auteurs s'intéressent à l'évolution de la population de phoques sur l'île Marion entre 1995 et 2015.

```
marion_coord <- data.frame(longitude = c(37.7), latitude = c(-46.9))
ggplot(data = world) +
  geom_sf() +
  geom_point(data = marion_coord, aes(x = longitude, y = latitude), size = 4,
            shape = 23, fill = "darkred") +
  coord_sf(xlim = c(5, 60), ylim = c(-62, 20), expand = FALSE)
```



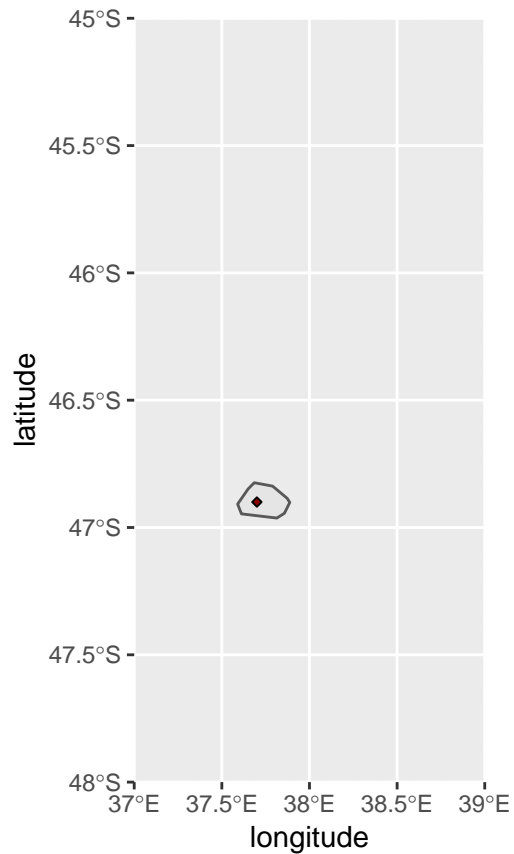
En zoomant, un peu

```
ggplot(data = world) +
  geom_sf() +
  geom_point(data = marion_coord, aes(x = longitude, y = latitude), size = 4,
            shape = 23, fill = "darkred") +
  coord_sf(xlim = c(20, 55), ylim = c(-62, -25), expand = FALSE)
```



et en zoomant beaucoup

```
ggplot(data = world) +  
  geom_sf() +  
  geom_point(data = marion_coord, aes(x = longitude, y = latitude), size = 1,  
            shape = 23, fill = "darkred") +  
  coord_sf(xlim = c(37, 39), ylim = c(-48, -45), expand = FALSE)
```



Les données de capture marquage recapture de Cape Davis

Les données de marquage sont disponibles ici

```
CMR_Davis <- read.table('datasets/TD_pup/CMR_Davis.csv', sep = ',', header = TRUE)
CMR_Davis %>% group_by(Year) %>% summarise(n_transect = n_distinct(Transect), n_rep = n_distinct(Repeat))
```

```
## # A tibble: 10 x 3
##   Year n_transect n_rep
##   <int>     <int> <int>
## 1 1995         3     2
## 2 2007         8     1
## 3 2008         8     1
## 4 2009         8     3
## 5 2010         9     1
## 6 2011         6     3
## 7 2012         6     3
## 8 2013         7     3
## 9 2014         7     3
## 10 2015        10     9
```

Modèle pour une seule observation

On ne considère dans un premier temps qu'une seule opération de recapture pour une année donnée

```
CMR_Davis %>% filter(Year == 2007, Transect ==1) -> CMR_2007_T1
CMR_2007_T1
```

```
##   Year Transect Repeat   M NM  n R   Site
## 1 2007         1     1 109 11 19 8 MM046M
```

On a marqué m=109 individus, 8 ont été capturés marqués et 11 individus non marqués ont été capturés.

But : Estimer la taille de la population.

Rappel : Mise en oeuvre du modèle CMR depuis R avec JAGS

```
library('rjags')
library('ggmcmc')

data.list <- dget(file = 'data_CMR.txt')

init.list <- dget(file = 'init_CMR.txt')

mjags <- jags.model(file = 'modelCMR.txt',
  data = data.list,
  inits = init.list, n.chains = 3)
```

```
## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 2
##   Unobserved stochastic nodes: 2
##   Total graph size: 10
##
## Initializing model

post_sample <- coda.samples(mjags, variable.names = c('p', 'N'),
  n.iter = 5000)
```

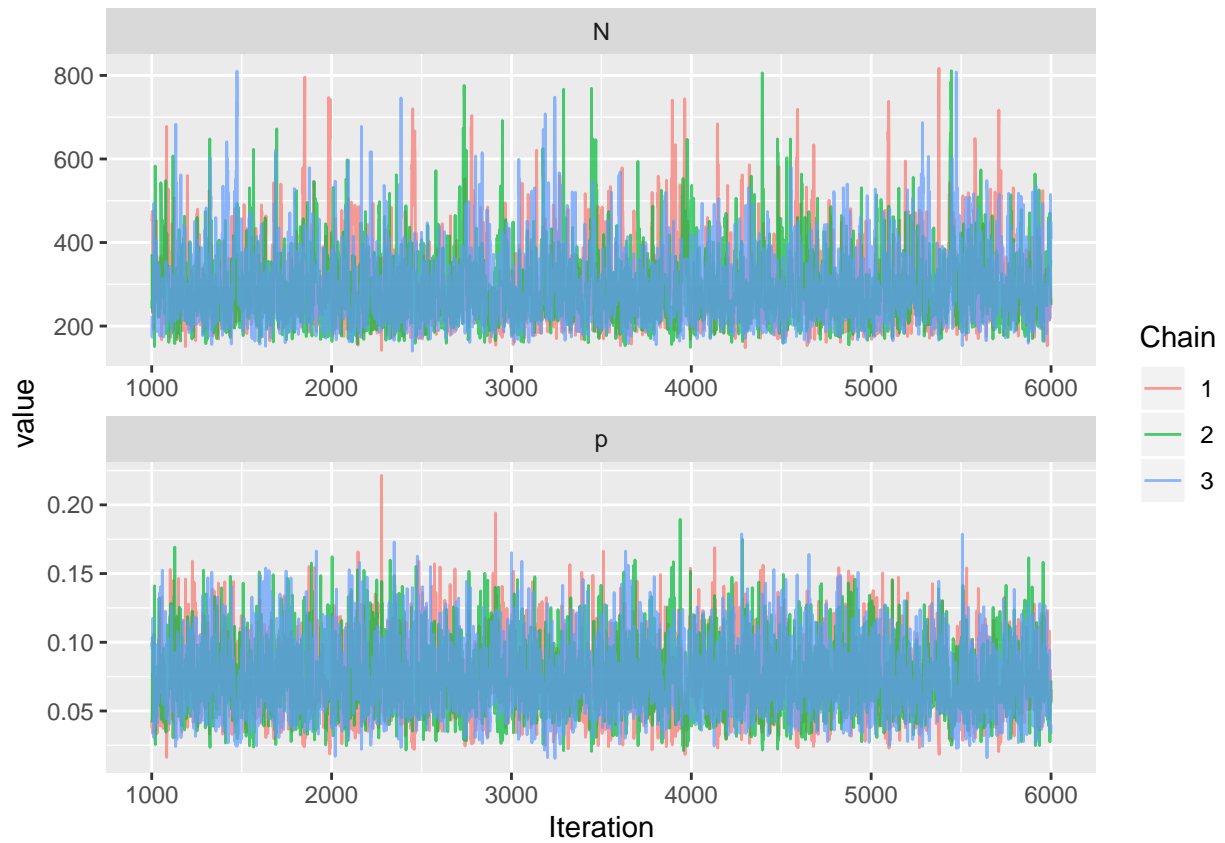
On peut obtenir des graphiques résumés facilement à l'aide du package `ggmcmc`.

Il faut pour cela, exporter les échantillons tirés dans la loi a posteriori dans le bon format

```
post <- ggs(S = post_sample)
```

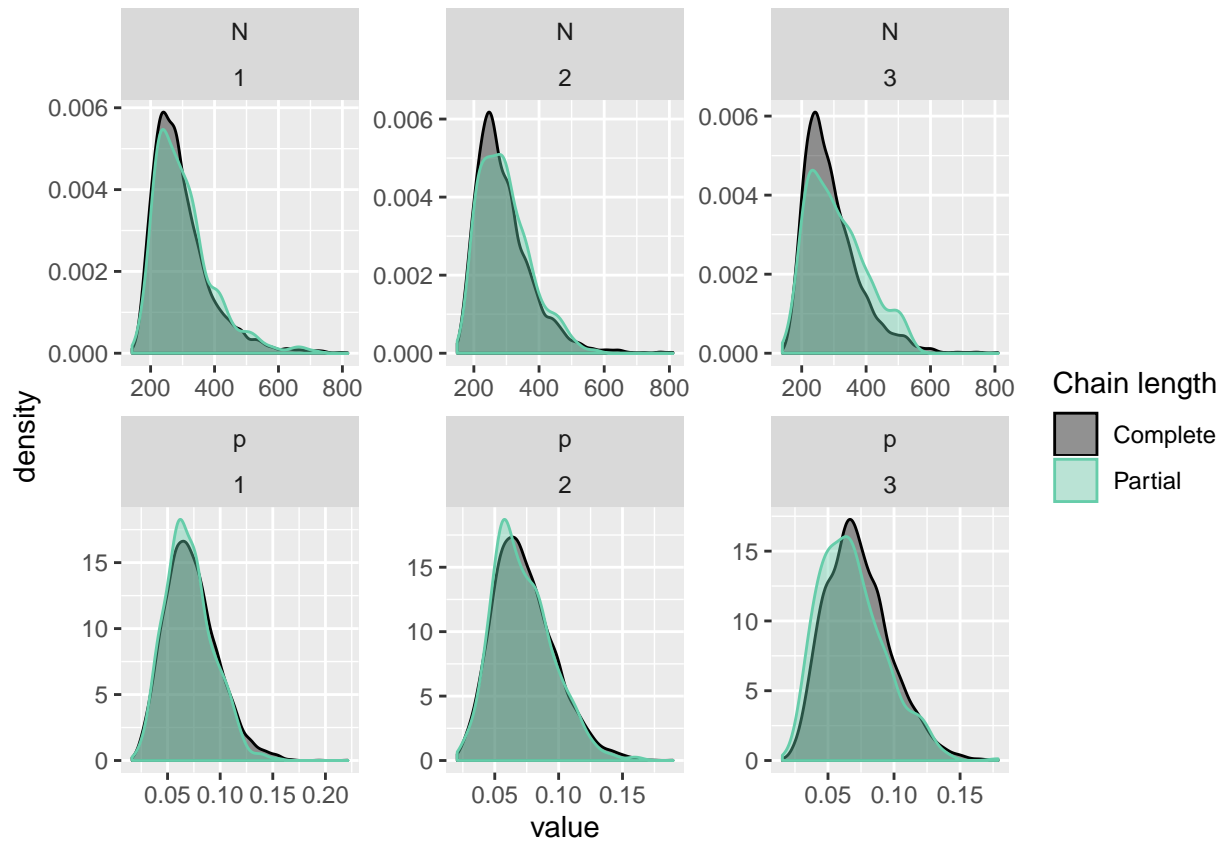
On peut ensuite, vérifier la convergence des chaînes MCMC en regardant l'historique

```
ggs_traceplot(post)
```



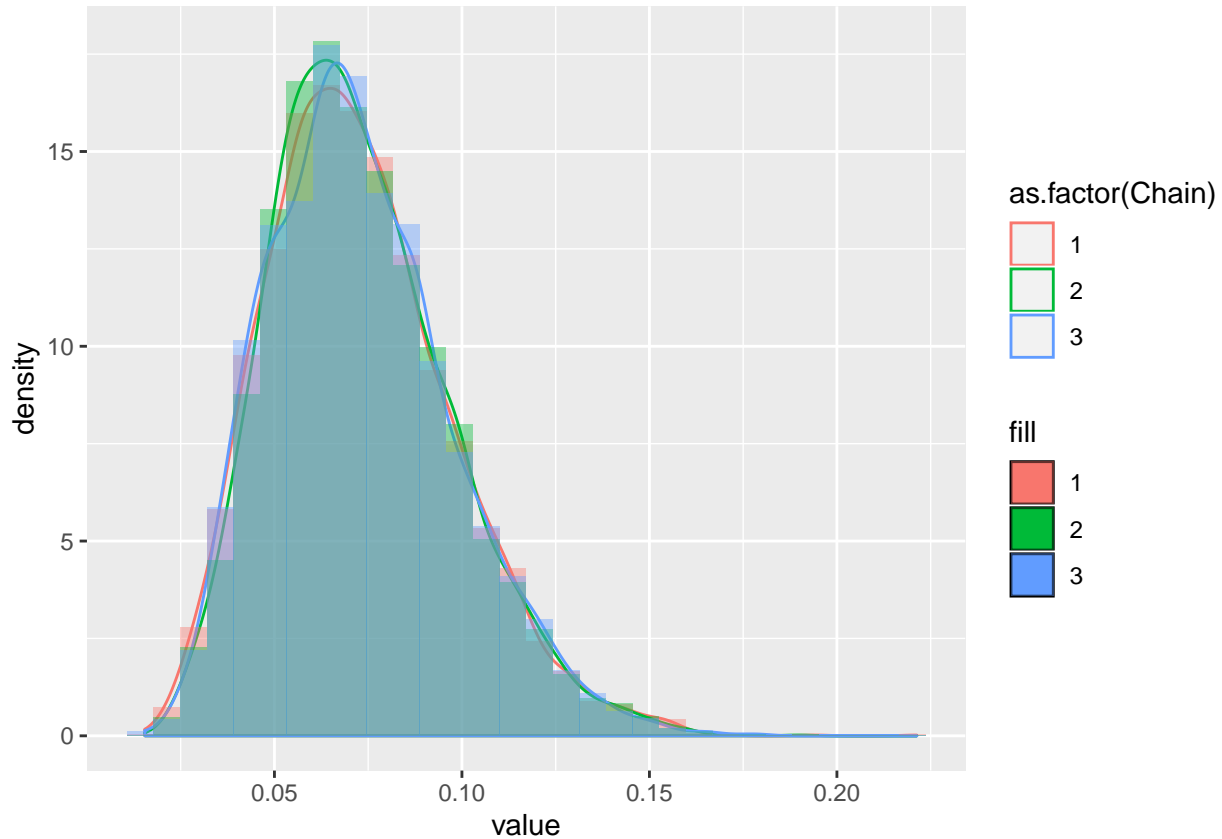
ou en comparant la distribution obtenue sur l'échantillon complet ou uniquement sur la dernière partie de l'échantillon

```
ggs_compare_partial(post)
```



Enfin on peut obtenir les lois a posteriori, sous forme d'histogramme

```
post_p <- post %>% filter(Parameter == 'p')
p_test <- ggs_histogram(post, family = 'p')
ggplot(post_p) + geom_density( aes(x=value, col=as.factor(Chain)) ) + geom_histogram(position = 'ident.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Enfin on peut facilement extraire des résumés numériques

```
post %>% group_by(Parameter) %>% summarize(m = mean(value), q025 = quantile(value, p = c(0.025)), q05 =
```

```
## # A tibble: 2 x 6
##   Parameter      m    q025    q05    q50    q95
##   <fct>      <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 N          293.   517   190   274   461
## 2 p           0.0729 0.128 0.0378 0.0702 0.117
```

Modélisation d'une unique observation

On ne considère dans un premier temps qu'une seule opération de recapture pour une année donnée

```
CMR_Davis %>% filter(Year == 2007, Transect == 1) -> CMR_2007_T1
CMR_2007_T1
```

```
##   Year Transect Repeat  M NM  n R  Site
## 1 2007         1      1 109 11 19 8 MM046M
```

On a marqué $m=109$ individus, 8 ont été capturés marqués et 11 individus non marqués ont été capturés.

But : Estimer la taille de la population.

Proposer un modèle permettant de le faire et le faire tourner dans jags.

Le modèle est donc

$$NM \sim \mathcal{B}(Size_{pop} - m, p), \quad R \sim \mathcal{B}(m, p)$$

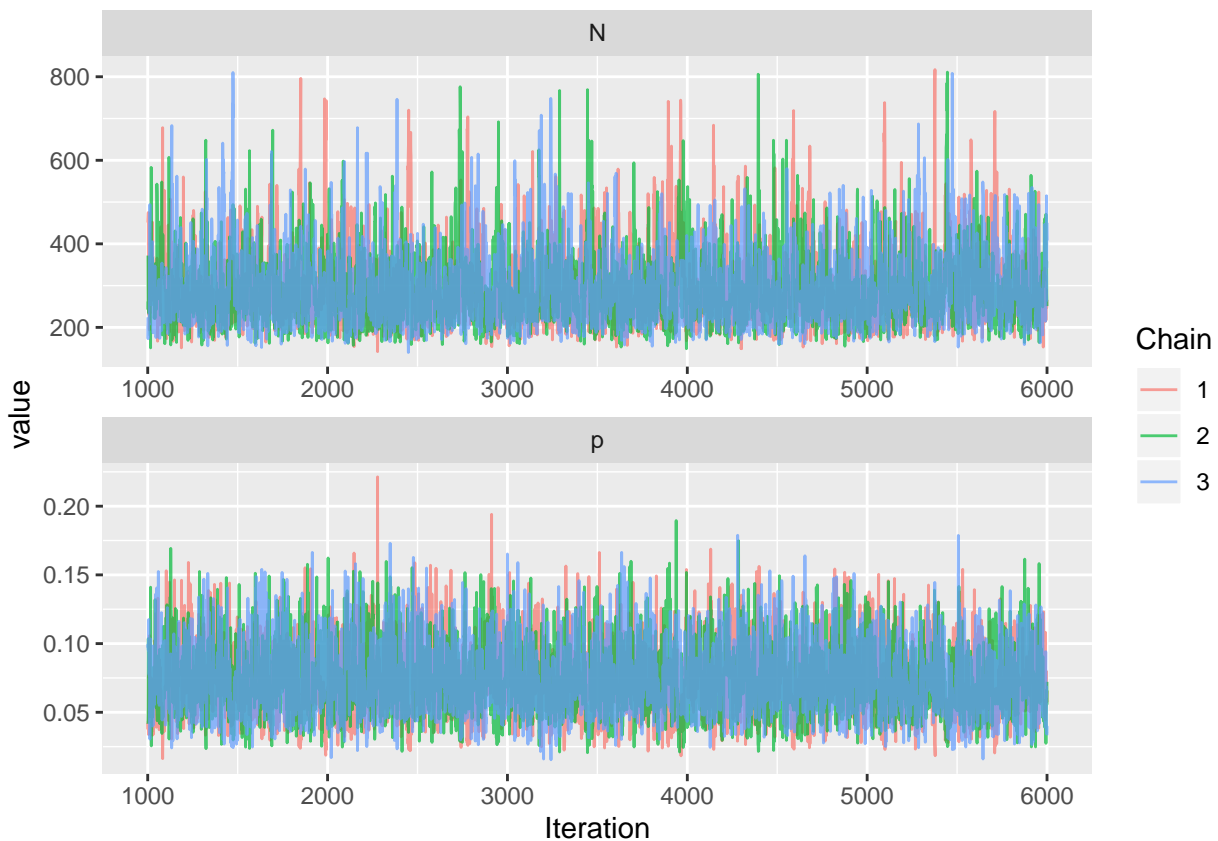
Prior pour $Size_{pop}$: exemple $Size_{pop} \sim \mathcal{P}(mean_{prior})$.

Puisqu'on ne précise pas de valeurs initiale pour les paramètres du modèle, l'algorithme est initialisé automatiquement par un tirage dans la loi a priori.

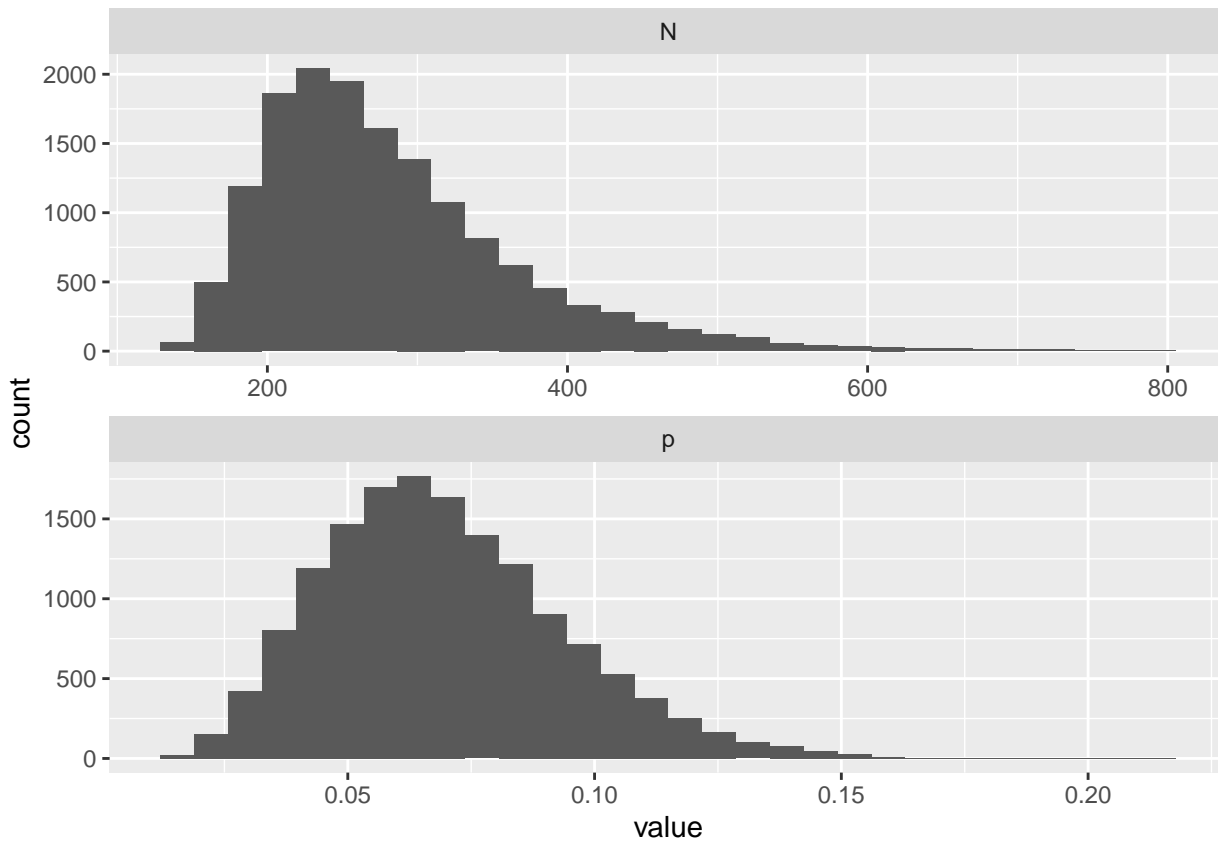
```
data.list <- list( M = CMR_2007_T1$M,  
  NM = CMR_2007_T1$NM,  
  R = CMR_2007_T1$R)  
  
mjags <- jags.model(file = 'modelCMR.txt',  
  data = data.list, n.chains = 3)
```

```
## Compiling model graph  
##   Resolving undeclared variables  
##   Allocating nodes  
## Graph information:  
##   Observed stochastic nodes: 2  
##   Unobserved stochastic nodes: 2  
##   Total graph size: 10  
##  
## Initializing model
```

```
post_sample <- coda.samples(mjags, variable.names = c('p', 'N'),  
  n.iter = 5000)  
ggs_traceplot(post)
```



```
ggs_histogram(post)
```

Modèle pour l'année 2007

On écrit l'emodèle suivant

$$NM_k \stackrel{i.i.d}{\sim} \mathcal{B}(Size_{pop} - m, p), \quad R_k \stackrel{i.i.d}{\sim} \mathcal{B}(m, p)$$

Soit le modèle suivant dans pour jags

```
cat(readLines( 'modelCMR_2.txt'), sep = '\n')
```

```
model{
  ## prior
  p ~ dbeta(1,1)
  Ncont ~ dunif(0, 1e6)
  N <- round(Ncont)

  ## modele d'observation

  #Nobs nombre total d'observations
  for( k in 1:Nobs){
    R[k] ~ dbin(p, M[k])
    NM[k] ~ dbin(p, N- M[k])
  }
}
```

```

CMR_Davis %>% filter(Year == 2007) -> CMR_Davis_2007

data.list2 <- list( M = CMR_Davis_2007$M,
  NM = CMR_Davis_2007$NM,
  R = CMR_Davis_2007$R,
  NObs = nrow(CMR_Davis_2007))

mjags2 <- jags.model(file = 'modelCMR_2.txt',
  data = data.list2, n.chains = 3)

## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 16
##   Unobserved stochastic nodes: 2
##   Total graph size: 32
##
## Initializing model

post_sample2 <- coda.samples(mjags2, variable.names = c('p', 'N'),
  n.iter = 5000)

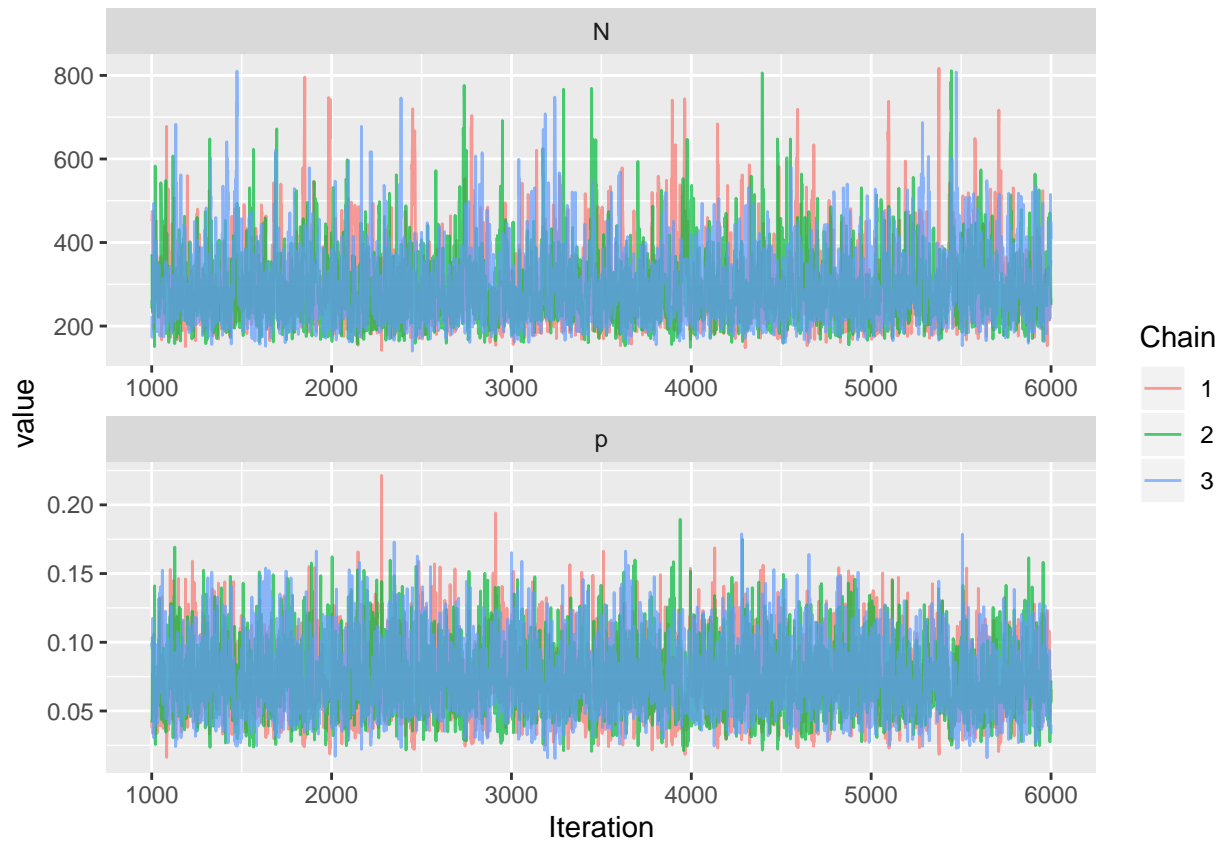
```

Verification de la convergence

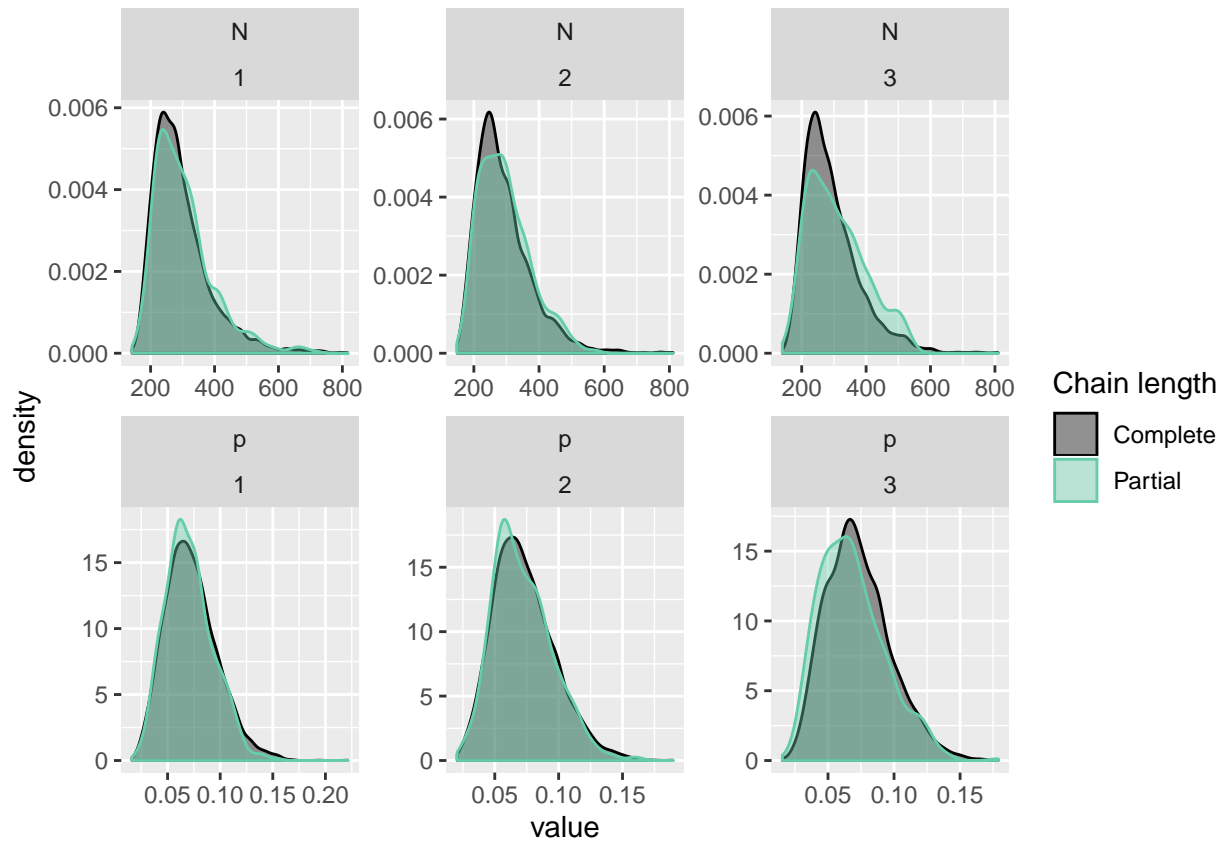
```

post2 <- ggs(post_sample2)
ggs_traceplot(post)

```

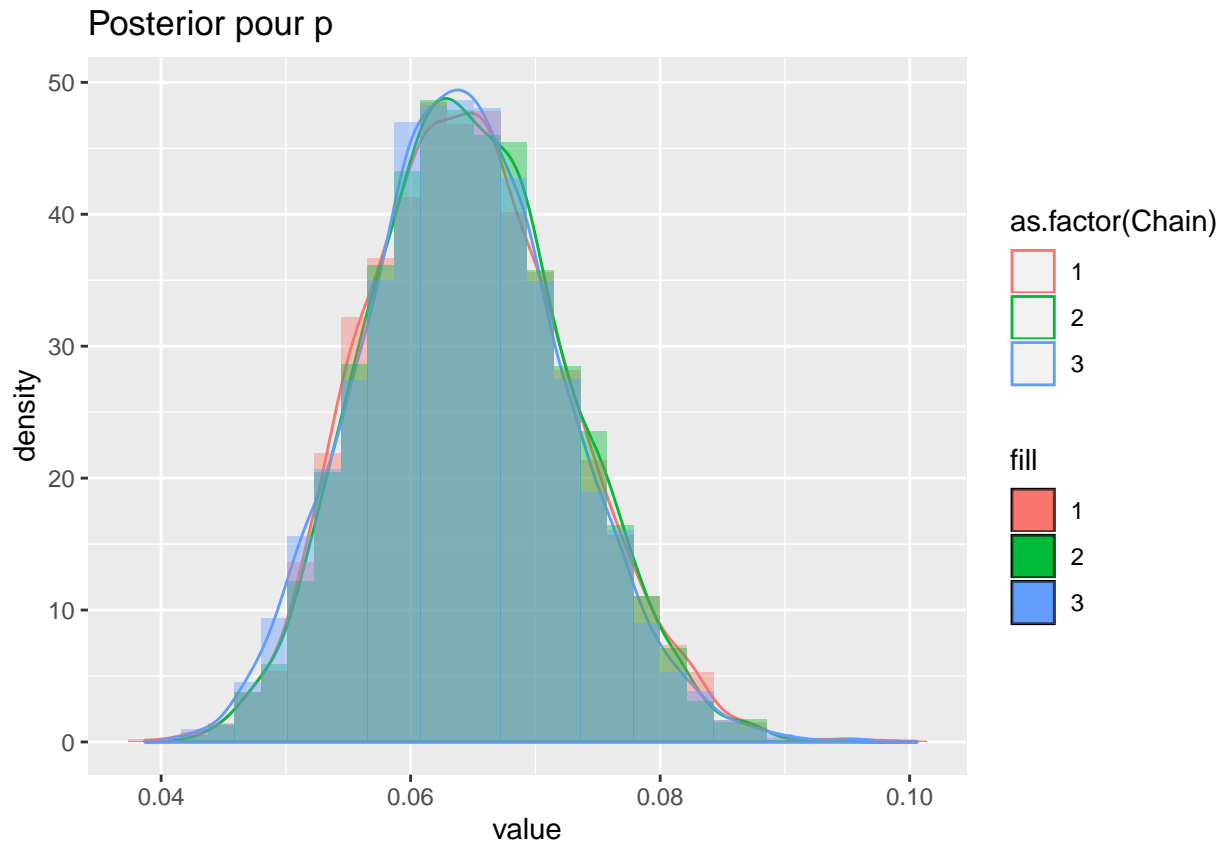


```
ggs_compare_partial(post)
```

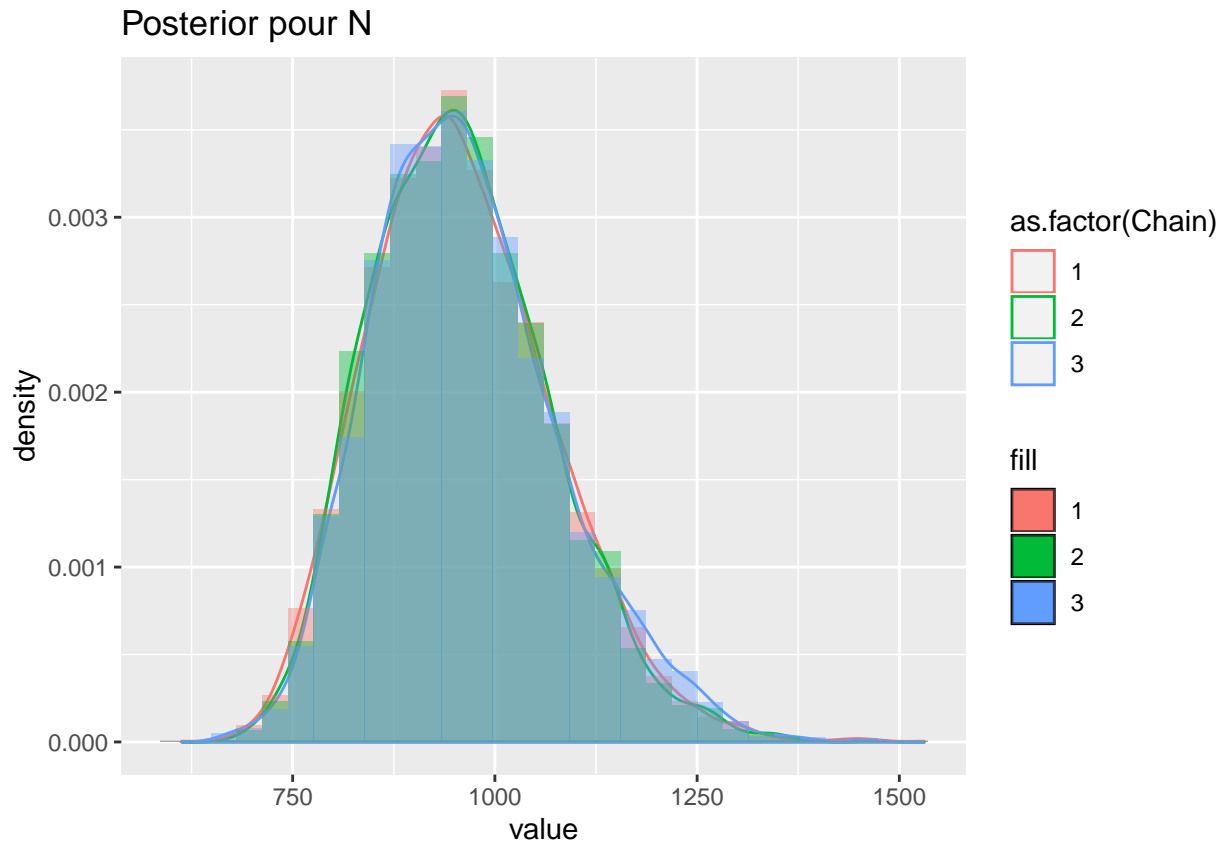


les lois a posteriori

```
post2 %>% filter(Parameter == 'p') %>% ggplot() + geom_density( aes(x=value, col=as.factor(Chain)) ) +
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
post2 %>% filter(Parameter == 'N') %>% ggplot() + geom_density( aes(x=value, col=as.factor(Chain)) ) +
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Modeleavec prise en compte de l'année

On suppose ici que la taille de la population varie indépendamment d'une année sur l'autre tandis que la probabilité de capture est la même chaque année

On transforme l'année en une variable variant de 1 à nombre d'années observées

```
CMR_Davis %>% mutate(Year_num = as.numeric(as.factor(CMR_Davis$Year))) -> CMR_Davis
```

```
data.list3 <- list(NM = CMR_Davis$NM, M=CMR_Davis$M, R=CMR_Davis$R,
  NObs = nrow(CMR_Davis),
  NYear = n_distinct(CMR_Davis$Year),
  Year_num = CMR_Davis$Year_num
)
```

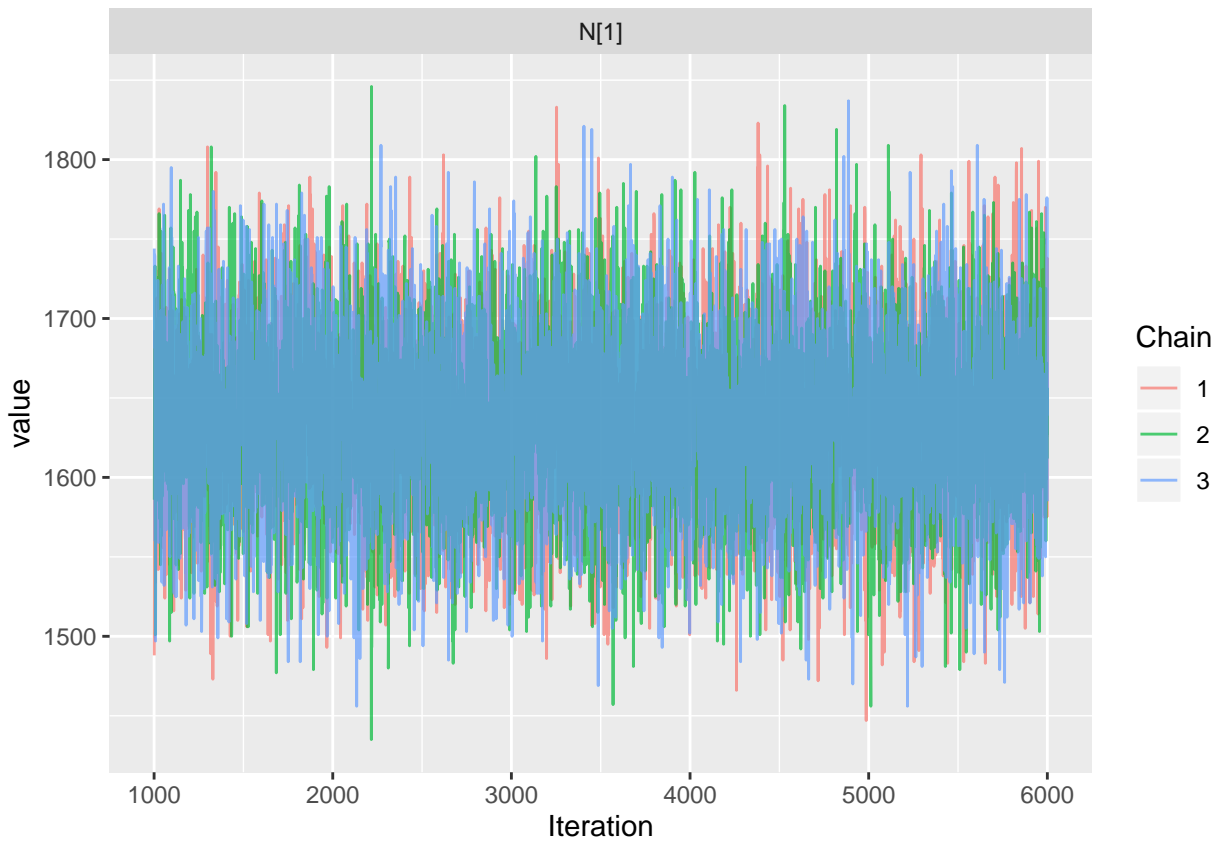
```
mjags3 <- jags.model(file = 'modelCMR_3.txt',
  data = data.list3,
  n.chains = 3, )
```

```
## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 446
##   Unobserved stochastic nodes: 11
##   Total graph size: 928
##
```

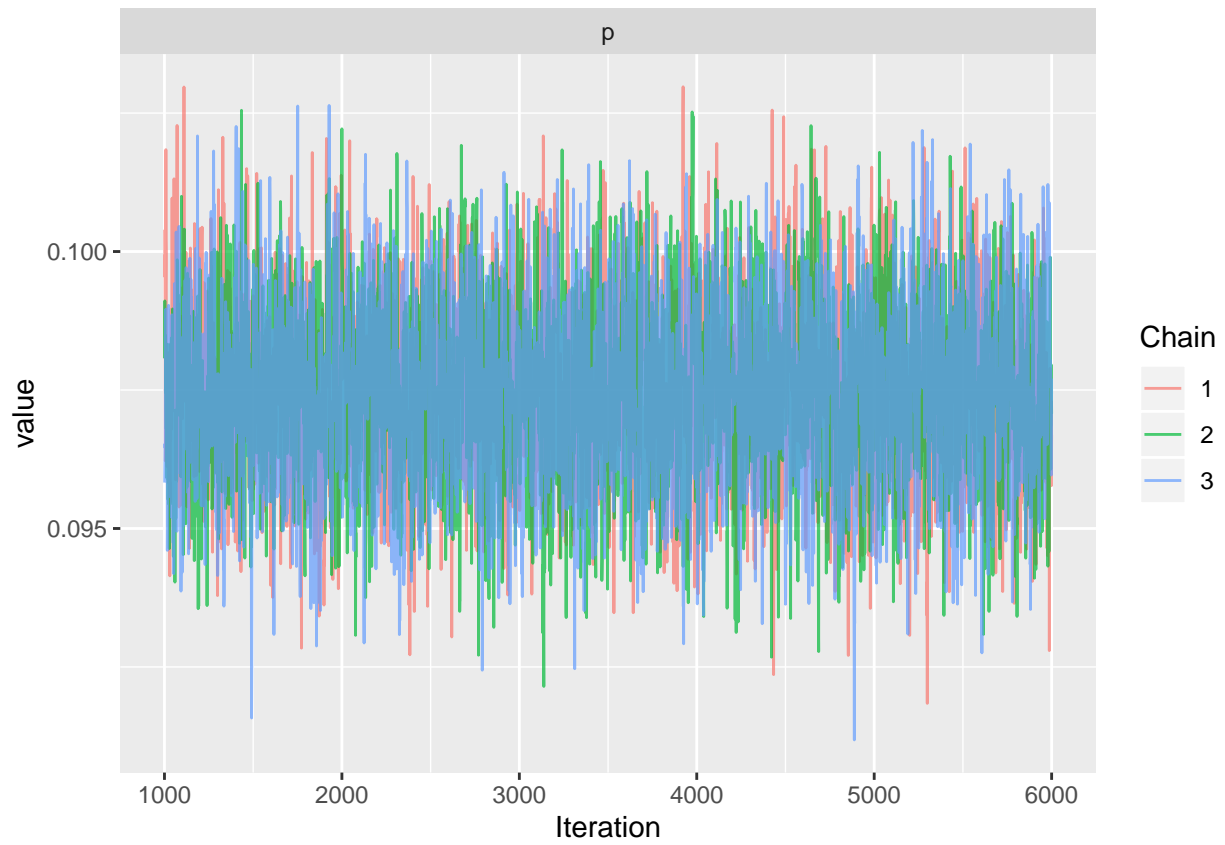
```
## Initializing model
post_sample3 <- coda.samples(mjags3, variable.names = c('p', 'N'),
                             n.iter = 5000)

post3 <- ggs(S = post_sample3)

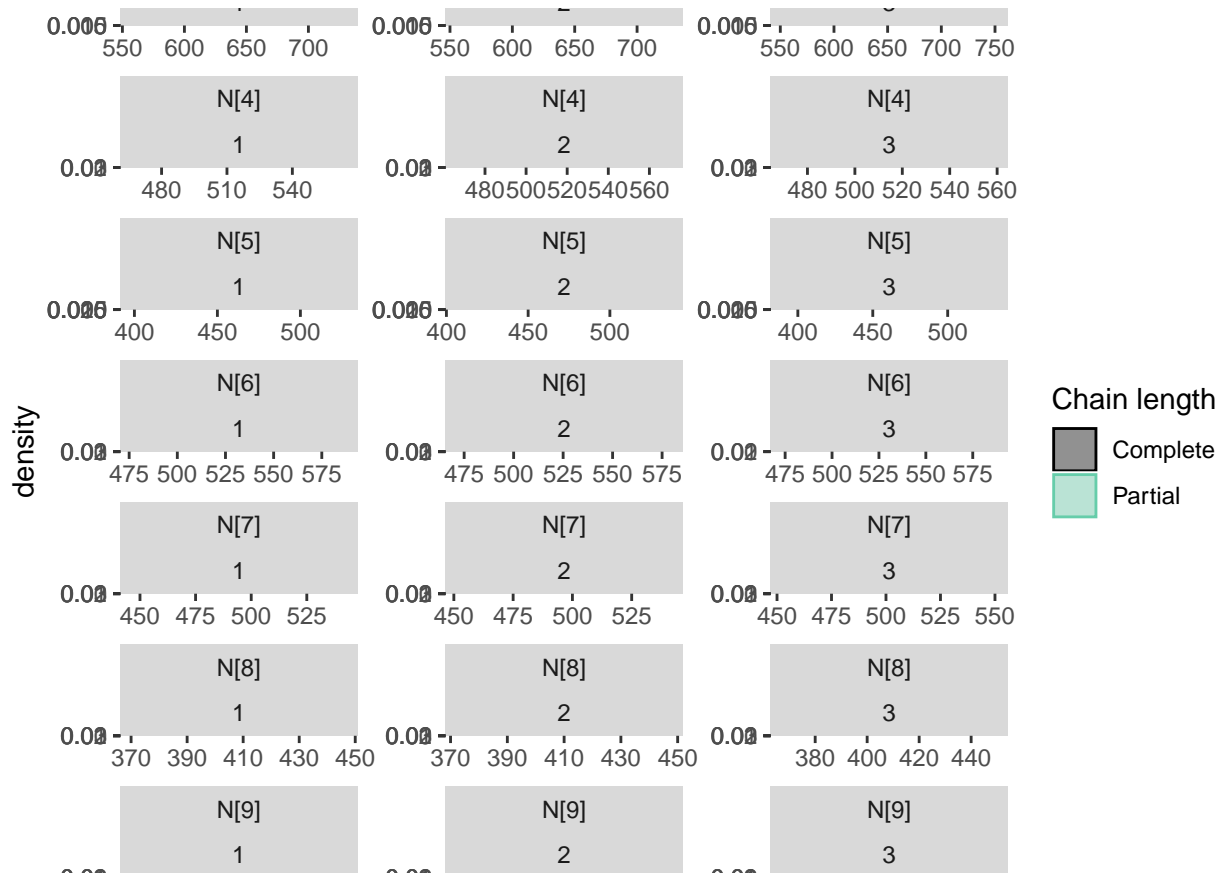
ggs_traceplot(post3, family="N\\[1\\]")
```



```
ggs_traceplot(post3, family="p")
```



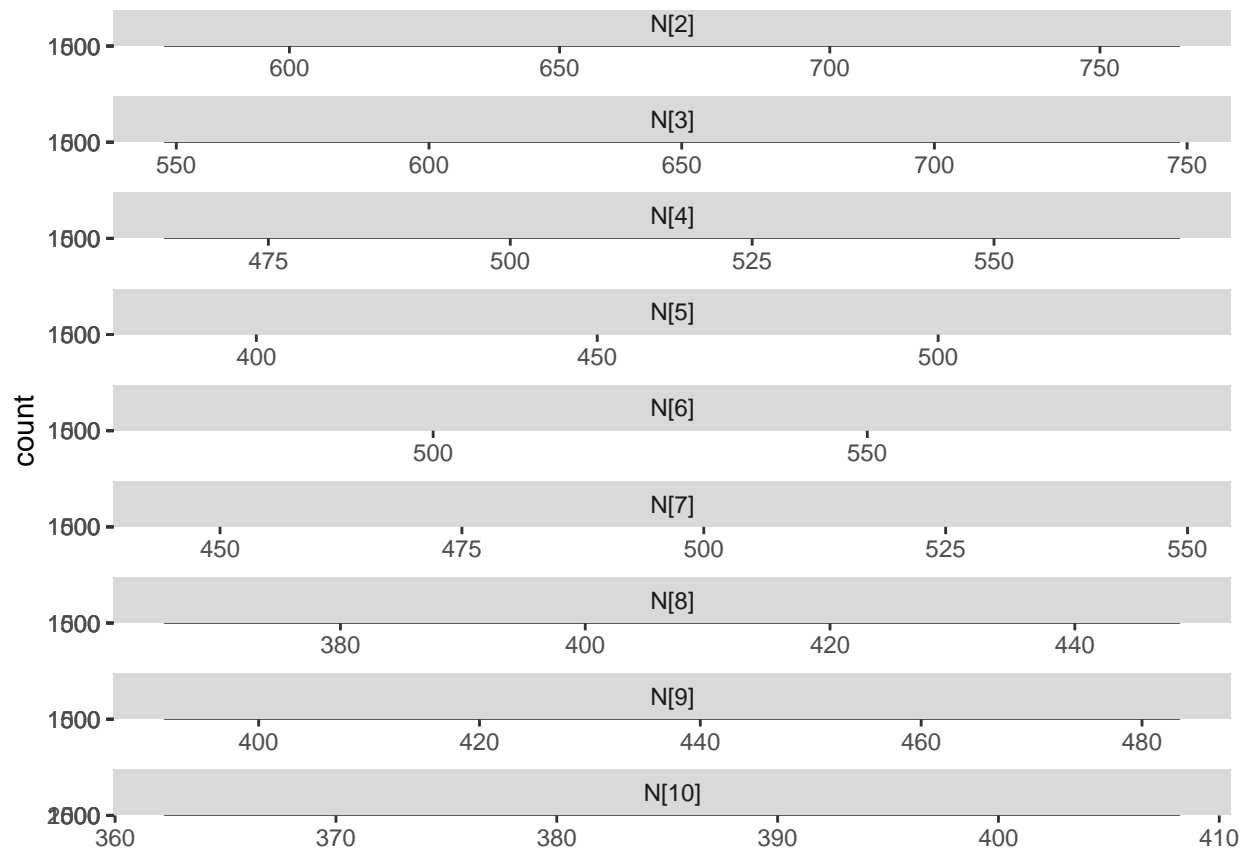
```
ggs_compare_partial(post3)
```

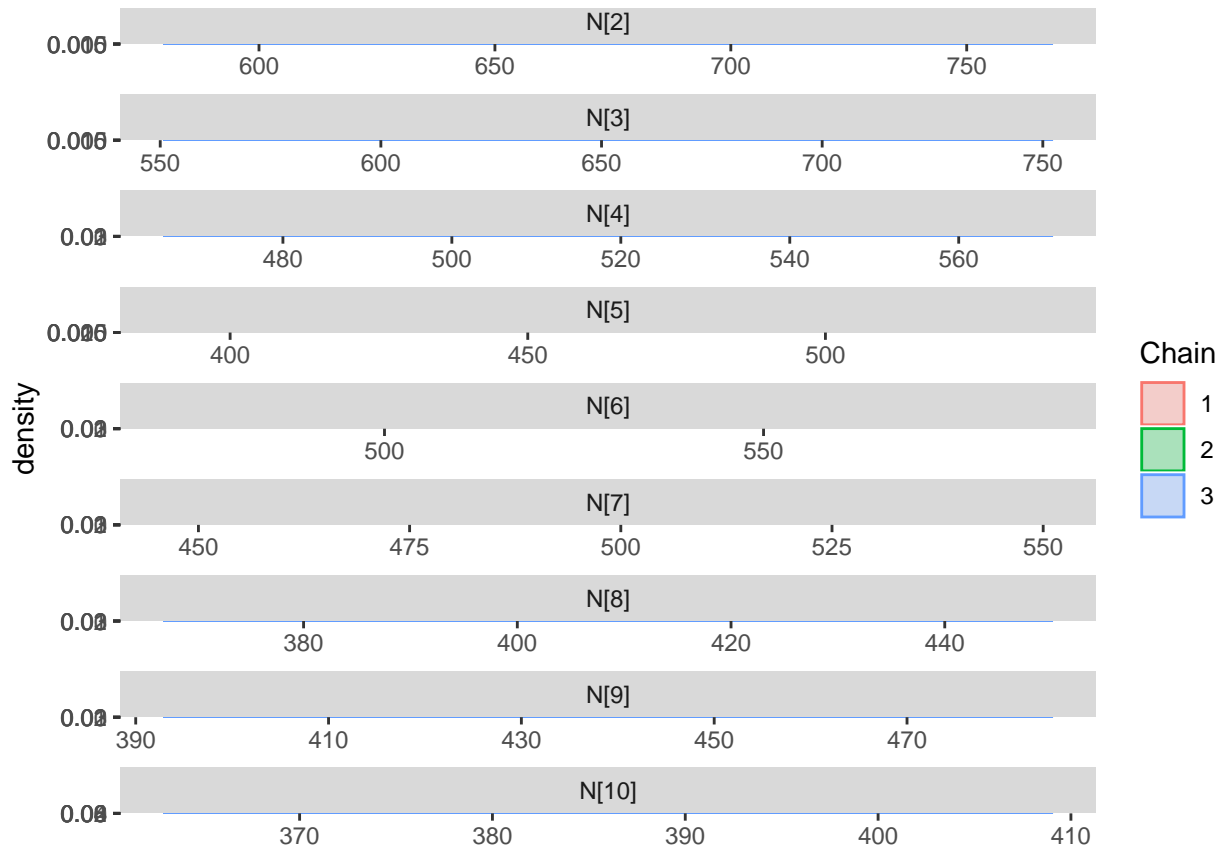
```
post3 %>% group_by(Parameter) %>% summarise(m = mean(value))
```

```
## # A tibble: 11 x 2
##   Parameter      m
##   <fct>         <dbl>
## 1 N[1]         1635.
## 2 N[2]          667.
## 3 N[3]          638.
## 4 N[4]          513.
## 5 N[5]          460.
## 6 N[6]          528.
## 7 N[7]          495.
## 8 N[8]          407.
## 9 N[9]          434.
## 10 N[10]       384.
## 11 p           0.0975
```

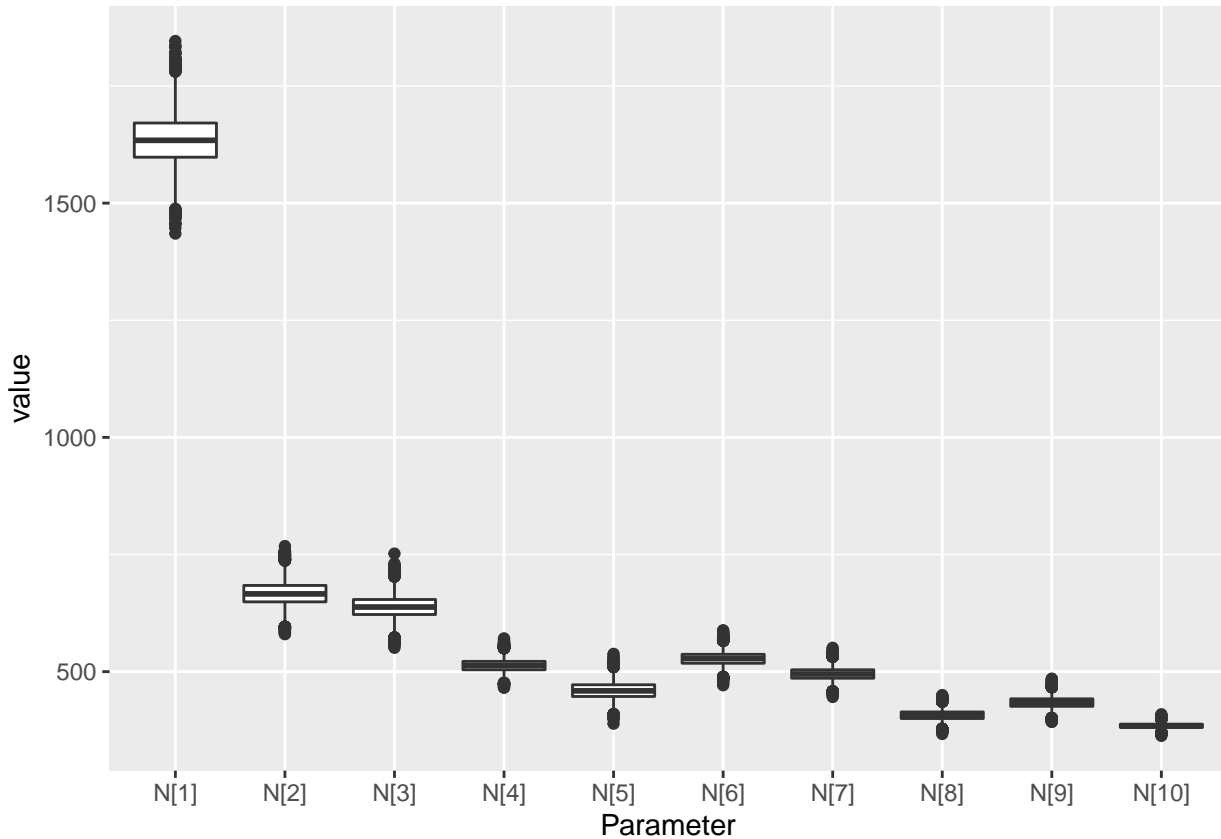
```
ggs_histogram((post3))
```



```
ggs_density((post3))
```



```
post3 %>% filter(stringr::str_detect(string = post3$Parameter, pattern = 'N')) %>% ggplot() + geom_boxplot()
```



Modèle avec évolution annuelle de la taille de la population et de la propabilité de capture

```

mjags4 <- jags.model(file = 'modelCMR_4.txt',
  data = data.list3,
  n.chains = 3, )

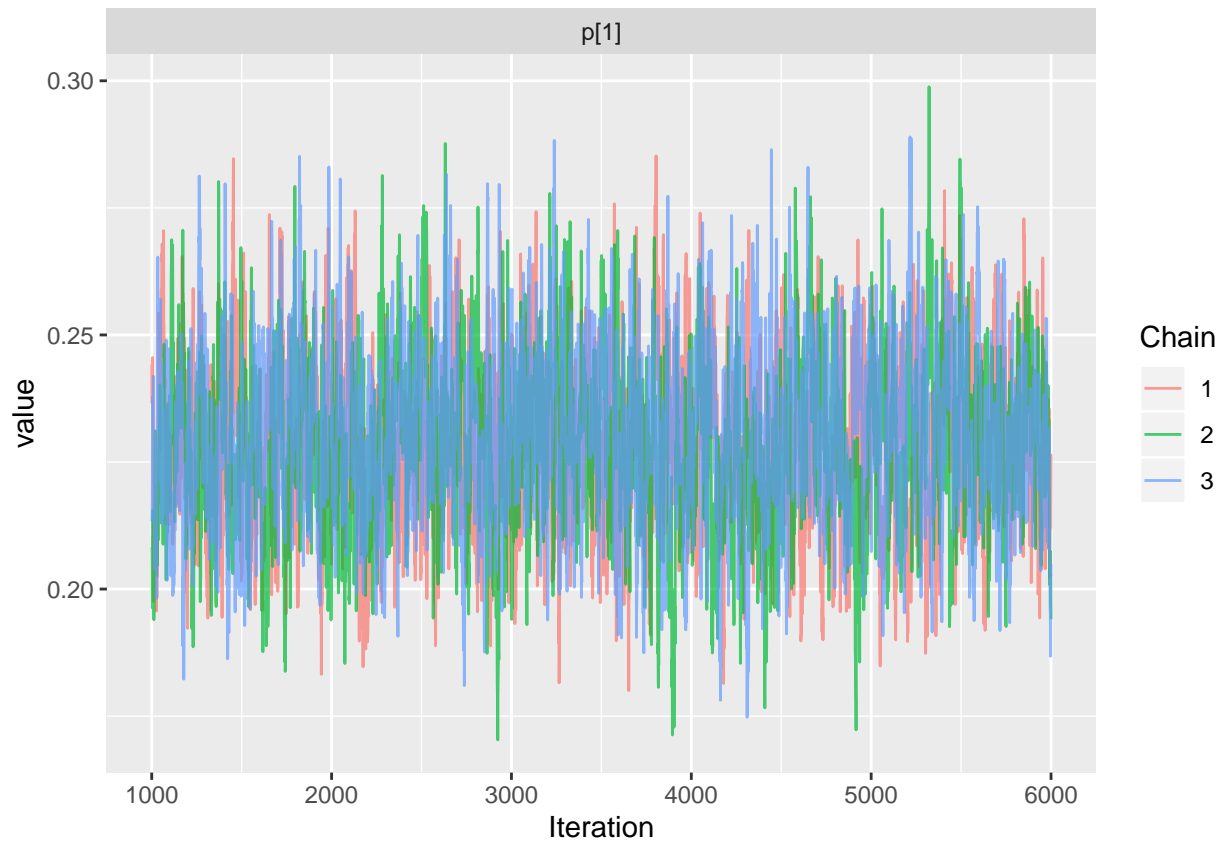
## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 446
##   Unobserved stochastic nodes: 20
##   Total graph size: 937
##
## Initializing model

post_sample4 <- coda.samples(mjags4, variable.names = c('p', 'N'),
  n.iter = 5000)

post4 <- ggs(S = post_sample4)

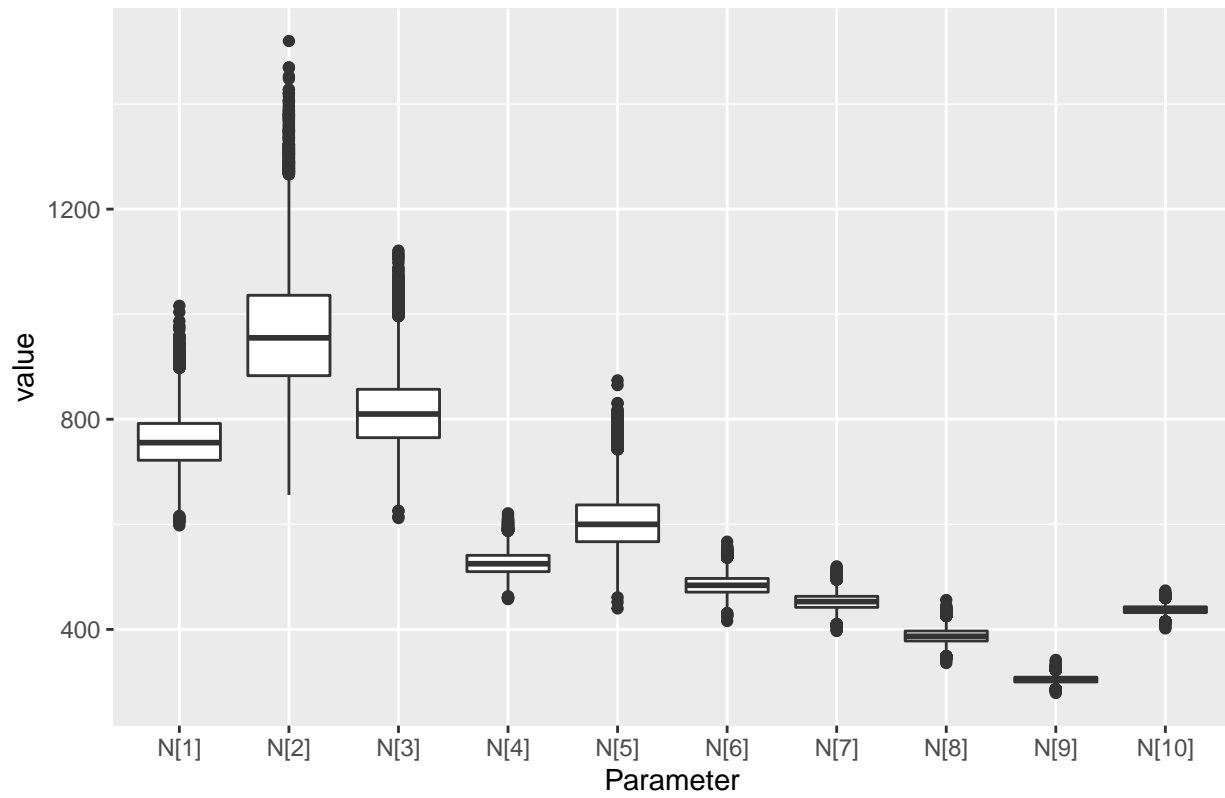
ggs_traceplot(post4, family="p\\[1\\]")

```



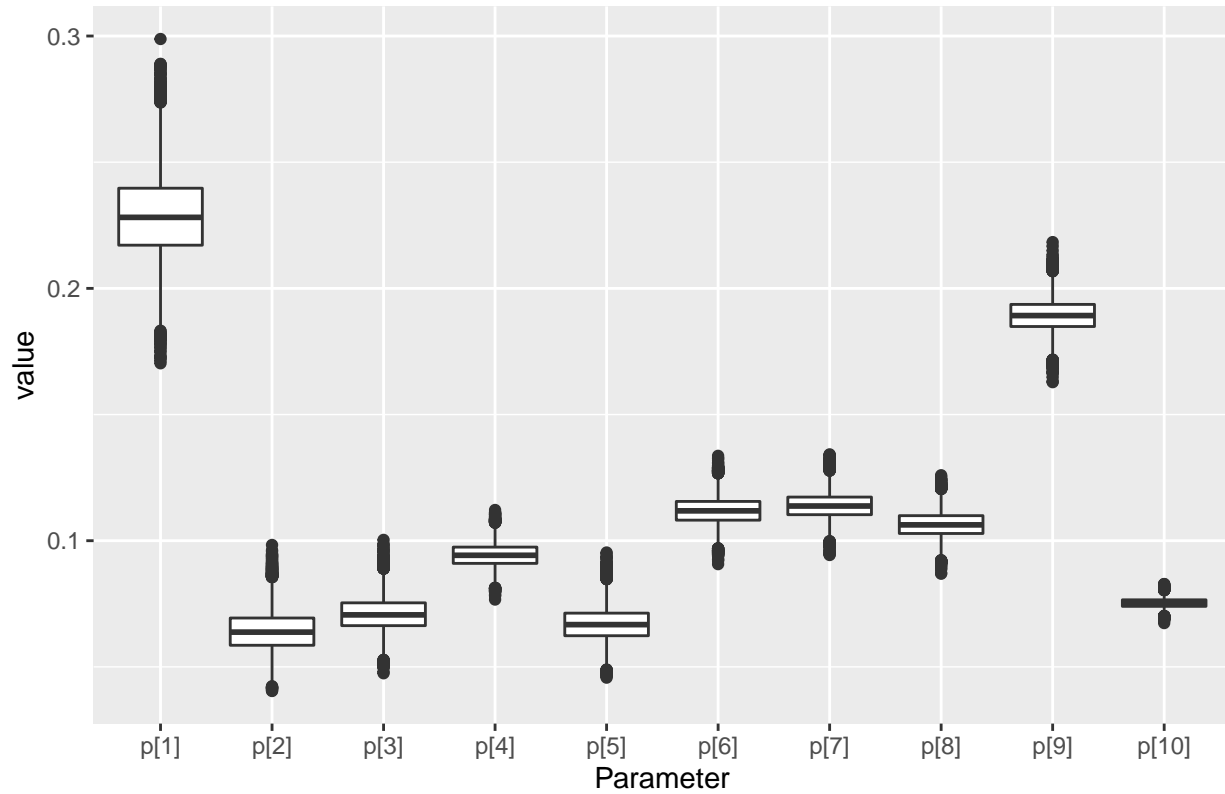
```
post4 %>% filter(stringr::str_detect(string = Parameter, pattern = 'N\\[') ) %>% ggplot() + geom_boxpl
```

Evolution de la taille de la population



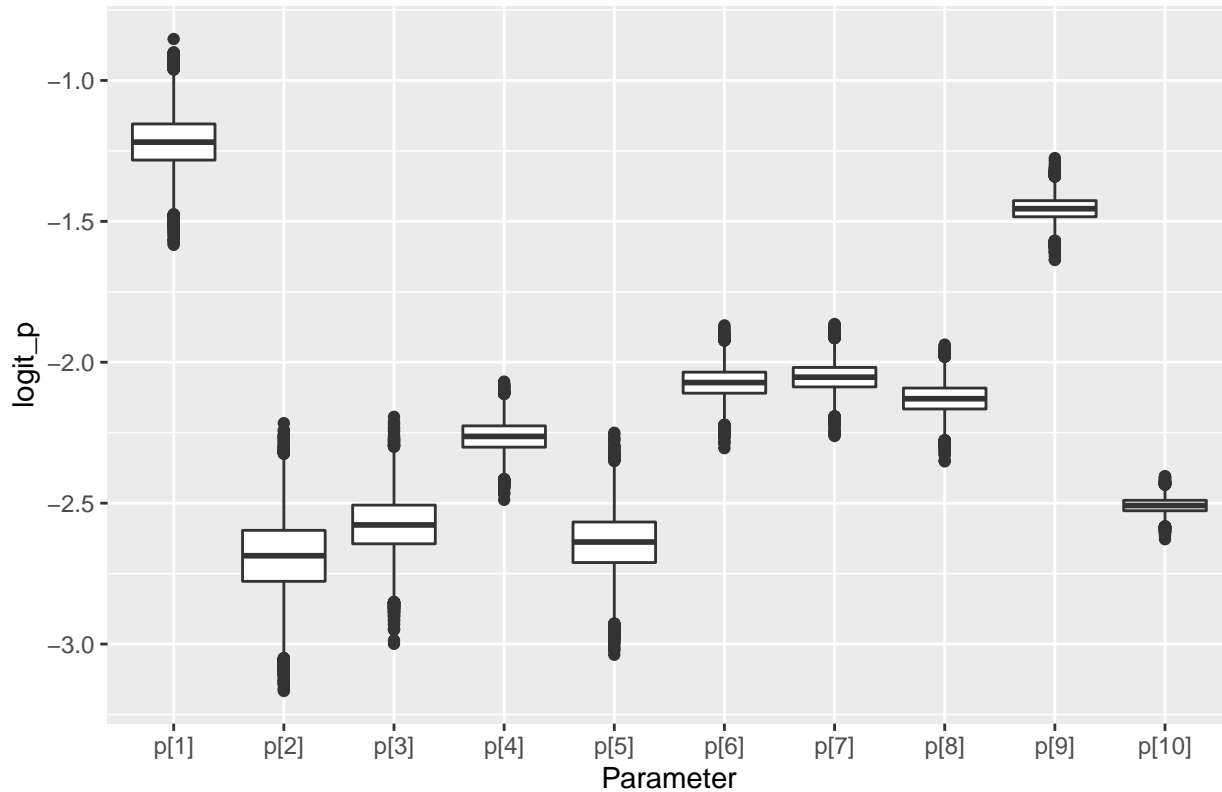
```
post4 %>% filter(stringr::str_detect(string = Parameter, pattern = 'p\\[')) %>% ggplot() + geom_boxp
```

Evolution de la taille de la probabilité d'être vu



```
post4 %>% filter(stringr::str_detect(string = post4$Parameter, pattern = 'p\\[')) %>% mutate(logit_p
```

Evolution de la taille de la probabilité d'être vu (logit scale)



Modele hiérarchique pour l'année

Un modèle dans lequel l'effet de l'année est traité comme aléatoire = modèle hiérarchique

```
cat(readLines( 'modelCMR_5.txt'), sep = '\n')
```

```
model{  
  
  ## prior sur mu et tauA  
  mu ~ dnorm(0, 1e-4)  
  tauA ~ dgamma(1e-3, 1e-3)  
  
  ## alternative  
  ## sigmaA ~ dunif(0,100)  
  ## tauA <- 1/(sigmaA*sigmaA)  
  ##  
  
  ## attention y varie de 1 à .....  
  
  for(y in 1:NYear){  
  
    ##tauA la precision tauA = 1/varA  
    A[y] ~ dnorm(0, tauA)  
    p[y] <- 1 / (1+ exp(- (mu + A[y] )))  
  
    Ncont[y] ~ dunif(0, 1e6)  
    N[y] <- round(Ncont[y])  
  }  
}
```



```

}

## modele d'observation CMR

#Nobs nombre total d'observations
for( k in 1:NObs){
  R[k] ~ dbin(p[Year_num[k] ], M[k])
  NM[k] ~ dbin(p[Year_num[k] ], N[ Year_num[k] ]- M[k])
}
}

mjags5 <- jags.model(file = 'modelCMR_5.txt', data = data.list3, n.chains = 3 )

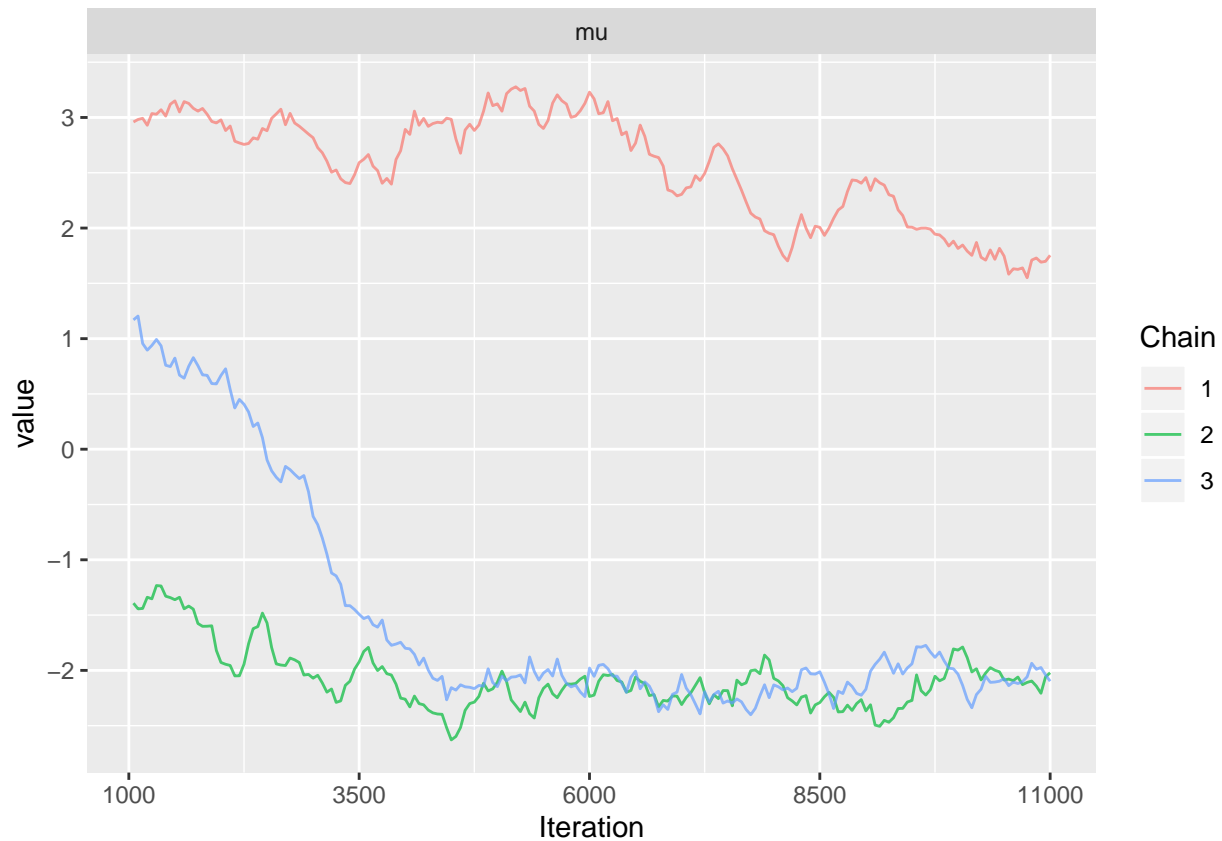
## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 446
##   Unobserved stochastic nodes: 22
##   Total graph size: 991
##
## Initializing model

post_sample5 <- coda.samples(mjags5, variable.names = c('p', 'N', 'mu', 'tauA'),
                             n.iter = 10000, thin = 50)

post5 <- ggs(S = post_sample5)

ggs_traceplot(post5, family = 'mu')

```



La convergence est difficile. Elle peut être améliorée en reparamétrisant le modèle

$$A_y \sim \mathcal{N}(\mu, \sigma_A^2)$$

au lieu de

$$A_y = \mu + E_y, \quad E_y \sim \mathcal{N}(0, \sigma_A^2)$$

```
mjags5_bis <- jags.model(file = 'modelCMR_5_bis.txt', data = data.list3, n.chains = 3 )

## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 446
##   Unobserved stochastic nodes: 22
##   Total graph size: 981
##
## Initializing model

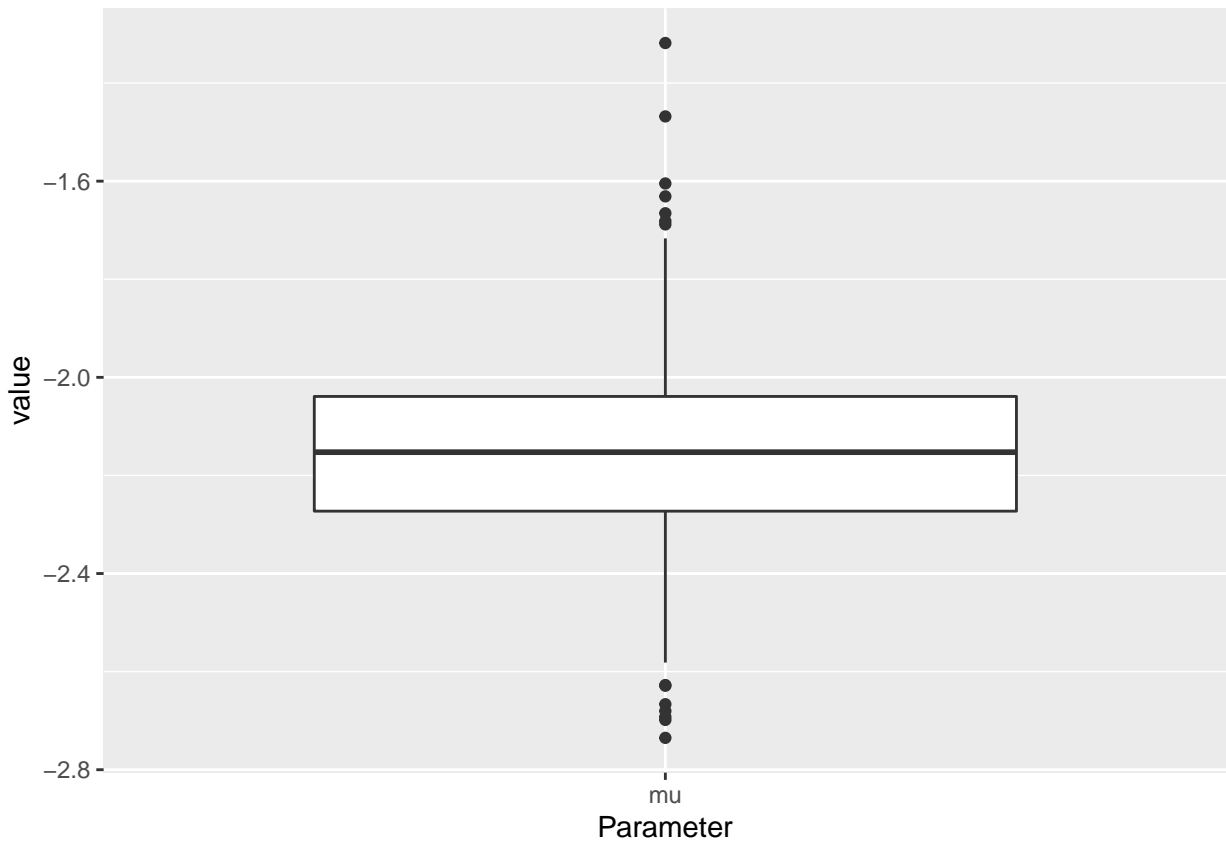
post_sample5 <- coda.samples(mjags5_bis, variable.names = c('p', 'N', 'mu', 'tauA'),
                             n.iter = 10000, thin = 50)

post5 <- ggs(S = post_sample5)

ggs_traceplot(post5, family = 'mu')
```

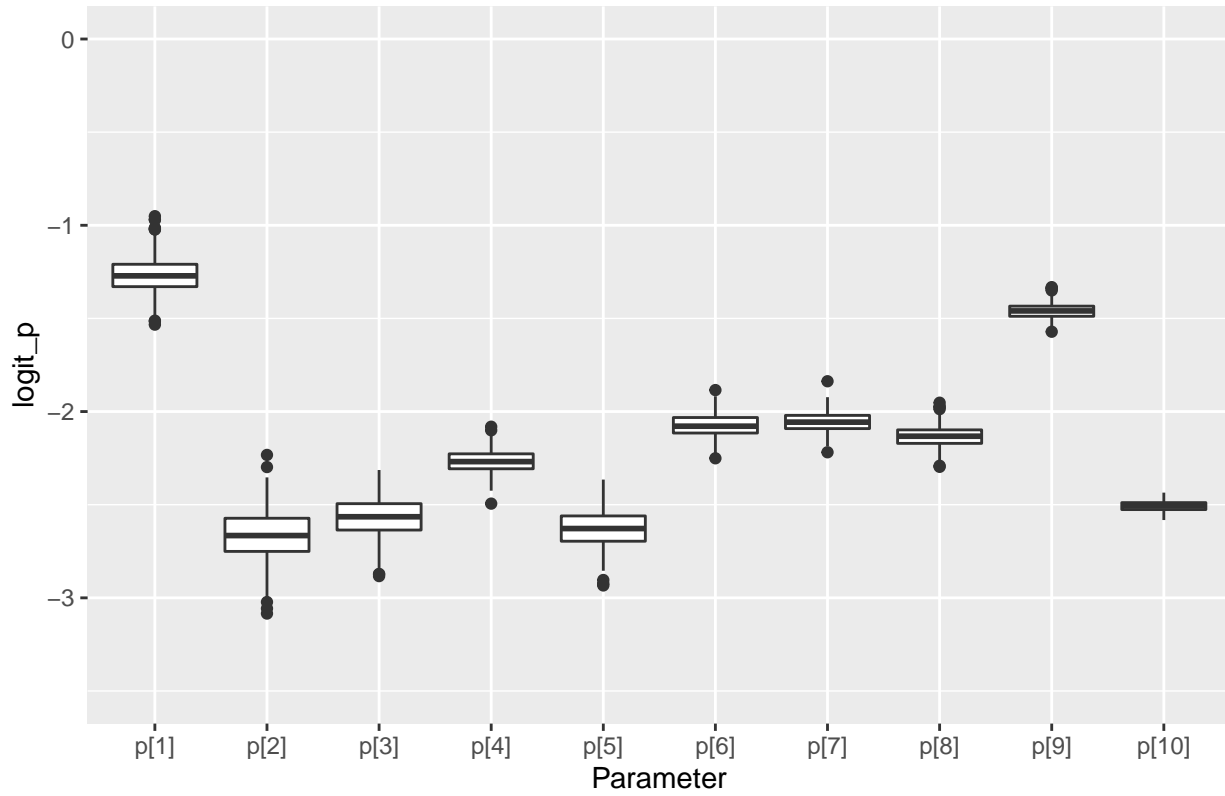


```
post5 %>% filter(stringr::str_detect(string = post5$Parameter, pattern = 'mu')) %>% ggplot() + geom_l
```



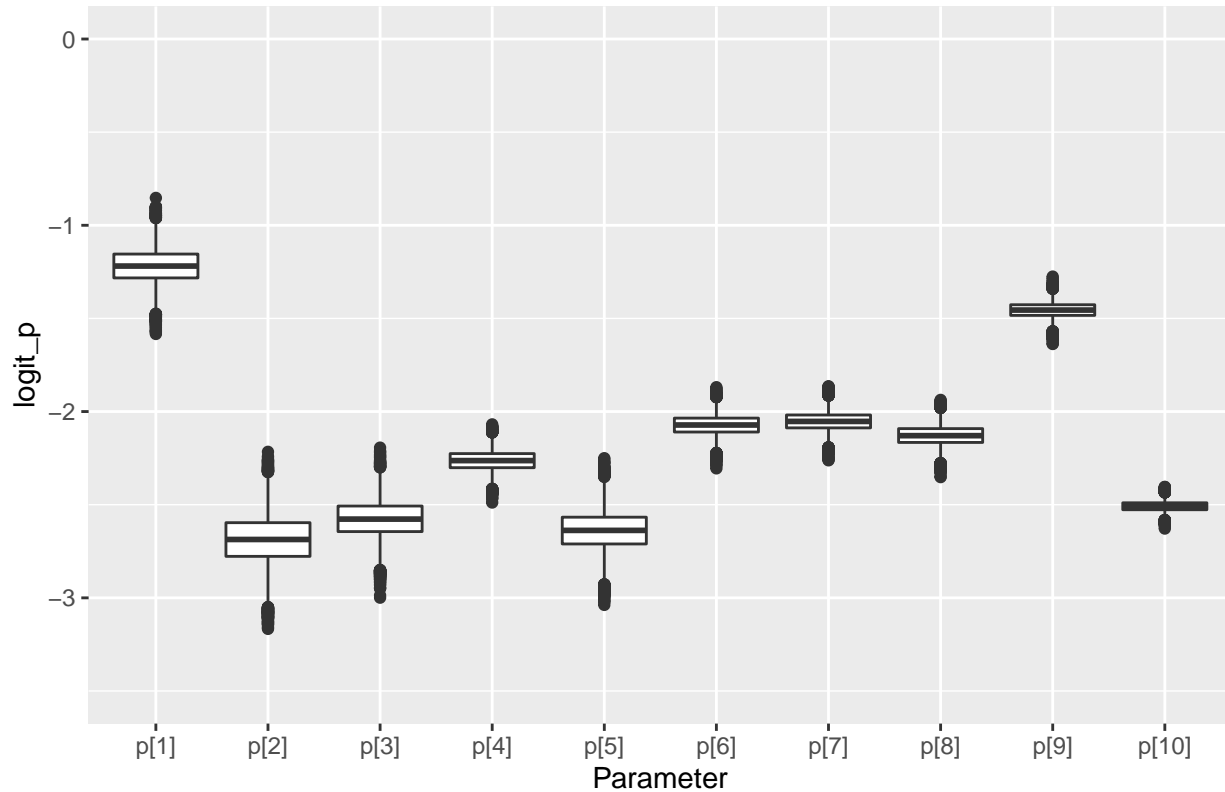
```
post5 %>% filter(stringr::str_detect(string = Parameter, pattern = 'p\\[') ) %>% mutate(logit_p = log
```

Probabilité de capture dans le modèle hiérarchique (logit scale)



```
post4 %>% filter(stringr::str_detect(string = Parameter, pattern = 'p\\[')) %>% mutate(logit_p = log
```

Probabilité de capture dans le modèle indépendant (logit scale)



Estimation de la probabilité de détection par comptage direct

Les données de comptage direct et leur version interprétée dans l'échelle 'Year_num'.

```
CD <- read.table('../Bayesien/PupCount-TD/data/CDDirectCount_Davis.csv', header = TRUE, sep = ',')
```

```
CD %>% mutate(Year_num_cd = ifelse(Year >=2007 , Year -2005, NA )) -> CD
```

Le modèle M7 intègre ces données

```
cat(readLines( 'modelCMR_7.txt'), sep = '\n')
```

```
model{
```

```
## prior sur mu et tauA  
mu ~ dnorm(0, 1e-4)  
tauA ~ dgamma(1e-3, 1e-3)
```

```
## prior sur muc et tauAc  
muc ~ dnorm(0, 1e-4)  
tauAc ~ dgamma(0.001, 0.001)
```

```
## alternative  
## sigmaA ~ dunif(0,100)  
## tauA <- 1/(sigmaA*sigmaA)  
##
```

```

## attention y varie de 1 a .....

for(y in 1:NYear){

  ##tauA la precision tauA = 1/varA
  A[y] ~ dnorm(mu, tauA)
  p[y] <- 1 / (1+ exp(- ( A[y] )))

  Ncont[y] ~ dunif(0, 1000000)
  N[y] <- round(Ncont[y])

  ##tauA la precision tauA = 1/varA
  Ac[y] ~ dnorm(muc, tauAc)
  pc[y] <- 1 / (1+ exp(- ( Ac[y] )))

}

## modele d'observation CMR

#Nobs nombre total d'observations
for( k in 1:NObs){
  R[k] ~ dbin(p[Year_num[k] ], M[k])
  Nprov[k] <- N[ Year_num[k] ]- M[k]
  NM[k] ~ dbin(p[Year_num[k] ], Nprov[k])
}

## modele d'observation Comptage direct
for( k in 1:NObs_direct){
  C[k] ~ dbin(pc[ Year_num_direct[k]], N[ Year_num_direct[k] ])
}
}

data.list7 <- list(NM = CMR_Davis$NM, M=CMR_Davis$M, R=CMR_Davis$R,
  NObs = nrow(CMR_Davis),
  NYear = n_distinct(CMR_Davis$Year),
  Year_num = CMR_Davis$Year_num,
  Year_num_direct = CD$Year_num_cd,
  C = CD$CountDavis,
  NObs_direct = nrow(CD)
)

mjags7 <- jags.model(file = 'modelCMR_7.txt', data = data.list7, n.chains = 3 )

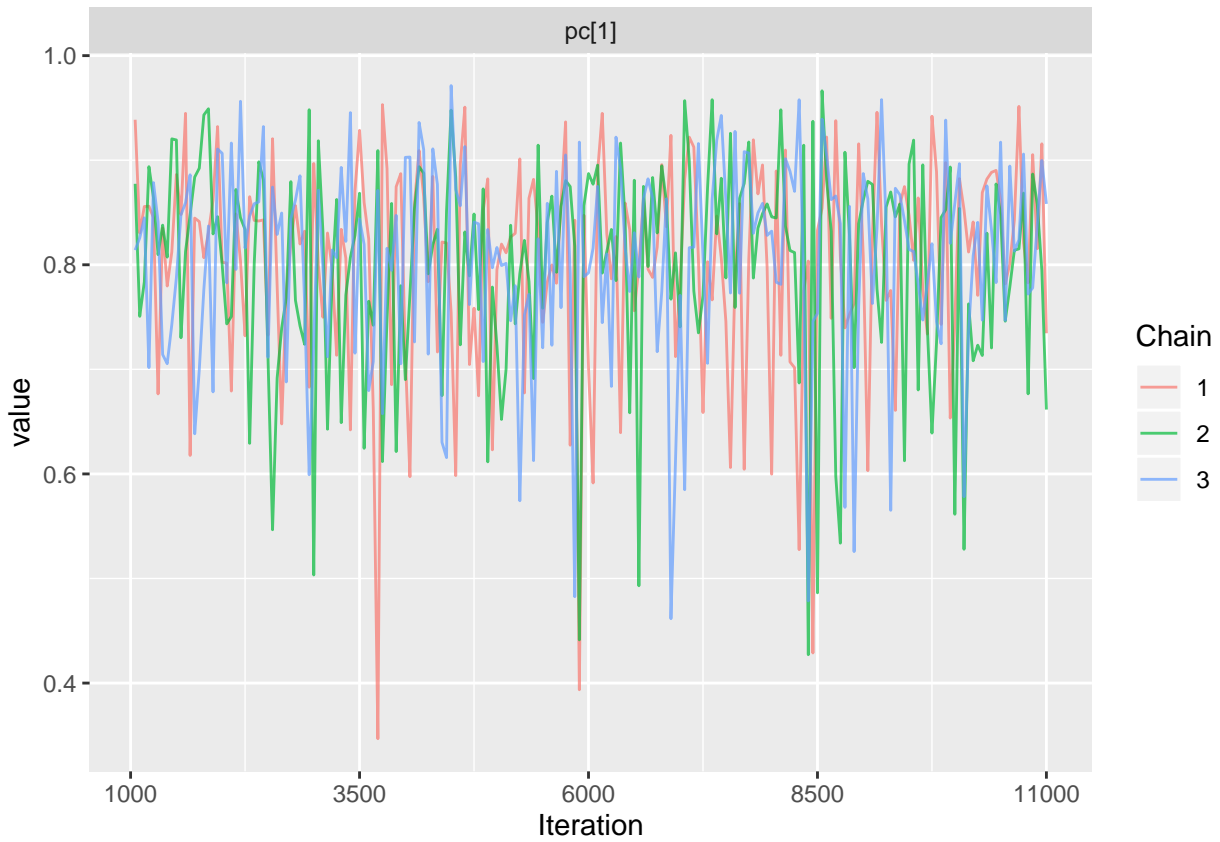
## Compiling model graph
## Resolving undeclared variables
## Allocating nodes
## Graph information:
## Observed stochastic nodes: 479
## Unobserved stochastic nodes: 34
## Total graph size: 1100
##
## Initializing model

```

```
post_sample7 <- coda.samples(mjags7, variable.names = c('p', 'N', 'mu', 'tauA', 'pc' ),
                             n.iter = 10000, thin = 50)
```

```
post7<- ggs(S = post_sample7)
```

```
ggs_traceplot(post7, family = 'pc\\[1\\]')
```



```
post7 %>% filter(stringr::str_detect(string = Parameter, pattern = 'pc\\[1\\]') ) %>% ggplot() + geom_boxp
```


Evolution de la taille de la probabilité d'être vu directement

