Bayesian Statistics week 4

```
library("COUNT")
```

```
## Loading required package: msme
```

```
## Loading required package: MASS
```

```
## Loading required package: lattice
```

```
## Loading required package: sandwich
```

```
data("badhealth")
?badhealth
head(badhealth)
```

```
##   numvisit badh age
## 1       30    0  58
## 2       20    0  54
## 3       16    0  44
## 4       20    0  57
## 5       15    0  33
## 6       15    0  28
```

```
library("rjags")
```

```
## Warning: package 'rjags' was built under R version 4.0.2
```

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

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

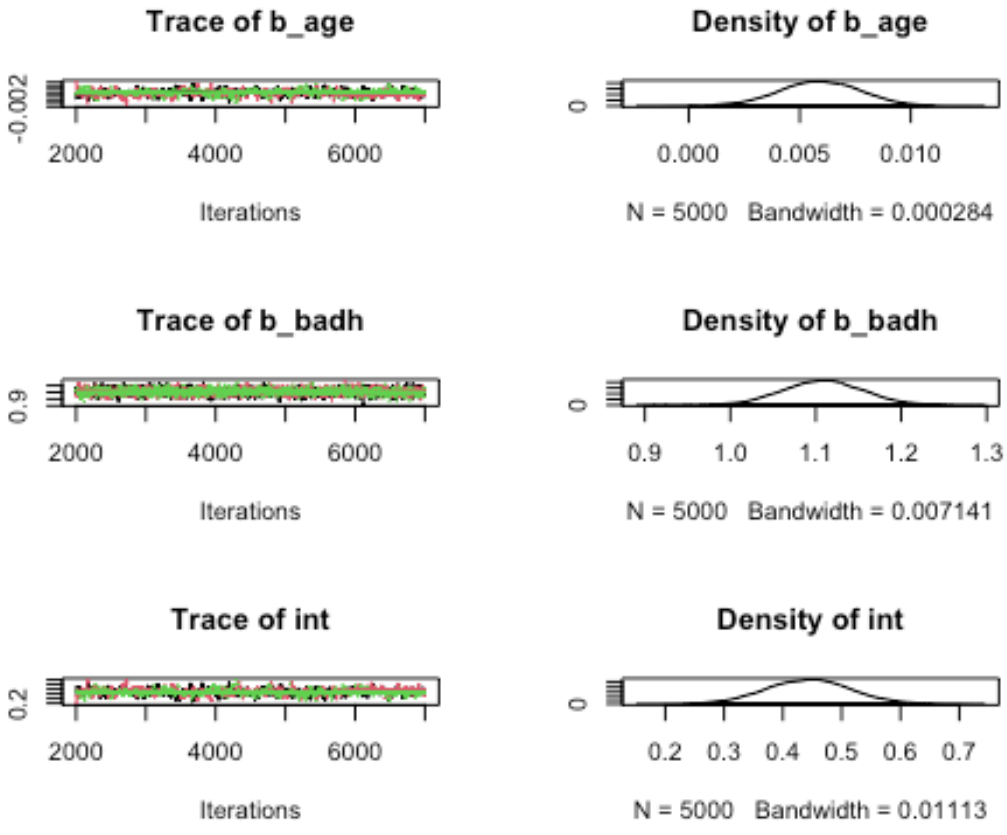
```
## Loaded modules: basemod,bugs
```

```
mod_string = " model {
    for (i in 1:length(numvisit)) {
        numvisit[i] ~ dpois(lam[i])
        log(lam[i]) = int + b_badh*badh[i] + b_age*age[i]
    }

    int ~ dnorm(0.0, 1.0/1e6)
    b_badh ~ dnorm(0.0, 1.0/1e4)
    b_age ~ dnorm(0.0, 1.0/1e4)
} "

set.seed(102)

data_jags = as.list(badhealth)

params = c("int", "b_badh", "b_age")
```

```
mod = jags.model(textConnection(mod_string), data=data_jags, n.chains=3)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 1127
##    Unobserved stochastic nodes: 3
##    Total graph size: 3587
##
## Initializing model
```

```
update(mod, 1e3)

mod_sim = coda.samples(model=mod,
                       variable.names=params,
                       n.iter=5e3)
mod_csim = as.mcmc(do.call(rbind, mod_sim))

## convergence diagnostics
plot(mod_sim)
```



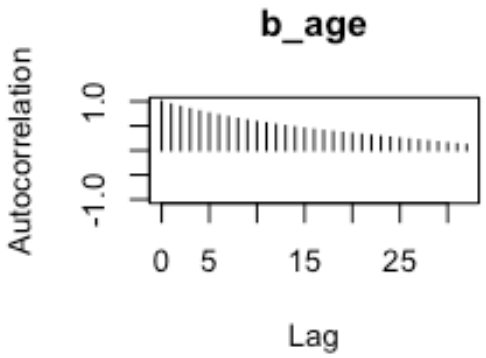
```r
gelman.diag(mod_sim)
```

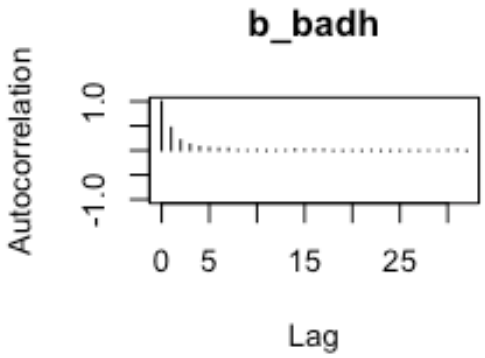
```
## Potential scale reduction factors:
##
##        Point est. Upper C.I.
## b_age        1.01       1.02
## b_badh       1.00       1.01
## int          1.01       1.02
##
## Multivariate psrf
##
## 1.01
```

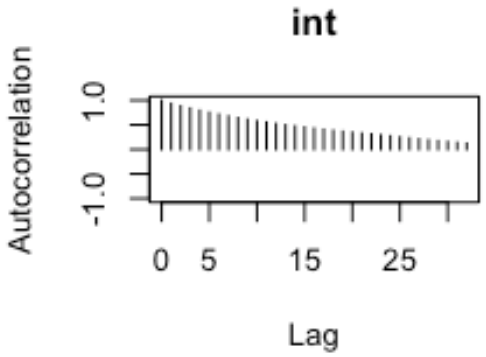
```r
autocorr.diag(mod_sim)
```

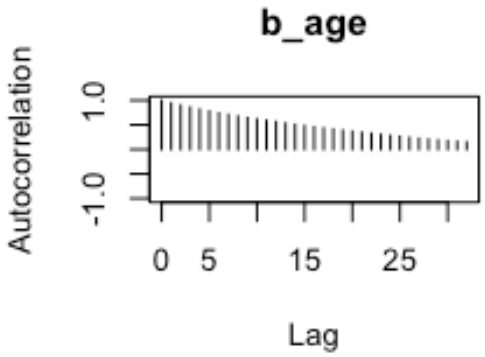
```
##             b_age      b_badh        int
## Lag 0  1.00000000 1.000000000 1.00000000
## Lag 1  0.94642378 0.474283632 0.94194252
## Lag 5  0.78216449 0.067730795 0.77663953
## Lag 10 0.62445905 0.037836302 0.62202174
## Lag 50 0.07126978 0.004483109 0.06460705
```

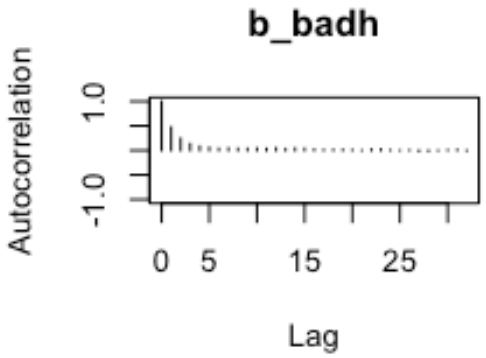
```r
autocorr.plot(mod_sim)
```

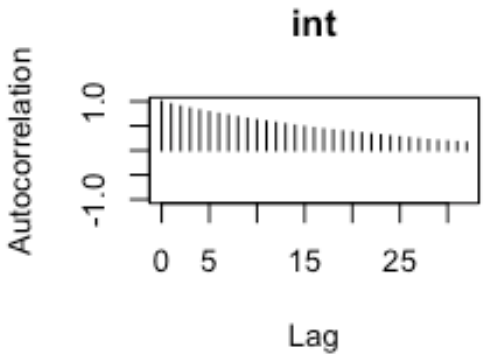

b_age
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

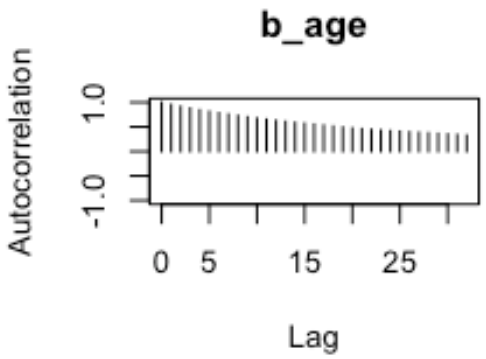

b_badh
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

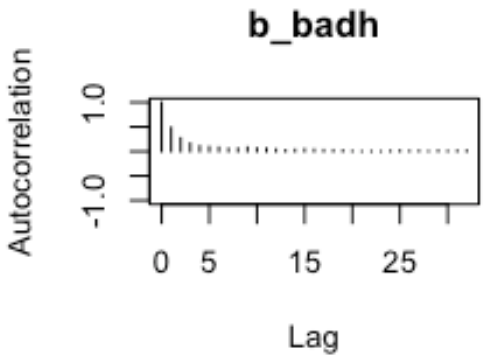

int
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

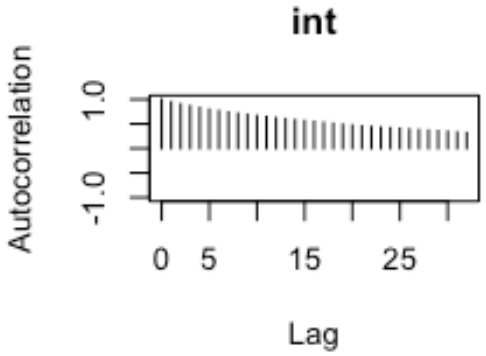

b_age
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b_badh
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


int
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b_age
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b_badh
Autocorrelation
-1.0
1.0
0
5
15
25
Lag




```
effectiveSize(mod_sim)
```

```
##     b_age    b_badh       int
##  357.4101 4589.2172  369.2683
```

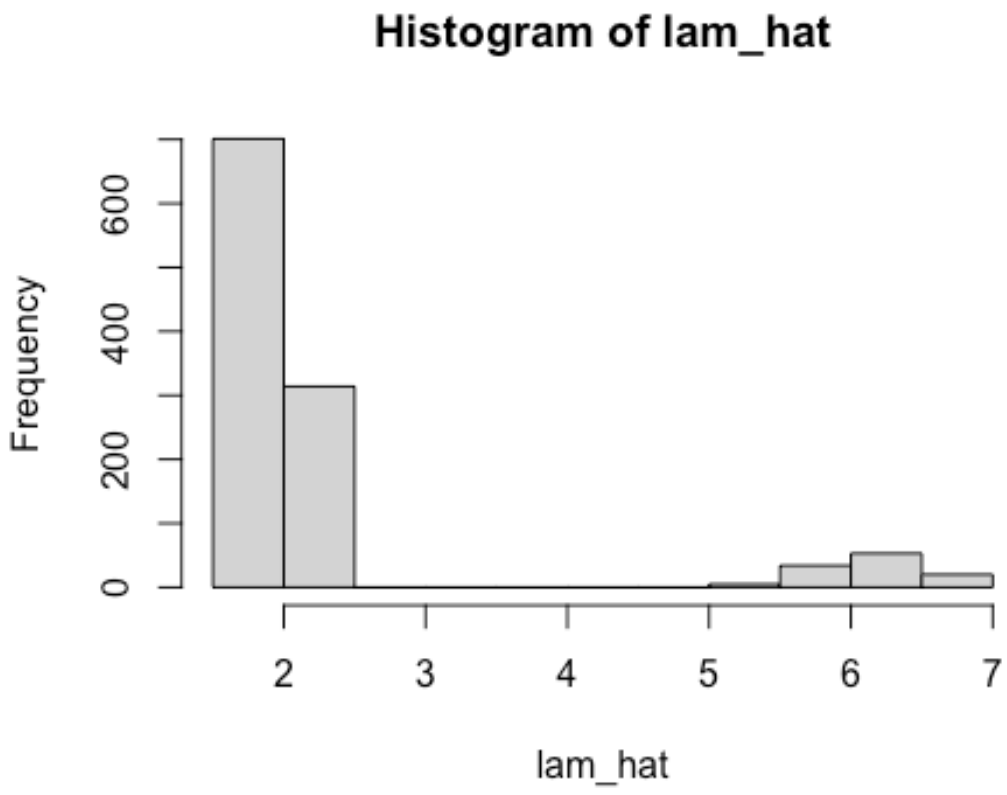
```
## compute DIC
dic = dic.samples(mod, n.iter=1e3)

X = as.matrix(badhealth[,-1])
head(X)
```

```
##      badh age
## [1,]    0  58
## [2,]    0  54
## [3,]    0  44
## [4,]    0  57
## [5,]    0  33
## [6,]    0  28
```

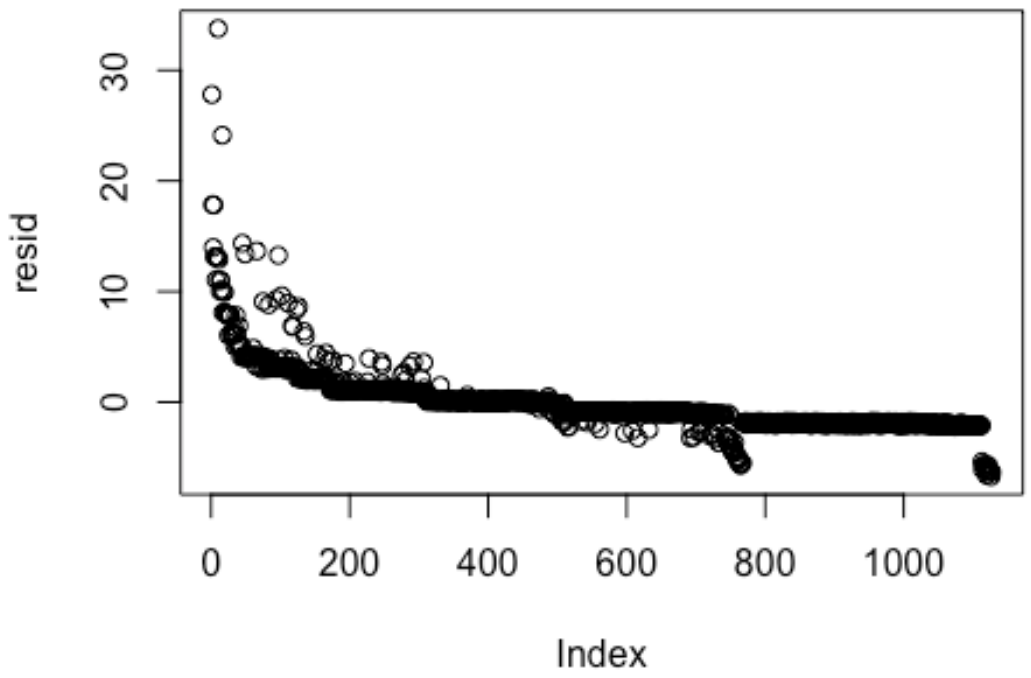
```
(pmed_coef = apply(mod_csim, 2, median))
```

```
##       b_age      b_badh         int
## 0.005900392 1.108531028 0.444110580
```

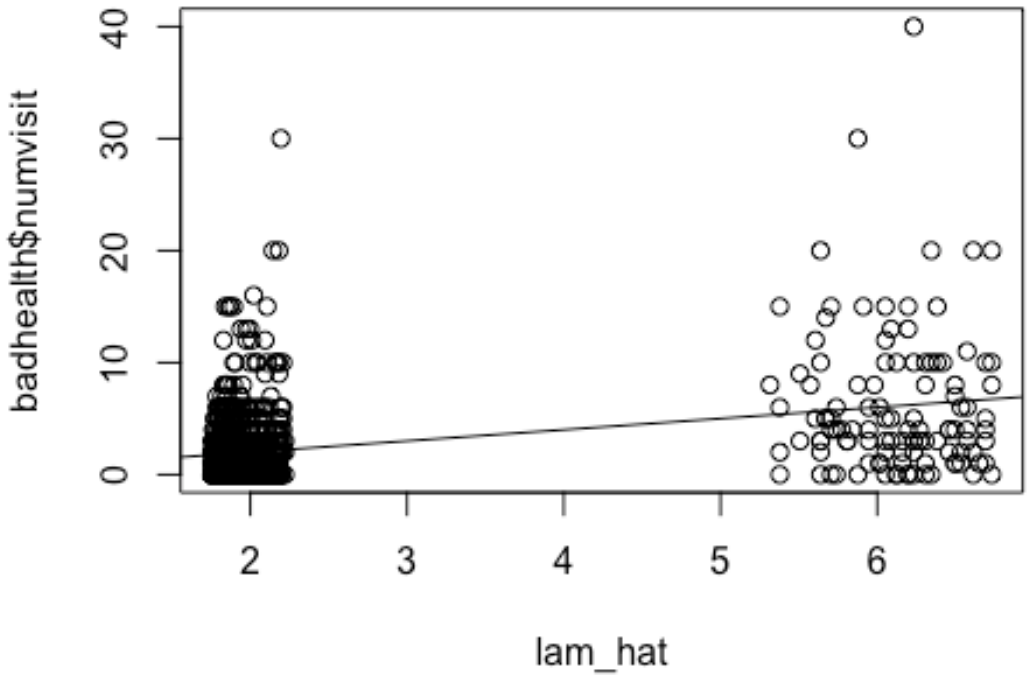
```r
llam_hat = pmed_coef["int"] + X %*% pmed_coef[c("b_badh", "b_age")]
lam_hat = exp(llam_hat)

hist(lam_hat)
```



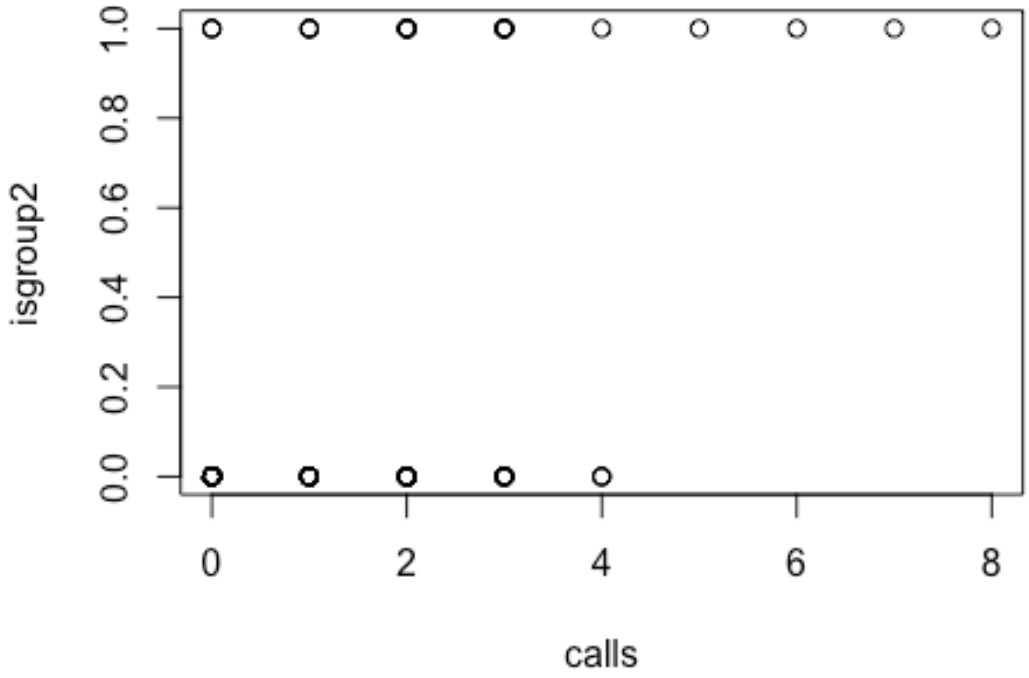

```r
resid = badhealth$numvisit - lam_hat
plot(resid) # the data were ordered
```



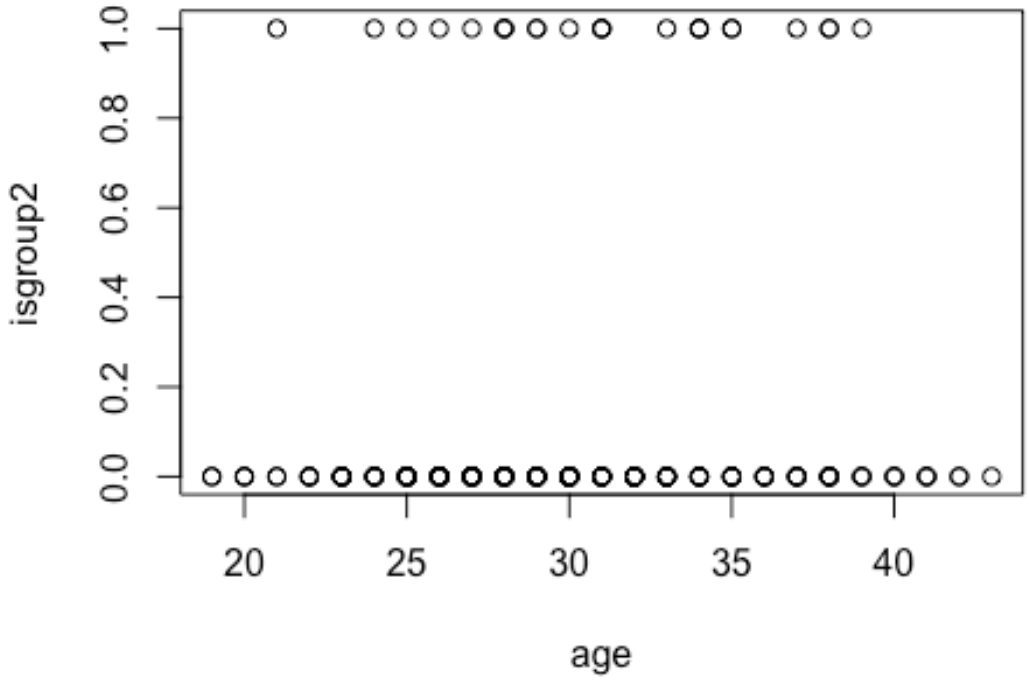

```
plot(lam_hat, badhealth$numvisit)
abline(0.0, 1.0)
```



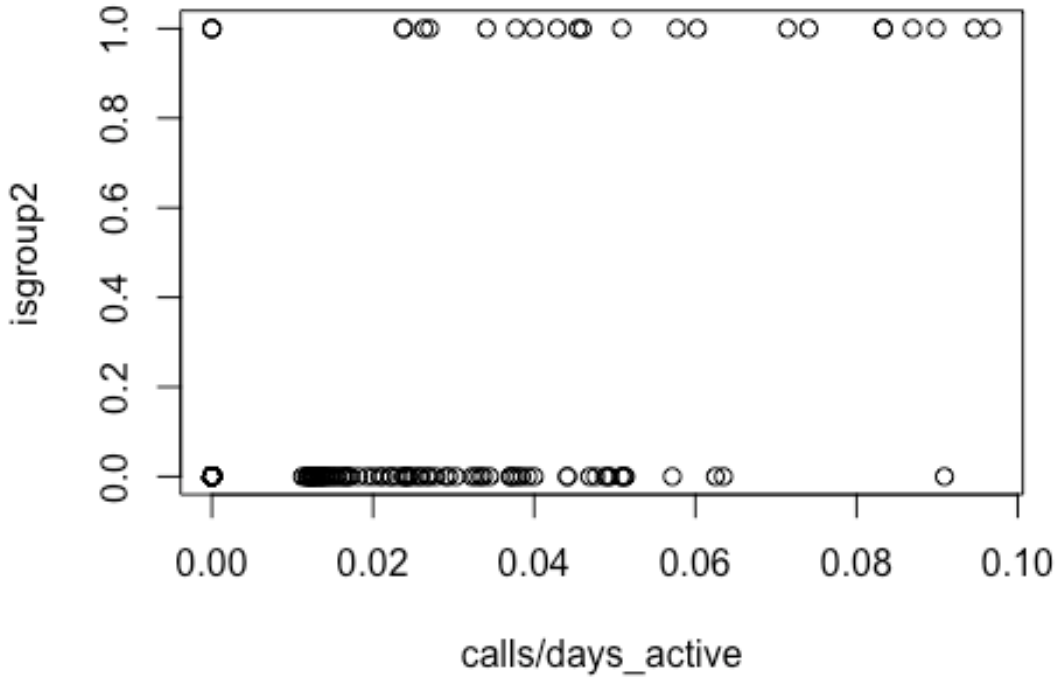

```
dat = read.csv(file="data.csv", header=TRUE)

attach(dat)
head(dat)

##   calls days_active isgroup2 age
## 1     2          32        0  27
## 2     4          81        0  32
## 3     0          41        0  22
## 4     1          36        0  28
## 5     0          55        0  31
## 6     0          25        0  33

plot(calls,isgroup2)
```



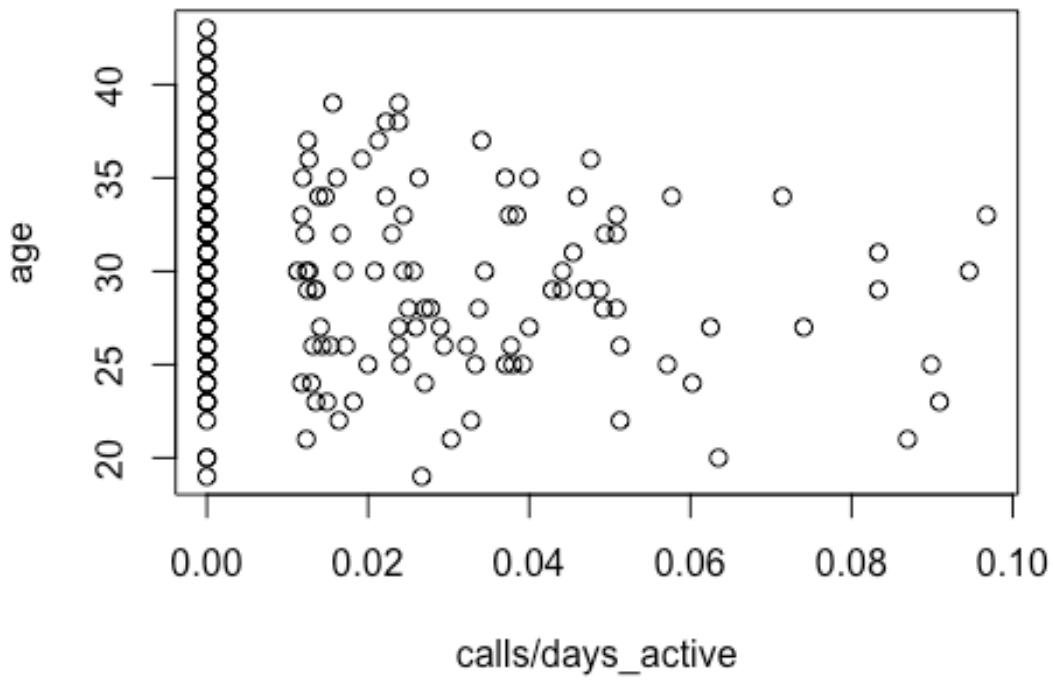

```
plot(age,isgroup2)
```




```
plot(calls/days_active,isgroup2)
```




```
plot(calls/days_active,age)
```



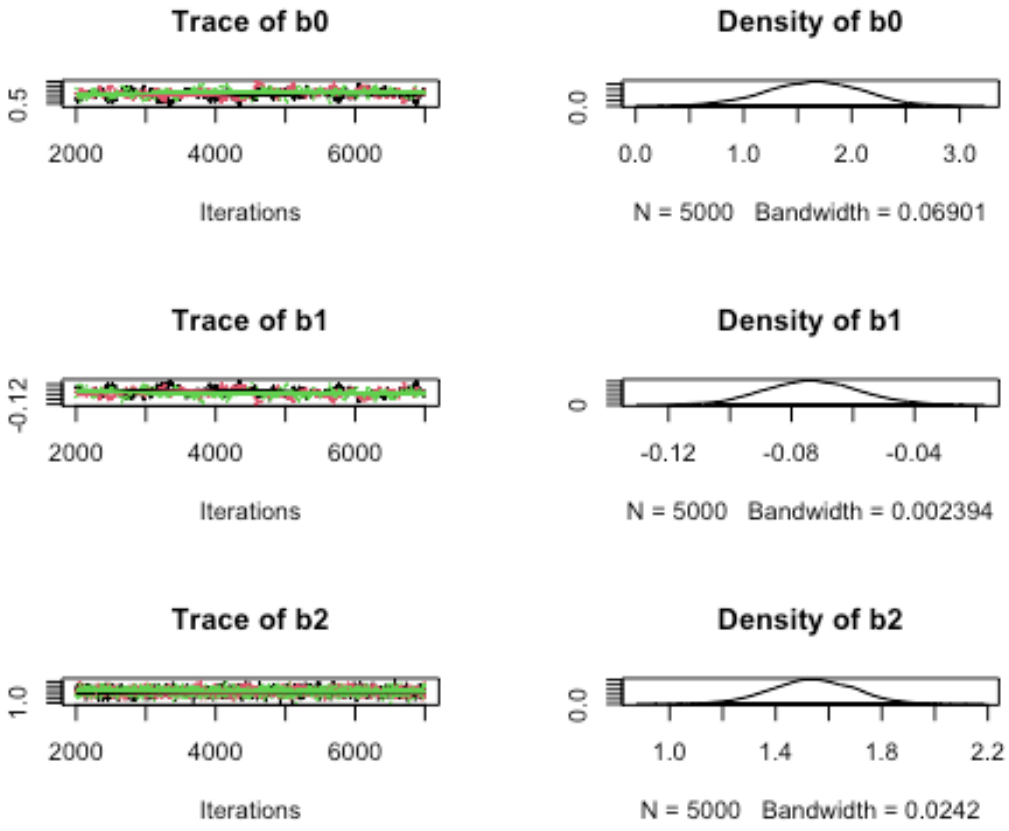

```
mod_string_callers = " model {
    for (i in 1:length(calls)) {
        calls[i] ~ dpois( lam[i] )
        log(lam[i]) = b0 + b1*age[i] + b2*isgroup2[i]
    }

    b0 ~ dnorm(0.0, 1.0/1e2)
    b1 ~ dnorm(0.0, 1.0/1e2)
    b2 ~ dnorm(0.0, 1.0/1e2)
} "

set.seed(102)

data_jags_callers = as.list(dat)

params = c("int", "b0", "b1", "b2")

mod_c = jags.model(textConnection(mod_string_callers), data=data_jags_callers,
 n.chains=3)

## Warning in jags.model(textConnection(mod_string_callers), data =
## data_jags_callers, : Unused variable "days_active" in data
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 224
##    Unobserved stochastic nodes: 3
##    Total graph size: 786
##
## Initializing model
```

```
update(mod_c, 1e3)

mod_sim_c = coda.samples(model=mod_c,
                         variable.names=params,
                         n.iter=5e3)
```

```
## Warning in FUN(X[[i]], ...): Failed to set trace monitor for int
## Variable int not found
```

```
mod_csim_c = as.mcmc(do.call(rbind, mod_sim_c))

## convergence diagnostics
plot(mod_sim_c)
```



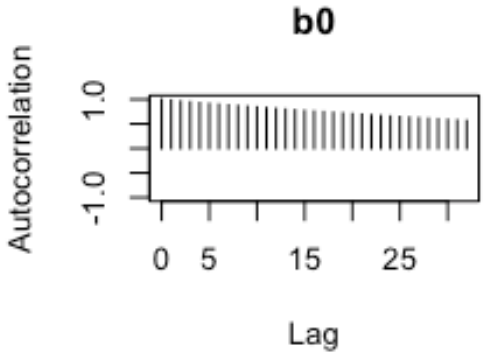
```r
gelman.diag(mod_sim_c)
```

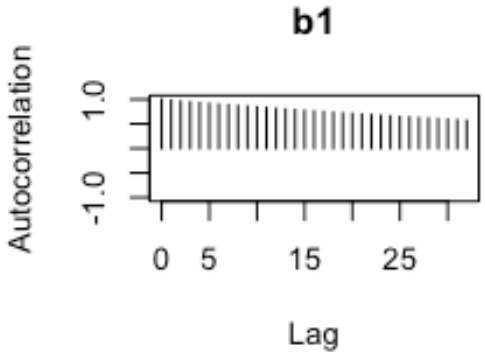
```
## Potential scale reduction factors:
##
##    Point est. Upper C.I.
## b0       1.03       1.12
## b1       1.04       1.12
## b2       1.00       1.00
##
## Multivariate psrf
##
## 1.03
```

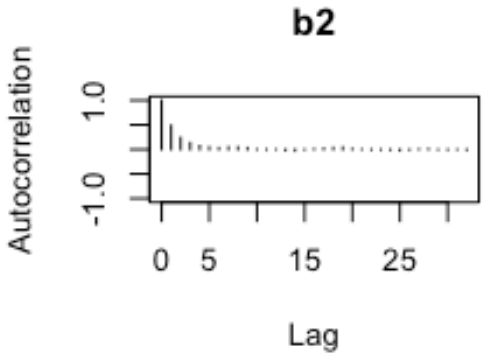
```r
autocorr.diag(mod_sim_c)
```

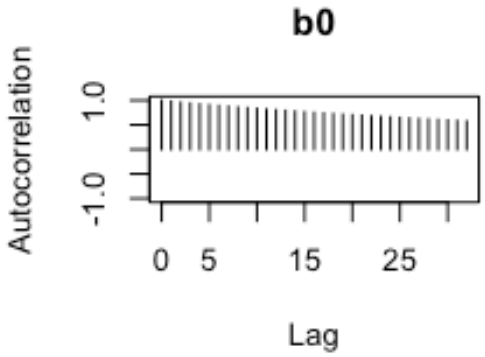
```
##                b0        b1          b2
## Lag 0   1.0000000 1.0000000 1.000000000
## Lag 1   0.9784843 0.9790483 0.499051191
## Lag 5   0.9086386 0.9096017 0.041799320
## Lag 10  0.8295925 0.8315188 0.009287233
## Lag 50  0.4092002 0.4152839 0.001305422
```

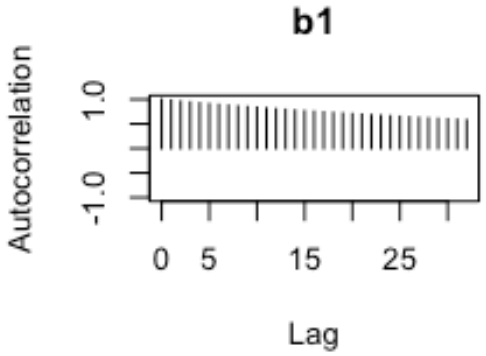
```r
autocorr.plot(mod_sim_c)
```

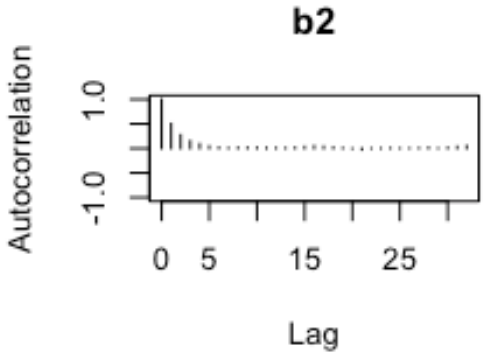

b0
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

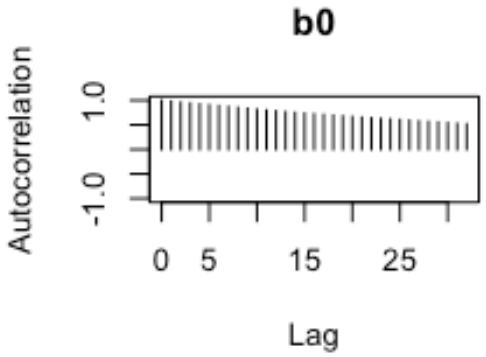

b1
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

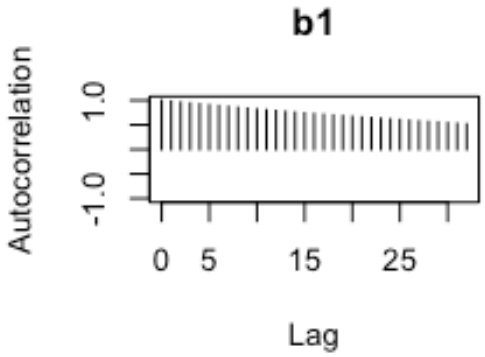

b2
Autocorrelation
-1.0
1.0
0
5
15
25
Lag

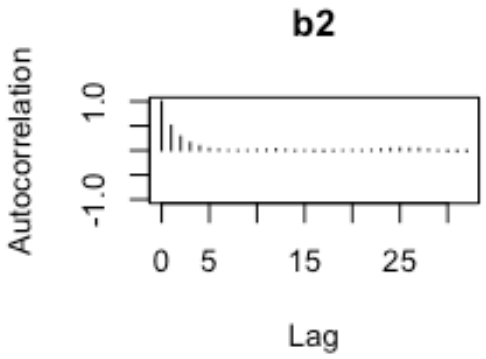

b0
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b1
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b2
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b0
Autocorrelation
-1.0
1.0
0
5
15
25
Lag


b1
Autocorrelation
-1.0
1.0
0
5
15
25
Lag



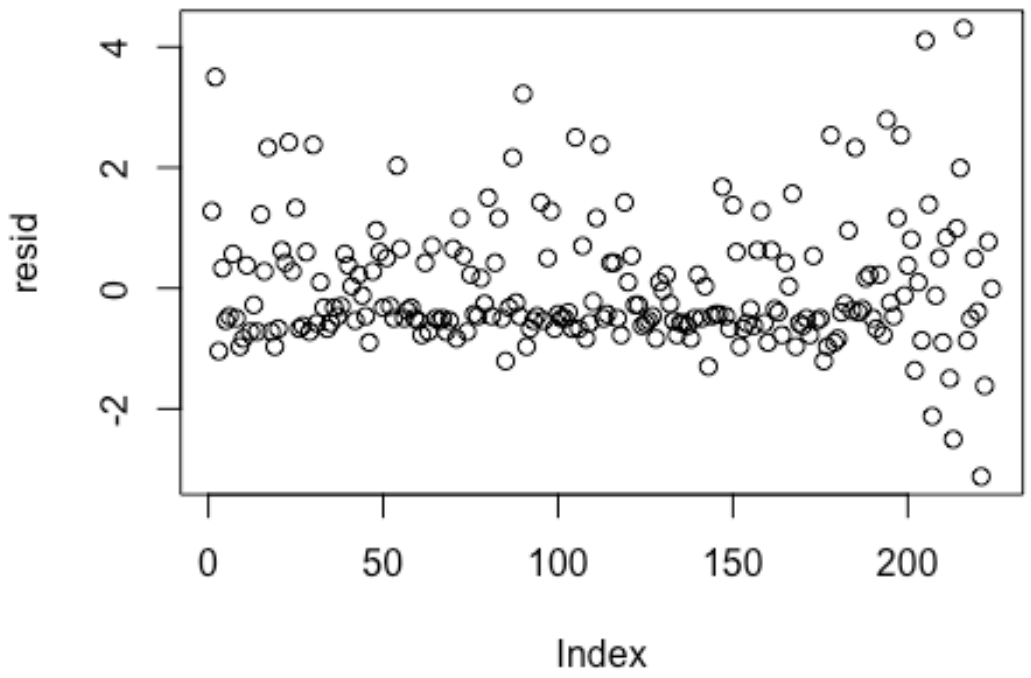

```
effectiveSize(mod_sim_c)
```

```
##        b0        b1        b2
##  141.0077  139.0833 4812.7578
```

```
## compute DIC

dic_c = dic.samples(mod_c, n.iter=1e3)

X = as.matrix(dat[,c(4,3)])
(pmed_coef = apply(mod_csim_c, 2, median))
```

```
##          b0          b1          b2
##  1.65976285 -0.07359971  1.53960069
```

```
llam_hat = pmed_coef["b0"] + X %*% pmed_coef[c("b1", "b2")]
lam_hat = exp(llam_hat)
resid = dat$calls - lam_hat
plot(resid)
```




```
mean(mod_csim_c[,"b2"]>0)
```

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