Metropolis sampling for the concussions data with adaptive tuning

Chapter 3.2.3: Metropolis sampling

Let Y_i be the number of concussions (aggregated over all teams and games) in season i (1=2012,…,4=2015). We model these counts as

$$Y_i \sim \text{Poisson}(N\lambda_i) \text{ where } \lambda_i = \exp(\beta_1 + i\beta_2),$$

N is the number of games played per year and λ_i is the rate in year i. To complete the Bayesian model, we specify uninformative priors $\beta_1, \beta_2 \sim \text{Normal}(0, \tau^2)$.

The log of the mean concussion rate is linear in time with β_2 determining the slope. The objective is to determine if the concussion rate is increasing, i.e., $\beta_2 > 0$.

In this analysis of these data with this algorithm, we tune the Metropolis candidate distributions during the burn-in period. The objective to tune the sampler so the acceptance rate is around 0.4 for all parameters. We check the acceptance proportion in the last 100 iterations and adjust the candidate SD if the acceptance probability is far from 0.4.

Load the data

```
Y <- c(171, 152, 123, 199)
t <- 1:4
n <- 4
N <- 256
```

Initialize

```
# Create an empty matrix for the MCMC samples

 S                <- 25000
 samples          <- matrix(NA,S,2)
 colnames(samples) <- c("beta1","beta2")
 fitted           <- matrix(0,S,4)

# Initial values

 beta   <- c(log(mean(Y/N)),0)

# priors: beta[j] ~ N(0,tau^2)

 tau    <- 10

# Initial candidate standard deviations

 can_sd <- c(1,1)
```

Define the log posterior as a function

```
log_post <- function(Y,N,t,beta,tau){
   mn    <- N*exp(beta[1]+beta[2]*t)
   like  <- sum(dpois(Y,mn,log=TRUE))
   prior <- sum(dnorm(beta,0,tau,log=TRUE))
   post  <- like + prior
return(post)}
```

Metropolis sampling

```r
burn  <- 5000      # Length of burn-in period for tuning
check <- 100       # Iterations between checks of the acceptance rate
att   <- rep(0,2)  # Keep track of the number of MH attempts
acc   <- rep(0,2)  # Keep track of the number of MH accepts

for(s in 1:S){
  for(j in 1:2){
    att[j] <- att[j] + 1
    can    <- beta
    can[j] <- rnorm(1,beta[j],can_sd[j])
    logR   <- log_post(Y,N,t,can,tau)-log_post(Y,N,t,beta,tau)
    if(log(runif(1))<logR){
      beta   <- can
      acc[j] <- acc[j] + 1
    }
  }

  # TUNING!
  for(j in 1:length(att)){
     if(s<burn & att[j]==check){
       print(paste0("Can sd of ", round(can_sd[j],3),
                    " for beta[",j,"] gave acc rate ",acc[j]/att[j]))
       if(acc[j]/att[j]<0.2){can_sd[j]<-can_sd[j]*0.8}
       if(acc[j]/att[j]>0.6){can_sd[j]<-can_sd[j]*1.2}
       acc[j] <- att[j] <- 0
     }
  }

  samples[s,] <- beta
  fitted[s,]  <- N*exp(beta[1]+beta[2]*t)
}
```

```
## [1] "Can sd of 1 for beta[1] gave acc rate 0.1"
## [1] "Can sd of 1 for beta[2] gave acc rate 0"
## [1] "Can sd of 0.8 for beta[1] gave acc rate 0.1"
## [1] "Can sd of 0.8 for beta[2] gave acc rate 0.03"
## [1] "Can sd of 0.64 for beta[1] gave acc rate 0.05"
## [1] "Can sd of 0.64 for beta[2] gave acc rate 0.07"
## [1] "Can sd of 0.512 for beta[1] gave acc rate 0.06"
## [1] "Can sd of 0.512 for beta[2] gave acc rate 0"
## [1] "Can sd of 0.41 for beta[1] gave acc rate 0.11"
## [1] "Can sd of 0.41 for beta[2] gave acc rate 0.03"
## [1] "Can sd of 0.328 for beta[1] gave acc rate 0.18"
## [1] "Can sd of 0.328 for beta[2] gave acc rate 0.04"
## [1] "Can sd of 0.262 for beta[1] gave acc rate 0.18"
## [1] "Can sd of 0.262 for beta[2] gave acc rate 0.04"
## [1] "Can sd of 0.21 for beta[1] gave acc rate 0.21"
## [1] "Can sd of 0.21 for beta[2] gave acc rate 0.05"
## [1] "Can sd of 0.21 for beta[1] gave acc rate 0.23"
## [1] "Can sd of 0.168 for beta[2] gave acc rate 0.1"
## [1] "Can sd of 0.21 for beta[1] gave acc rate 0.24"
## [1] "Can sd of 0.134 for beta[2] gave acc rate 0.14"
## [1] "Can sd of 0.21 for beta[1] gave acc rate 0.18"
## [1] "Can sd of 0.107 for beta[2] gave acc rate 0.15"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.28"
## [1] "Can sd of 0.086 for beta[2] gave acc rate 0.22"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.23"
## [1] "Can sd of 0.086 for beta[2] gave acc rate 0.16"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.31"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.34"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.29"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.28"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.2"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.3"
```

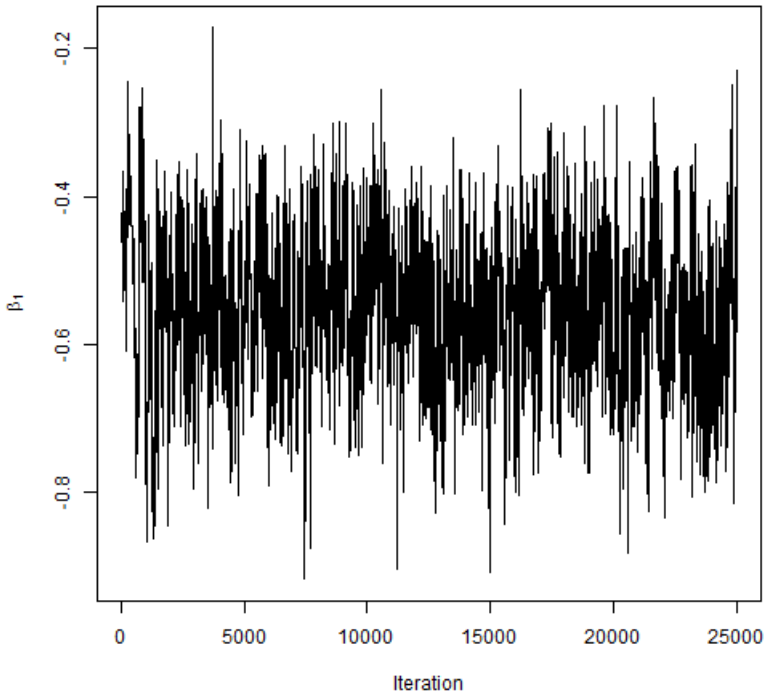
```
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.28"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.28"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.25"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.168 for beta[1] gave acc rate 0.17"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.22"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.36"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.28"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.24"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.41"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.21"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.32"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.25"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.31"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.25"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.25"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.36"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.21"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.27"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.2"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.39"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.26"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.28"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.26"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.35"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.41"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.28"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.37"
## [1] "Can sd of 0.069 for beta[2] gave acc rate 0.18"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.4"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.31"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.36"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.35"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.32"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.35"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.32"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.36"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.3"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.35"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.3"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.35"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.3"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.38"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.33"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.38"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.3"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.27"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.37"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.37"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.4"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.37"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.29"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.3"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.34"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.36"
## [1] "Can sd of 0.134 for beta[1] gave acc rate 0.34"
## [1] "Can sd of 0.055 for beta[2] gave acc rate 0.38"
```

Compute the acceptance rates and plot the samples

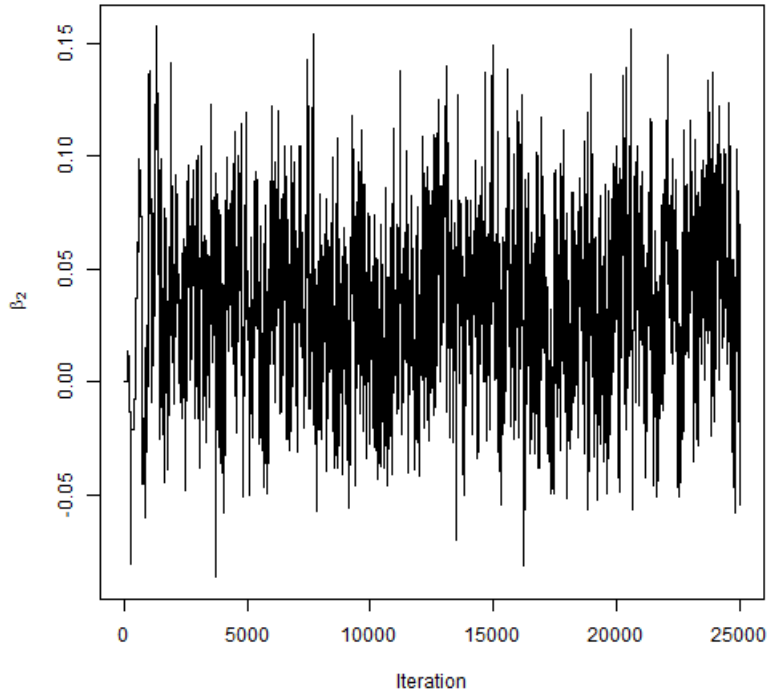
```
# Acceptance rates
colMeans(samples[burn:S,]!=samples[burn:S - 1,])
```

```
##     beta1     beta2
## 0.3355832 0.3066847
```

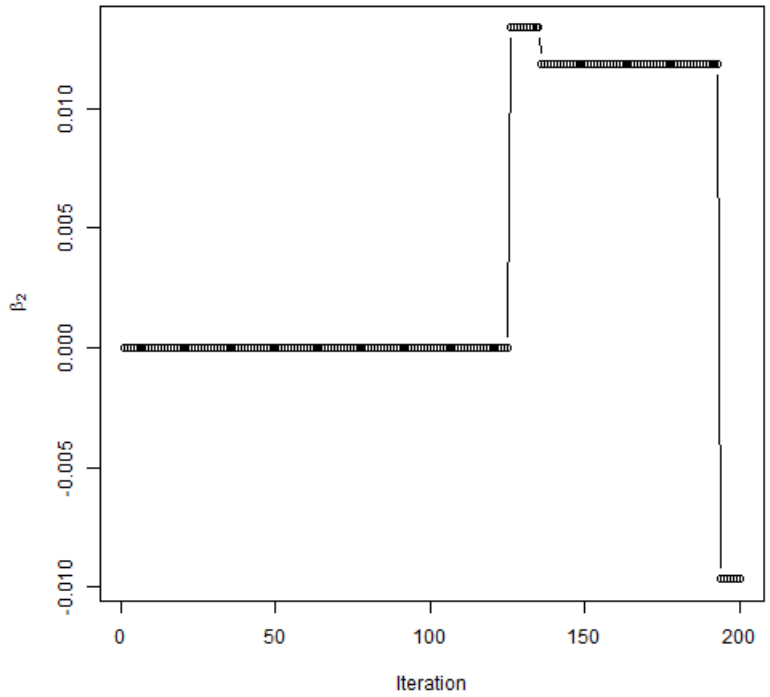
```
plot(samples[,1],type="l",xlab="Iteration",ylab=expression(beta[1]))
```



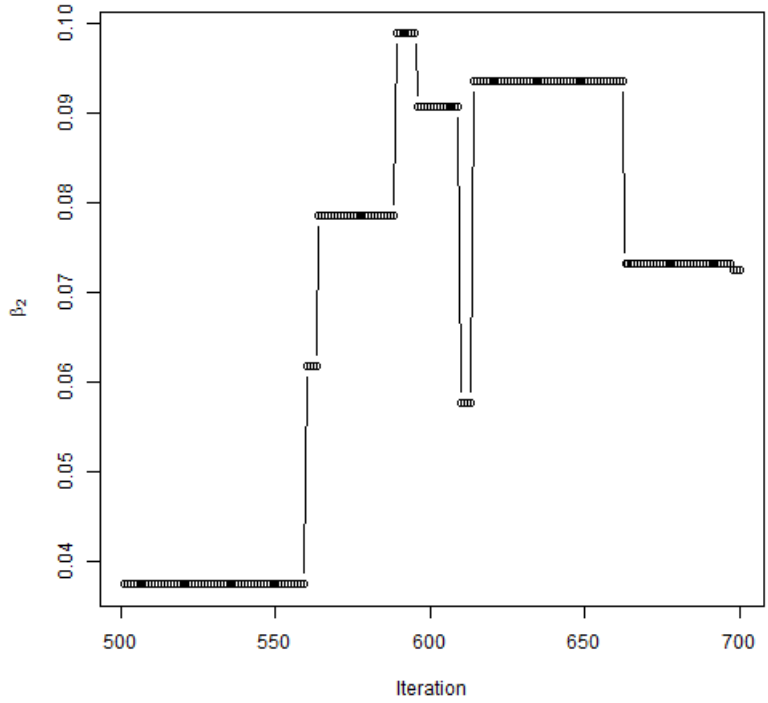

```
plot(samples[,2],type="l",xlab="Iteration",ylab=expression(beta[2]))
```



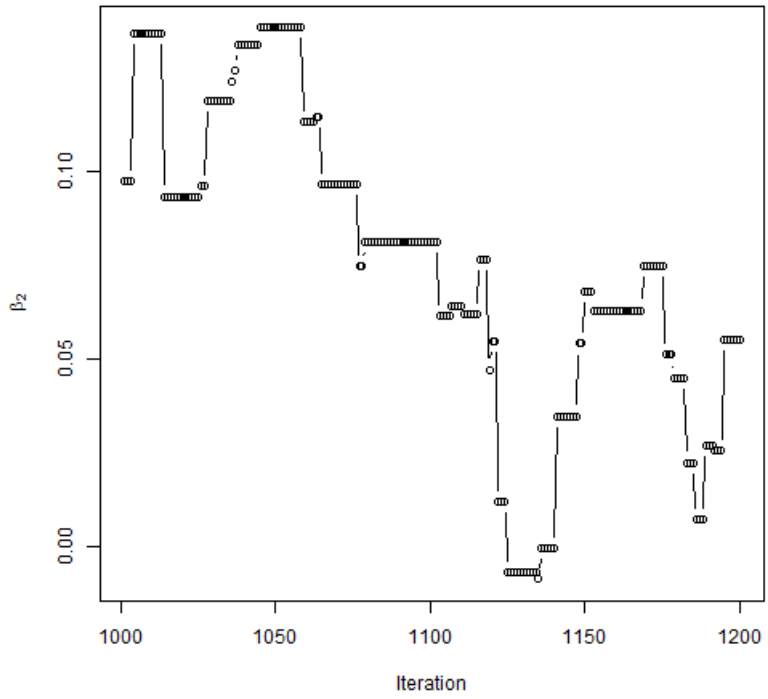

```
plot(1:200,samples[1:200,2],type="b",xlab="Iteration",ylab=expression(beta[2]))
```



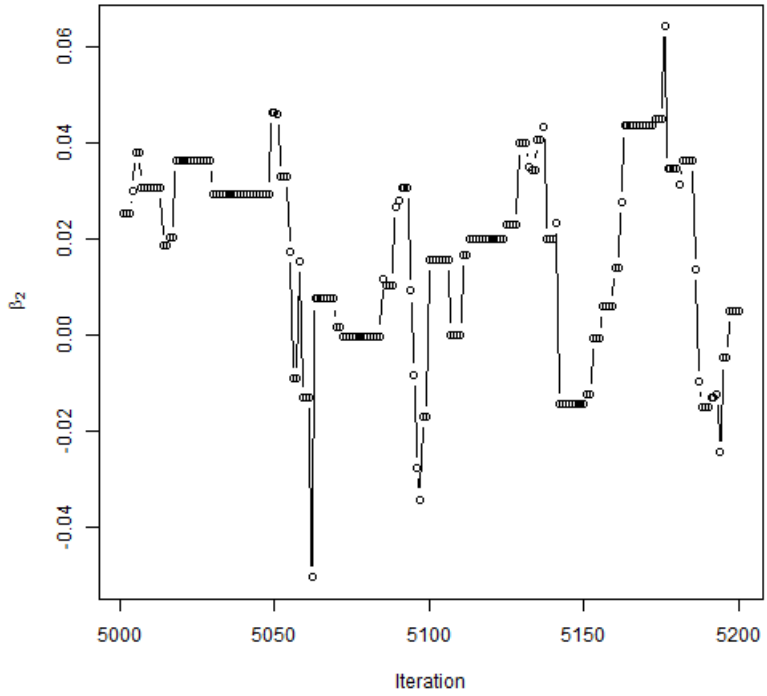

```
plot(1:200+500,samples[1:200+500,2],type="b",xlab="Iteration",ylab=expression(beta[2]))
```




```
plot(1:200+1000,samples[1:200+1000,2],type="b",xlab="Iteration",ylab=expression(beta[2]))
```




```
plot(1:200+5000,samples[1:200+5000,2],type="b",xlab="Iteration",ylab=expression(beta[2]))
```



Mixing improves as the candidate sd is tuned.