Random effects model

Suppose that

$$Y_{ij} = \theta_i + \epsilon_{ij}$$

for $i = 1, \ldots, M, j = 1, \ldots, n_i$, where

$$\epsilon_{ij} \sim Normal(0, \sigma^2)$$

are independent residual errors. The unknown parameters are

$$\theta_1, \theta_2, \ldots, \theta_M, \sigma^2$$

where the θ_i are *subgroup means*, and where

$$Y_{ij} \sim Normal(\theta_i, \sigma^2)$$

are conditionally independent.

Random effects model

This model is known as a *one-way layout*

- a form of Normal linear model
- it is the basis of one-way ANOVA, where a classical hypothesis test of the null hypothesis

$$H_0 : \theta_1 = \theta_2 = \cdots = \theta_M$$

  is carried out.
- for a Bayesian analysis, we may use the previously derived Normal linear theory; we might assume that, given σ^2

$$\theta = (\theta_1, \theta_2, \ldots, \theta_M)^\top \sim Normal_M(\mathbf{m}_0, \mathbf{M}_0)$$

  and so on.

Random effects model

Suppose instead that for $i = 1, \ldots, M$

$$\theta_i \sim Normal(\phi, \tau^2)$$

are independent. We have a *hierarchical model*, and might assume that

- $\phi|\tau^2 \sim Normal(0, \tau^2/\lambda_0)$,
- $\tau^2 \sim InverseGamma(g_0, h_0)$

to complete the model. The unknown parameters are now

$$\theta_1, \theta_2, \ldots, \theta_M, \sigma^2, \phi, \tau^2$$

and we are treating the θ_i as *random effects*.

Random effects model

The values

$$(\theta_1, \theta_2, \ldots, \theta_M)$$

are assumed to be drawn from some second stage *population*.

This encourages the θ_i to 'shrink' towards each other.

Random effects model

If $\theta = (\theta_1, \theta_2, \ldots, \theta_M)$, the posterior is

$$\pi_n(\theta, \sigma^2, \phi, \tau^2) \propto \mathcal{L}_n(\theta, \sigma^2)\pi_0(\theta|\phi, \tau^2)\pi_0(\sigma^2)\pi_0(\phi|\tau^2)\pi_0(\tau^2)$$

and is not analytically tractable.

We can instead use MCMC via the Gibbs sampler.

Random effects model

We need the *full conditional distributions*

- θ: we have by conditional independence

$$\pi_n(\theta|\sigma^2, \phi, \tau^2) \equiv \prod_{i=1}^{M} \pi_{n_i}(\theta_i|\sigma^2, \phi, \tau^2)$$

where

$$\pi_{n_i}(\theta_i|\sigma^2, \phi, \tau^2) \propto \mathcal{L}_{n_i}(\theta_i, \sigma^2)\pi_0(\theta_i|\sigma^2, \phi, \tau^2).$$

Each of the M full conditionals is univariate Normal.

Random effects model

- σ^2: we have

$$\pi_n(\sigma^2|\theta, \phi, \tau^2) \propto \mathcal{L}_n(\theta, \sigma^2)\pi_0(\sigma^2)$$

  which is an Inverse Gamma distribution.

  Note that the θ_i are treated as *known* in this calculation.

Random effects model

- ϕ: we have

$$\pi_n(\phi|\theta,\sigma^2,\tau^2) \propto \prod_{i=1}^{M} \pi_0(\theta_i|\phi,\tau^2)\pi_0(\phi|\tau^2)$$

  which is a Normal distribution.

Random effects model

- ϕ: we have

$$\pi_n(\tau^2|\theta,\sigma^2,\phi) \propto \prod_{i=1}^{M} \pi_0(\theta_i|\phi,\tau^2)\pi_0(\phi|\tau^2)\pi_0(\tau^2)$$

  which is an Inverse Gamma distribution.

Random effects model

Note that we may *reparameterize* the model

$$Y_{ij} = \phi + \delta_i + \epsilon_{ij}$$

where

$$\theta_i = \phi + \delta_i$$

so that

$$\pi_0(\delta_i|\phi, \tau^2) \equiv \pi_0(\delta_i|\tau^2) = Normal(0, \tau^2)$$

are conditionally independent and identically distributed.

Random effects model

Which is the better parameterization for MCMC?

This depends on several factors:

- sample sizes
- prior specification
- relative magnitudes of the *true values* of the parameters, especially

$$\sigma^2 \quad \text{and} \quad \tau^2.$$

e.g.

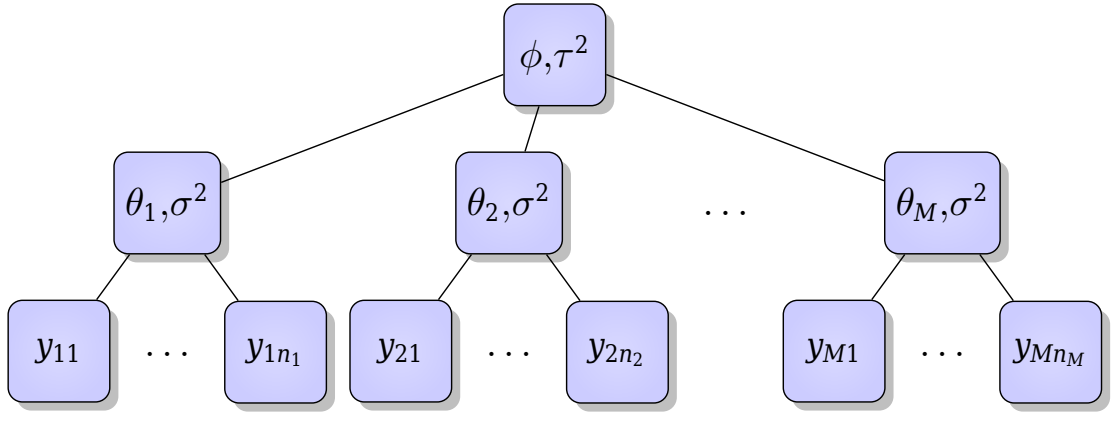
- τ^2 'large', infer θ_i directly as second stage model is not very informative
- τ^2 'small', use δ_i.

Random effects model

1 $\pi_0(\phi,\tau^2)$

2 $\prod_{i=1}^{M} \pi_0(\theta_i|\phi,\tau^2)\pi_0(\sigma^2)$

3 $f(y_{ij};\theta_i,\sigma^2)$



Random effects model

- Stage 3:

$$f(y_{ij};\theta_i,\sigma^2)=\left(\frac{1}{2\pi\sigma^2}\right)^{1/2}\exp\left\{-\frac{1}{2\sigma^2}(y_{ij}-\theta_i)^2\right\}$$

- Stage 2:

$$\pi_0(\theta_1,\ldots,\theta_M,\sigma^2)=\prod_{i=1}^{M}\left(\frac{1}{2\pi\tau^2}\right)^{1/2}\exp\left\{-\frac{1}{2\tau^2}(\theta_i-\phi)^2\right\}$$
$$\times\frac{(b_0/2)^{a_0/2}}{\Gamma(a_0/2)}\left(\frac{1}{\sigma^2}\right)^{a_0/2+1}\exp\left\{-\frac{b_0}{2\sigma^2}\right\}$$

Random effects model

- Stage 1:

$$\pi_0(\phi, \tau^2) = \left(\frac{\lambda_0}{2\pi\tau^2}\right)^{1/2} \exp\left\{-\frac{\lambda_0}{2\tau^2}\phi^2\right\}$$
$$\times \frac{(h_0/2)^{g_0/2}}{\Gamma(g_0/2)} \left(\frac{1}{\tau^2}\right)^{g_0/2+1} \exp\left\{-\frac{h_0}{2\tau^2}\right\}$$

Hyperparameters: $a_0, b_0, \lambda_0, g_0, h_0$.

Random effects model: simulation

```r
set.seed(253)
M<-10
a0<-b0<-2; g0<-8;h0<-20
lam0<-1

ni<-rpois(M,4)
ni
```

```
##  [1] 4 4 3 2 3 1 3 3 5 7
```

```r
n<-sum(ni)
tau<-sqrt(1/rgamma(1,g0/2,h0/2))
sig<-sqrt(1/rgamma(1,a0/2,b0/2))
phi<-rnorm(1,0,tau/sqrt(lam0))

c(sig^2,phi,tau^2)
```

```
## [1]  0.6470075 -2.9466268  3.3089643
```

```r
th0<-rnorm(M,phi,tau)
th0
```

```
##  [1] -2.880424 -3.824574 -3.417930 -3.500194 -2.086139 -1.701824 -2.242565
##  [8] -3.504905 -2.385957 -1.349649
```

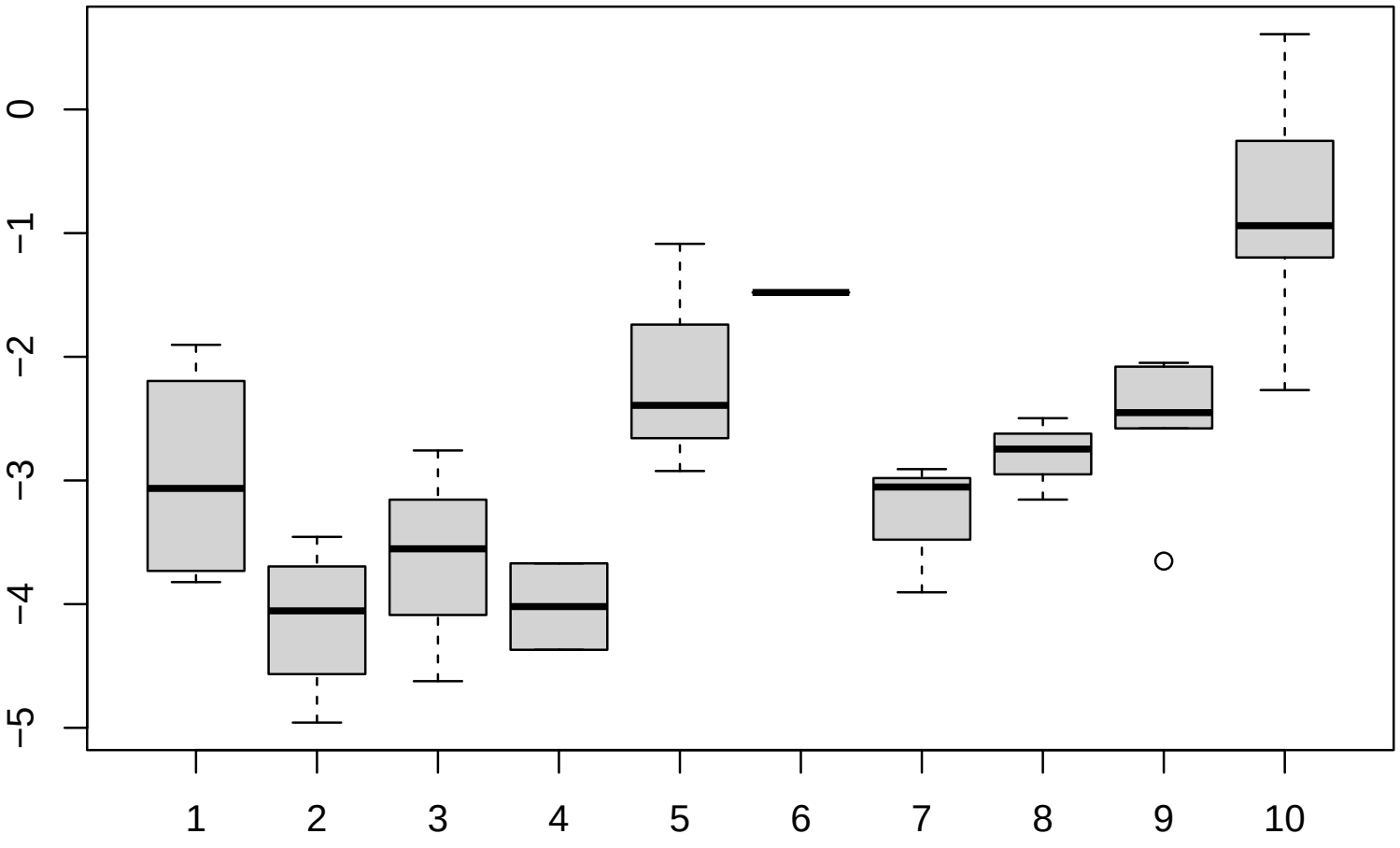
Random effects model: simulation

```
Y<-list()
for(i in 1:M){Y[[i]]<-rnorm(ni[i],th0[i],sig)}
(Ybar<-unlist(lapply(Y,mean)))

##  [1] -2.9633954 -4.1304871 -3.6442969 -4.0196674 -2.1342784 -1.4801636
##  [7] -3.2885510 -2.7991495 -2.5622026 -0.7859892

par(mar=c(4,4,2,2))
boxplot(Y)
```

Random effects model: simulation



Random effects model: simulation

```
#MCMC
nburn<-100
nthin<-10
nsamp<-2000
nits<-nburn+nthin*nsamp

#Initial values
old.th<-Ybar
old.ssq<-rep(0,M)
for(i in 1:M){
    old.ssq[i]<-sum((Y[[i]]-Ybar[i])^2)
}
old.sigsq<-sum(old.ssq)/(n-M)
old.phi<-mean(old.th)
old.tausq<-sum((old.th-old.phi)^2)/(M-1)
old.th
```

```
##  [1] -2.9633954 -4.1304871 -3.6442969 -4.0196674 -2.1342784 -1.4801636
##  [7] -3.2885510 -2.7991495 -2.5622026 -0.7859892
```

```
c(old.sigsq,old.phi,old.tausq)
```

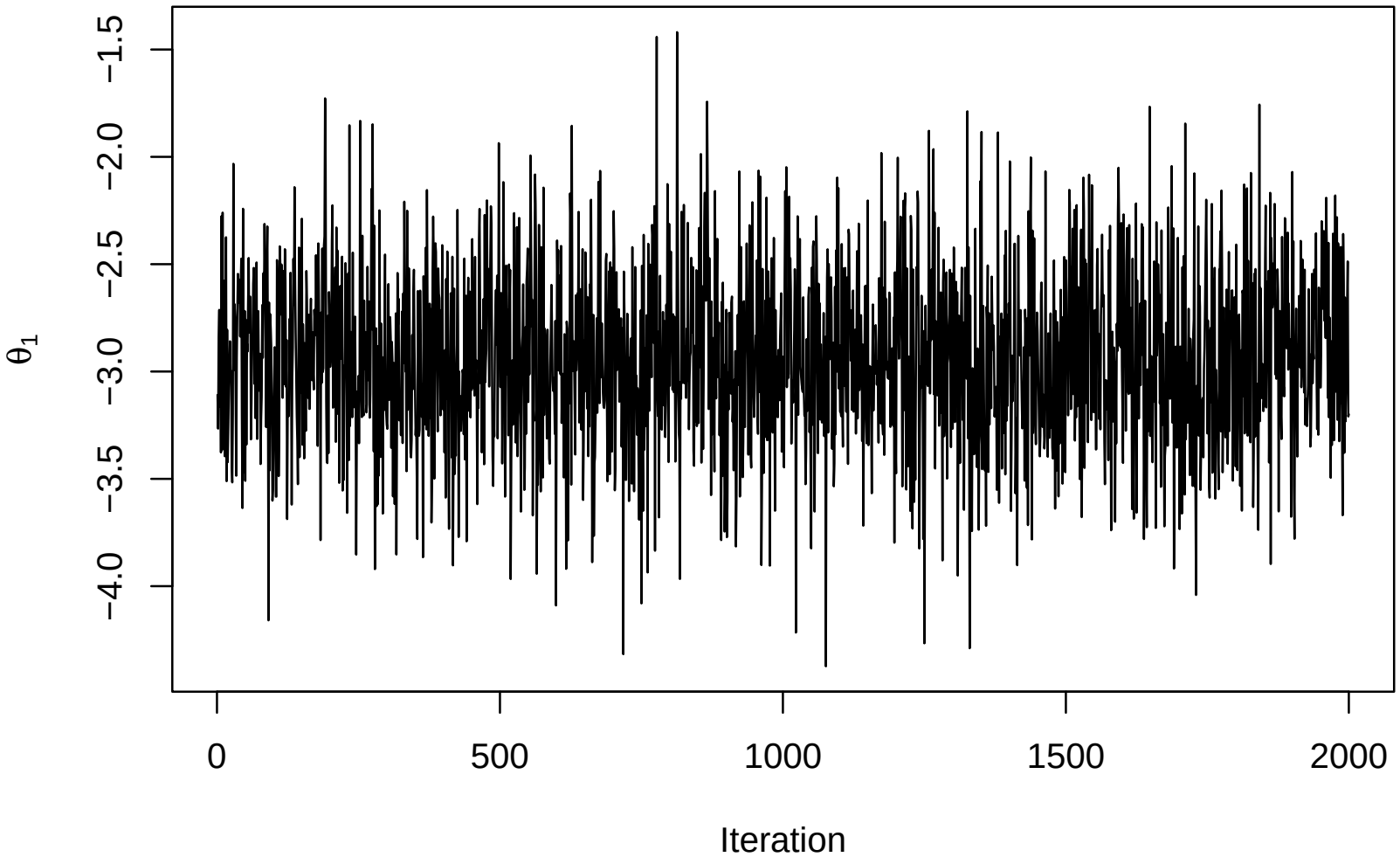
```
## [1]  0.6297057 -2.7808181  1.1700293
```

Random effects model: simulation

```
#Gibbs sampler
ico<-0;th.samp<-matrix(0,nrow=nsamp,ncol=M)
sig.samp<-phi.samp<-tau.samp<-rep(0,nsamp)
for(iter in 1:nits){
    #Update th
    mval<-(ni*Ybar/old.sigsq+old.phi/old.tausq)/(ni/old.sigsq+1/old.tausq)
    Mval<-sqrt(1/(ni/old.sigsq+1/old.tausq))
    old.th<-rnorm(M,mval,Mval)
    #Update phi
    mval<-(sum(old.th)/old.tausq)/(M/old.tausq+lam0/old.tausq)
    Mval<-sqrt(1/(M/old.tausq+lam0/old.tausq))
    old.phi<-rnorm(1,mval,Mval)
    #Update sigma^2
    old.ssq<-rep(0,M)
    for(i in 1:M){
        old.ssq[i]<-sum((Y[[i]]-old.th[i])^2)
    }
    an<-a0+n; bn<-b0+sum(old.ssq)
    old.sigsq<-1/rgamma(1,an/2,bn/2)
    #Update tau^2
    gn<-g0+M+1; hn<-h0+sum((old.th-old.phi)^2)+lam0*old.phi^2
    old.tausq<-1/rgamma(1,gn/2,hn/2)
    if(iter > nburn & iter %% nthin ==0){
        ico<-ico+1
        th.samp[ico,]<-old.th
        sig.samp[ico]<-old.sigsq
        phi.samp[ico]<-old.phi
        tau.samp[ico]<-old.tausq
    }
}
```

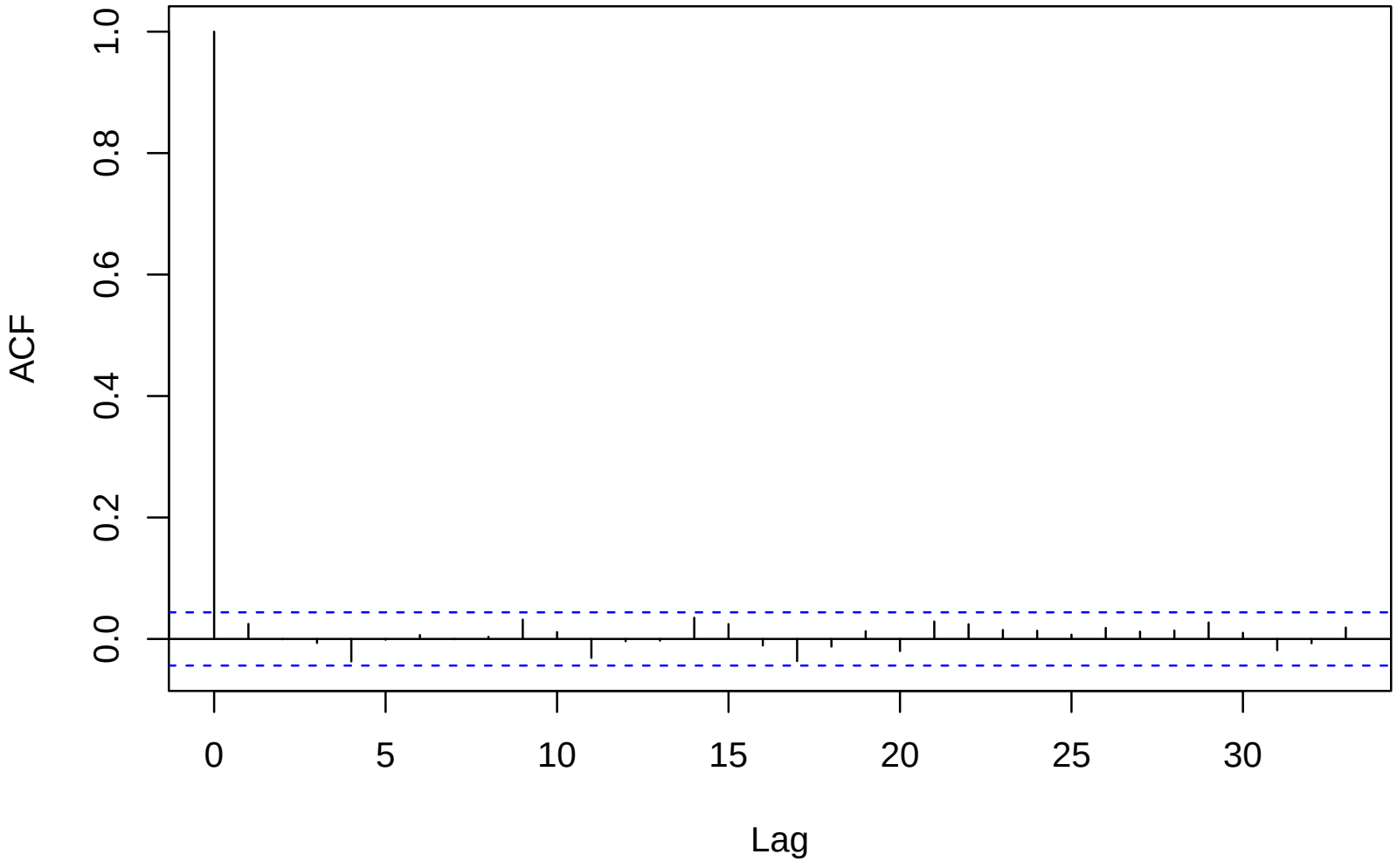
Random effects model: simulation

```
par(mar=c(4,4,1,2));plot(th.samp[,1],type='l',ylab=expression(theta[1]),xlab='Iteration')
```



Random effects model: simulation

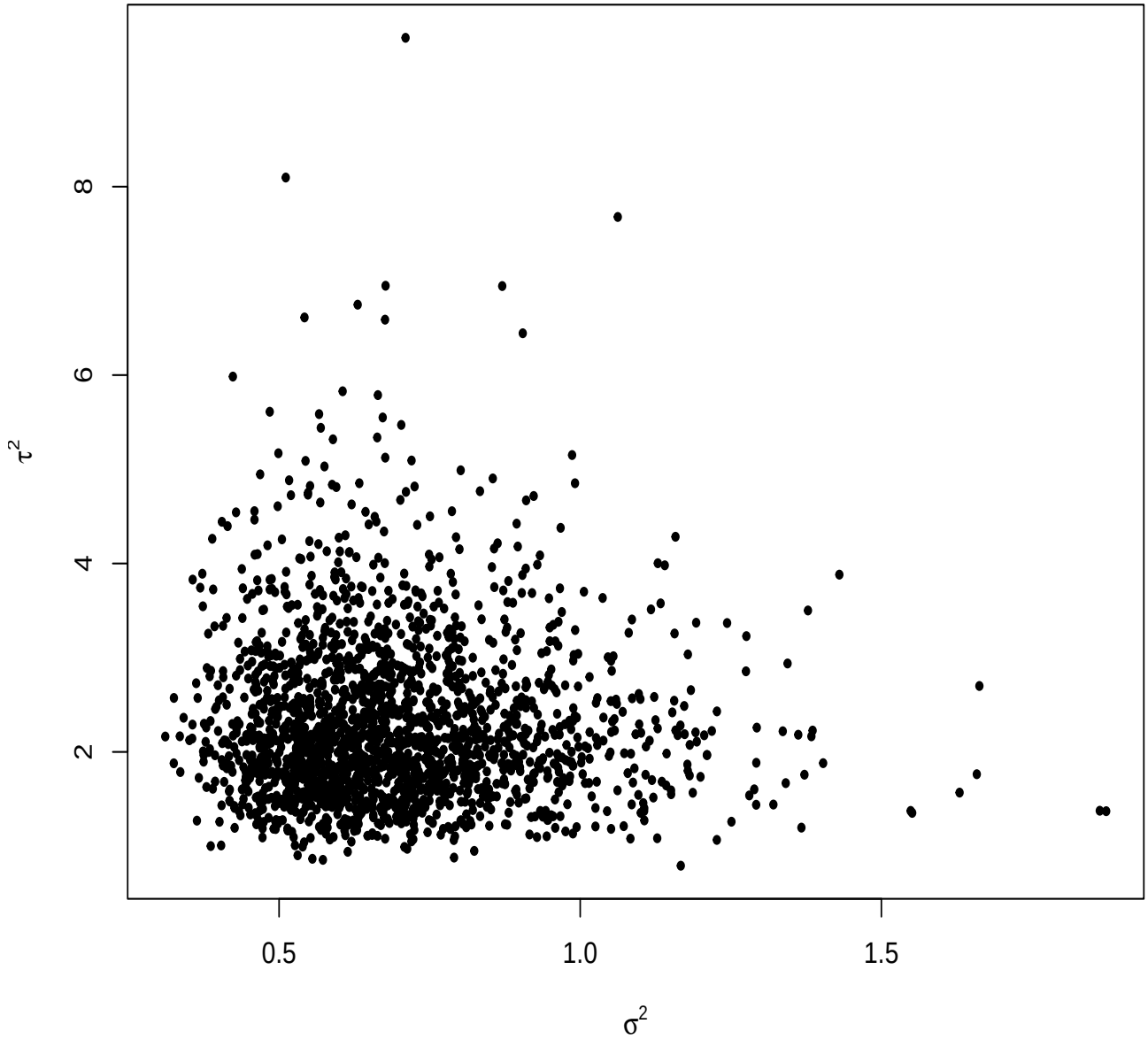
```
par(mar=c(4,4,1,2));acf(th.samp[,1],main='')
```



Random effects model: simulation

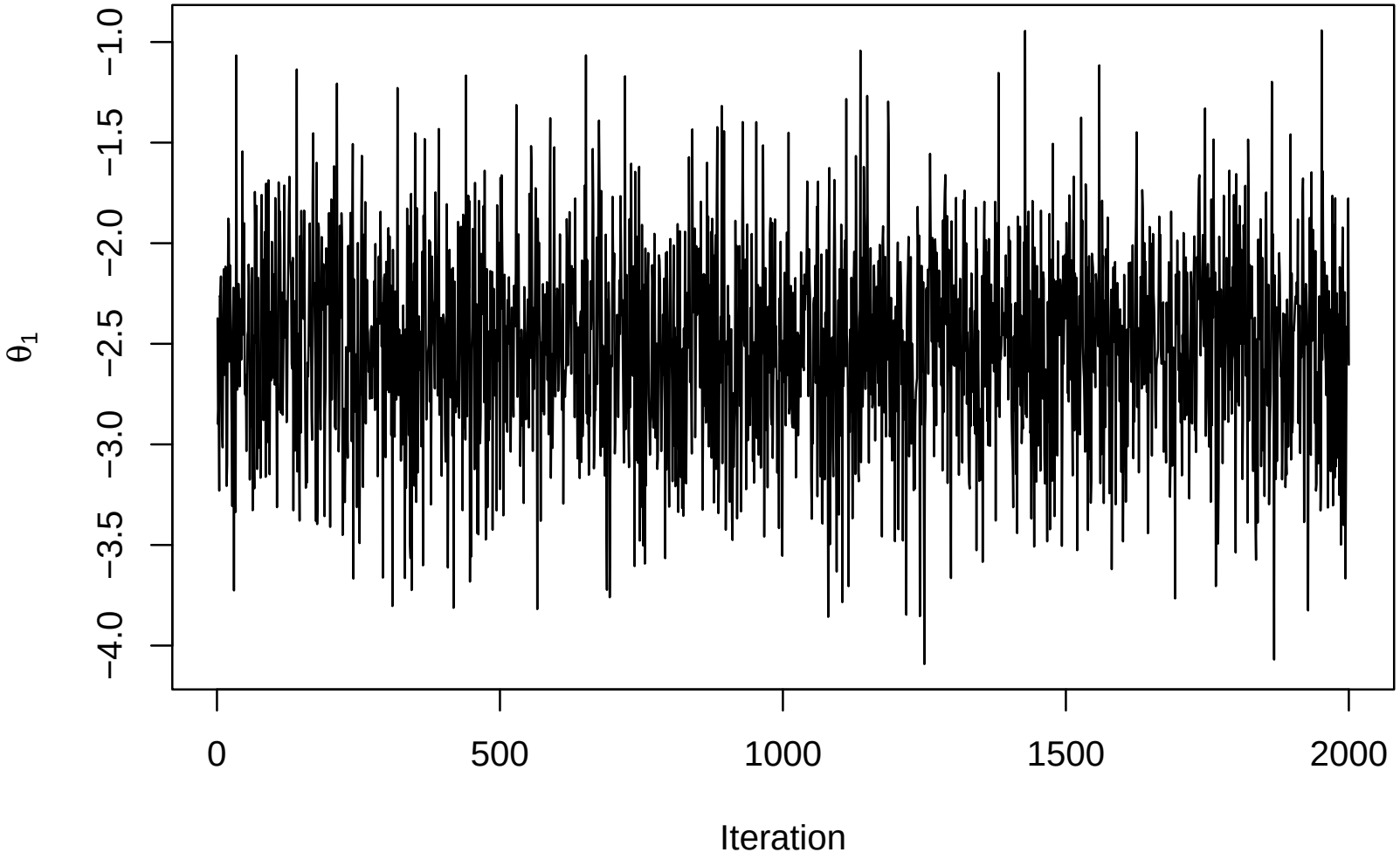
```
par(mar=c(4,4,1,2));
plot(sig.samp,tau.samp,pch=19,cex=0.6,ylab=expression(tau^2),xlab=expression(sigma^2))
```

Random effects model: simulation



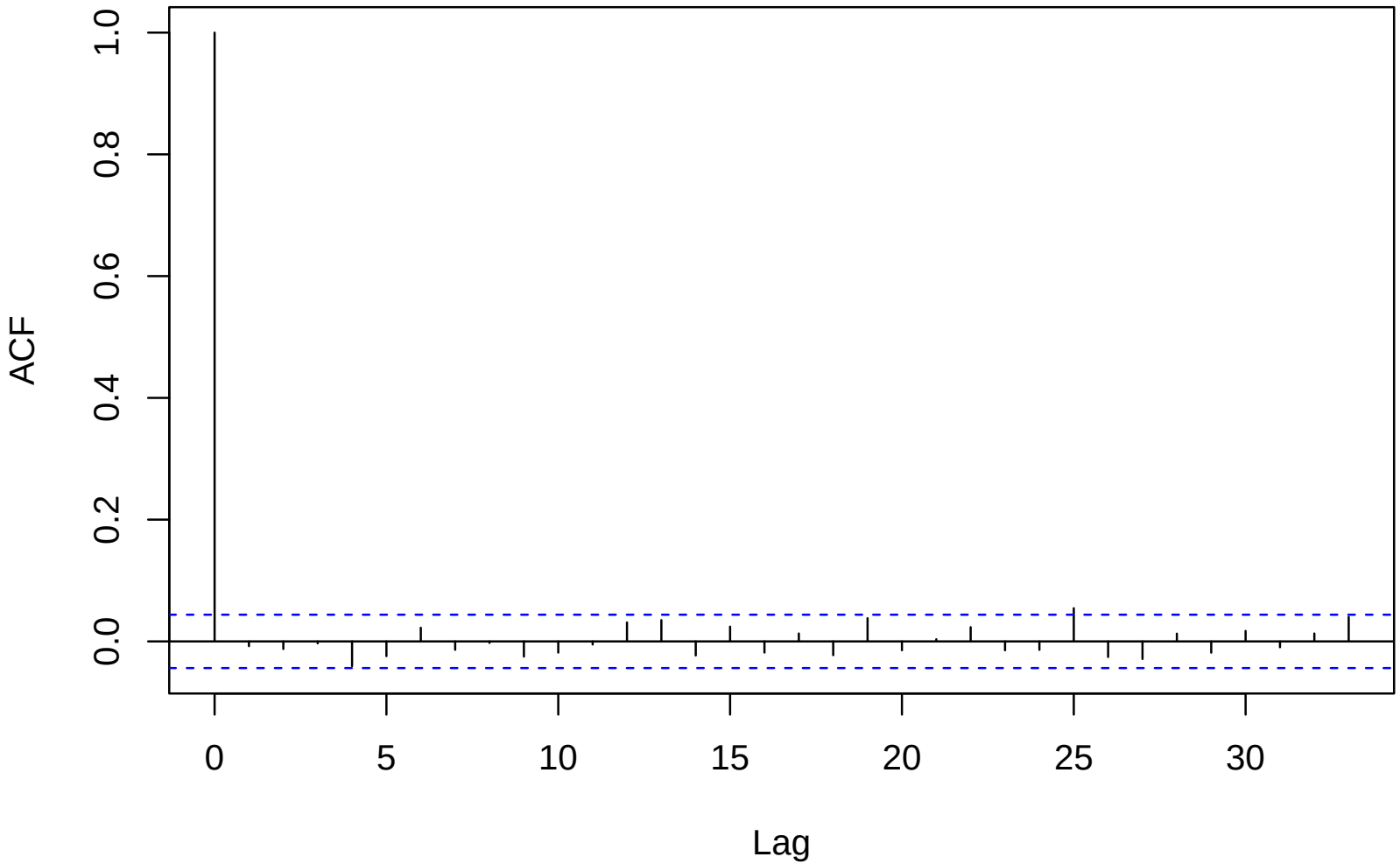
Random effects model: simulation

```
par(mar=c(4,4,1,2));plot(phi.samp,type='l',ylab=expression(theta[1]),xlab='Iteration')
```



Random effects model: simulation

```
par(mar=c(4,4,1,2));acf(phi.samp,main='')
```



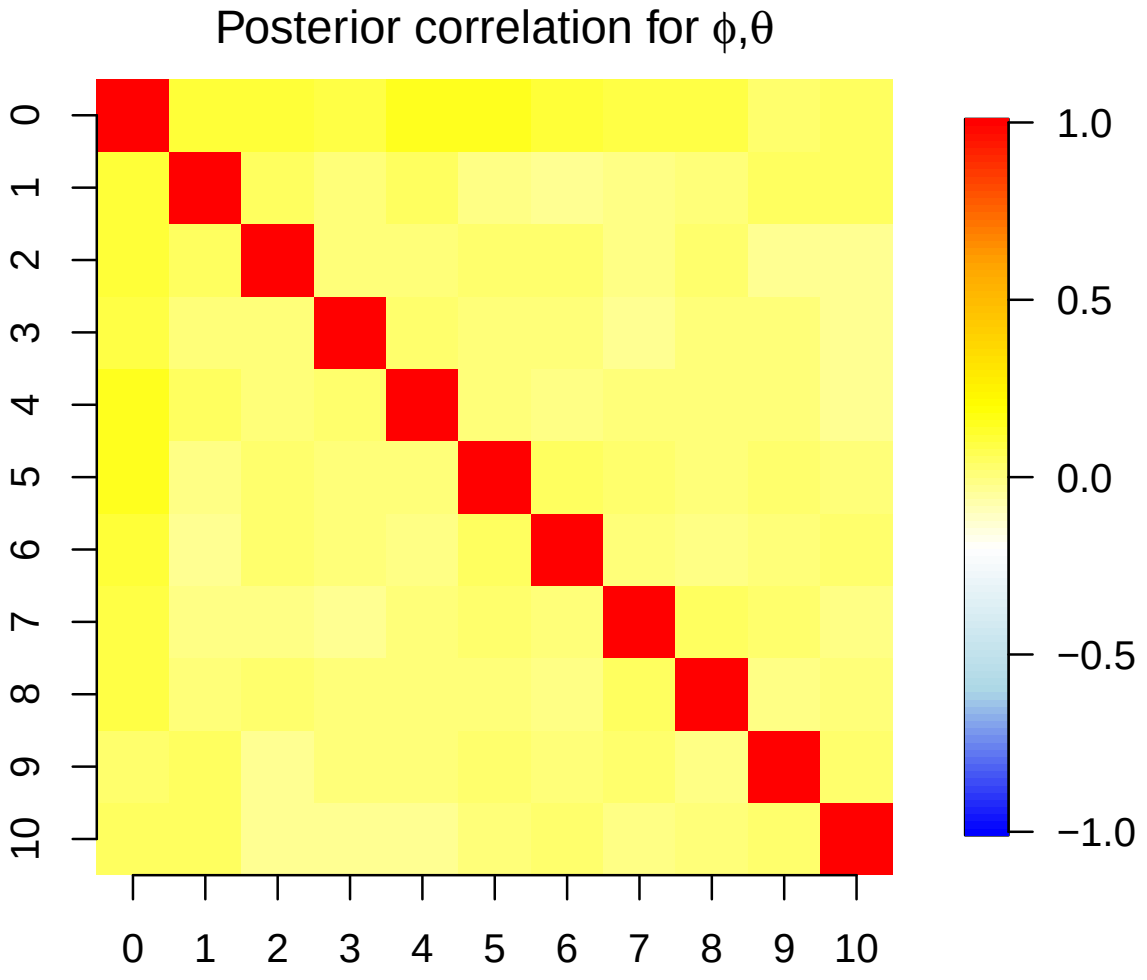
Random effects model: simulation

```
suppressMessages(library(fields,quietly=TRUE))
par(pty='s',mar=c(4,4,2,2))
colfunc <- colorRampPalette(c("blue","lightblue","white","yellow","orange","red"))

old.param<-cbind(phi.samp,th.samp)

post.cor<-post.cor.flip<-cor(old.param)
for(i in 1:(M+1)){
    post.cor.flip[i,]<-post.cor[M-i+2,]
}
for(j in 1:(M+1)){
    post.cor.flip[,j]<-post.cor[,M-j+2]
}
image.plot(0:(M+1),0:(M+1),post.cor.flip,col=colfunc(100),zlim=range(-1,1),
           xlab='',ylab='',cex.axis=0.8,axes=FALSE)
axis(1,at=c(0:M)+0.5,labels=c(0:M))
axis(2,at=c(0:M)+0.5,labels=rev(c(0:M)))
title(expression(paste('Posterior correlation for ',phi,',',theta)))
```

Random effects model: simulation



Random effects model: simulation

```
#Posterior correlation in initial parameterization
round(post.cor,2)
```

```
##           phi.samp
## phi.samp      1.00  0.11  0.12  0.08  0.15 0.14  0.12  0.09  0.08  0.04  0.05
##               0.11  1.00  0.04  0.00  0.04 0.00 -0.02  0.00  0.00  0.05  0.05
##               0.12  0.04  1.00  0.01  0.01 0.02  0.03 -0.01  0.03 -0.03 -0.03
##               0.08  0.00  0.01  1.00  0.02 0.01  0.00 -0.03  0.01  0.00 -0.02
##               0.15  0.04  0.01  0.02  1.00 0.01 -0.02  0.01  0.02  0.02 -0.03
##               0.14  0.00  0.02  0.01  0.01 1.00  0.05  0.04  0.01  0.04  0.01
##               0.12 -0.02  0.03  0.00 -0.02 0.05  1.00  0.02 -0.01  0.01  0.04
##               0.09  0.00 -0.01 -0.03  0.01 0.04  0.02  1.00  0.05  0.02 -0.01
##               0.08  0.00  0.03  0.01  0.02 0.01 -0.01  0.05  1.00 -0.01  0.01
##               0.04  0.05 -0.03  0.00  0.02 0.04  0.01  0.02 -0.01  1.00  0.02
##               0.05  0.05 -0.03 -0.02 -0.03 0.01  0.04 -0.01  0.01  0.02  1.00
```

Random effects model: simulation

New parameterization:

$$\theta_i = \phi + \delta_i$$

so that

$$\begin{pmatrix} \phi \\ \delta_1 \\ \delta_2 \\ \vdots \\ \delta_M \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} \phi \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_M \end{pmatrix}$$

Random effects model: simulation

```
L<-matrix(0,M+1,M+1)
L[,1]<--1
diag(L)<-1

L
```

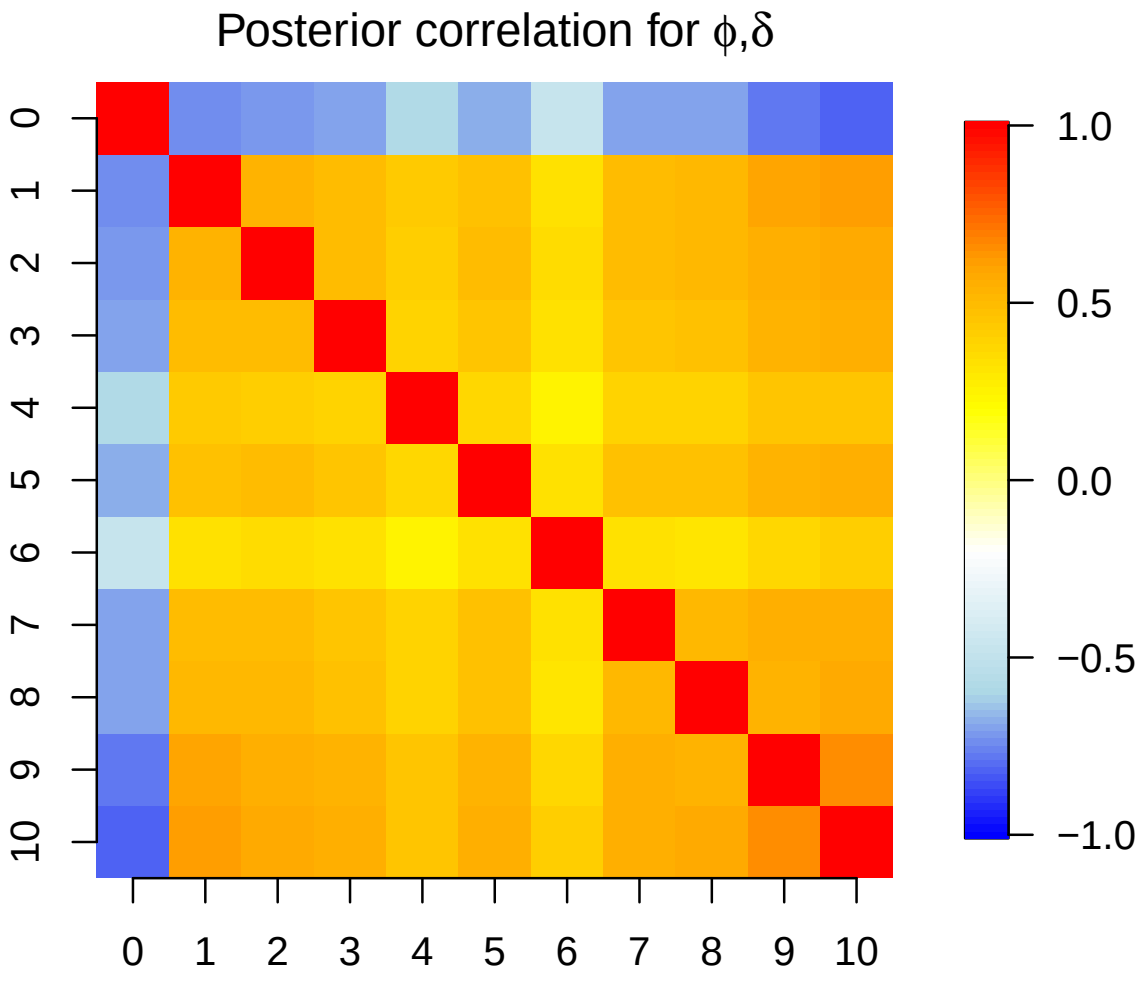
```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
##  [1,]    1    0    0    0    0    0    0    0    0     0     0
##  [2,]   -1    1    0    0    0    0    0    0    0     0     0
##  [3,]   -1    0    1    0    0    0    0    0    0     0     0
##  [4,]   -1    0    0    1    0    0    0    0    0     0     0
##  [5,]   -1    0    0    0    1    0    0    0    0     0     0
##  [6,]   -1    0    0    0    0    1    0    0    0     0     0
##  [7,]   -1    0    0    0    0    0    1    0    0     0     0
##  [8,]   -1    0    0    0    0    0    0    1    0     0     0
##  [9,]   -1    0    0    0    0    0    0    0    1     0     0
## [10,]   -1    0    0    0    0    0    0    0    0     1     0
## [11,]   -1    0    0    0    0    0    0    0    0     0     1
```

```
new.param<-t( L %*% t(old.param))
```

Random effects model: simulation

```
par(pty='s',mar=c(4,4,2,2))
colfunc <- colorRampPalette(c("blue","lightblue","white","yellow","orange","red"))
post.cor<-post.cor.flip<-cor(new.param)
for(i in 1:(M+1)){
    post.cor.flip[i,]<-post.cor[M-i+2,]
}
for(j in 1:(M+1)){
    post.cor.flip[,j]<-post.cor[,M-j+2]
}
image.plot(0:(M+1),0:(M+1),post.cor.flip,col=colfunc(100),zlim=range(-1,1),
           xlab='',ylab='',cex.axis=0.8,axes=FALSE)
axis(1,at=c(0:M)+0.5,labels=c(0:M))
axis(2,at=c(0:M)+0.5,labels=rev(c(0:M)))
title(expression(paste('Posterior correlation for ',phi,',',delta)))
```

Random effects model: simulation



Random effects model: simulation

```
#Posterior correlation in new parameterization
round(post.cor,2)
```

```
##         [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11]
##  [1,]  1.00 -0.73 -0.72 -0.69 -0.57 -0.67 -0.48 -0.69 -0.70 -0.79 -0.82
##  [2,] -0.73  1.00  0.54  0.50  0.43  0.48  0.33  0.50  0.51  0.60  0.62
##  [3,] -0.72  0.54  1.00  0.50  0.41  0.49  0.35  0.49  0.51  0.56  0.58
##  [4,] -0.69  0.50  0.50  1.00  0.40  0.46  0.33  0.46  0.48  0.54  0.56
##  [5,] -0.57  0.43  0.41  0.40  1.00  0.38  0.25  0.39  0.40  0.46  0.45
##  [6,] -0.67  0.48  0.49  0.46  0.38  1.00  0.34  0.48  0.47  0.55  0.56
##  [7,] -0.48  0.33  0.35  0.33  0.25  0.34  1.00  0.34  0.32  0.38  0.41
##  [8,] -0.69  0.50  0.49  0.46  0.39  0.48  0.34  1.00  0.51  0.55  0.56
##  [9,] -0.70  0.51  0.51  0.48  0.40  0.47  0.32  0.51  1.00  0.54  0.57
## [10,] -0.79  0.60  0.56  0.54  0.46  0.55  0.38  0.55  0.54  1.00  0.65
## [11,] -0.82  0.62  0.58  0.56  0.45  0.56  0.41  0.56  0.57  0.65  1.00
```