

STAT7016 Assignment 3

Rui Qiu

2017-09-24

Settings

```
library(mvtnorm)
library(LearnBayes)
library(MCMCglmm)
library(lattice)
library(coda)
set.seed(7016)
```

Problem 1 Parameter expansion and the t model

Suppose we have n independent data points from the $t_\nu(\mu, \sigma^2)$ distribution and we assume the degrees of freedom ν is known. The t likelihood for each data point is equivalent to the model:

$$y_i \sim N(\mu, V_i) \\ V_i \sim \text{InvGamma}(\nu/2, \nu\sigma^2/2)$$

A Gibbs sampler can be used to obtain posterior draws of μ, σ^2 , and each V_i . However, convergence will be slow if simulation draw of σ^2 is close to zero. We can add an extra parameter as follows:

$$y_i \sim N(\mu, \alpha^2 U_i) \\ U_i \sim \text{InvGamma}(\nu/2, \nu\tau^2/2)$$

where $\alpha > 0$ is an additional scale parameter. The parameter α has no meaning and its only role is to allow the Gibbs sampler to move in more directions to avoid getting stuck. Assuming a uniform prior distribution on $\mu, \log \tau$ and $\log \alpha$, derive the four steps of the Gibbs sampler for the parameter expanded model. That is, derive the conditional distributions for $U_i (i = 1, \dots, n), \mu, \tau^2$ and α^2 .

Solution

Note that $V_i \sim \text{InvGamma}(\nu/2, \nu\sigma^2/2)$, i.e. $V_i \sim \text{Inv-}\chi^2(\nu, \sigma^2)$. Also $U_i \sim \text{Inv-}\chi^2(\nu, \tau^2)$.

First we take a look at the original parametrization (without α).

1. Conditional posterior distributions of V_i .

Since V_i is a variance of a normal distribution with a inverse-Gamma (scaled inverse Chi-squared) prior, so that the posterior distribution should also follow a inverse Chi-squared distribution.

$$V_i \mid \mu, \sigma^2, \nu, y \sim \text{Inv-}\chi^2 \left(\nu + 1, \frac{\nu\sigma^2 + (y_i - \mu)^2}{\nu + 1} \right).$$

2. Conditional posterior distribution of μ .

μ is conditional on V, σ^2, ν, y , all its information comes from y_i 's, μ also has a uniform prior such that

$$\mu \mid V, \sigma^2, \nu, y \sim N \left(\frac{\sum_{i=1}^n \frac{1}{V_i} y_i}{\sum_{i=1}^n \frac{1}{V_i}}, \frac{1}{\sum_{i=1}^n \frac{1}{V_i}} \right).$$

3. Conditional posterior distribution of σ^2 , all its information comes from the variances of V_i 's such that

$$\begin{aligned} p(\sigma^2 \mid \mu, V, \nu, y) &\propto \sigma^{-2} \prod_{i=1}^n \sigma^\nu e^{-\nu \sigma^2 / (2V_i)} \\ &= (\sigma^2)^{n\nu/2-1} \exp \left(-\frac{\nu}{2} \sum_{i=1}^n \frac{1}{V_i} \sigma^2 \right) \\ &\propto \text{Gamma} \left(\sigma^2 \mid \frac{n\nu}{2}, \frac{\nu}{2} \sum_{i=1}^n \frac{1}{V_i} \right) \end{aligned}$$

While we introduce an additional scale parameter α , note that $\alpha^2 U_i$ works like V_i did, and $\alpha^2 \tau^2$ as σ^2 .

So now, the four steps of the Gibbs sampler for parameter expanded model are:

1. Conditional posterior distribution of U_i , very similar to the one of V_i before:

$$U_i \mid \alpha^2, \mu, \tau^2, \nu, y \sim \text{Inv-}\chi^2 \left(\nu + 1, \frac{\nu \tau^2 + ((y_i - \mu)/\alpha)^2}{\nu + 1} \right).$$

2. Conditional posterior distribution of μ is totally identical to the old one:

$$\mu \mid \alpha^2, \tau^2, U, \nu, y \sim N \left(\frac{\sum_{i=1}^n \frac{1}{U_i \alpha^2} y_i}{\sum_{i=1}^n \frac{1}{U_i \alpha^2}}, \frac{1}{\sum_{i=1}^n \frac{1}{U_i \alpha^2}} \right).$$

3. Conditional posterior distribution of τ^2 :

$$\tau^2 \mid \alpha^2, U, \mu, \nu, y \sim \text{Gamma} \left(\frac{n\nu}{2}, \frac{\nu}{2} \sum_{i=1}^n \frac{1}{U_i} \right).$$

4. Since α^2 is also a normal variance parameter like V_i and U_i , so the conditional posterior distribution of α^2 should also follow a Inverse Chi-squared distribution.

$$\alpha^2 \mid \mu, \tau^2, U, \nu, y \sim \text{Inv-}\chi^2 \left(n, \frac{1}{n} \sum_{i=1}^n \frac{(y_i - \mu)^2}{U_i} \right).$$

Problem 2

The file `interexp.dat` contains data from an experiment that was interrupted before all the data could be gathered. Of interest was the difference in reaction times of experimental subjects when they were given stimulus A versus stimulus B. Each subject is tested under one of the two stimuli on their first day of participation in the study, and is tested under the other stimulus at some later date. Unfortunately the experiment was interrupted before it was finished, leaving the researchers with 26 subjects with both A and B responses, 15 subjects with only A responses and 17 subjects with only B responses.

- (a) Calculate empirical estimates $\theta_A, \theta_B, \rho, \sigma_A^2, \sigma_B^2$ from the data using the commands `mean`, `cor` and `var`. Use *all* the A responses to get $\hat{\theta}_A$ and $\hat{\sigma}_A^2$, and use *all* the B responses to get $\hat{\theta}_B$ and $\hat{\sigma}_B^2$. Use only the complete data cases to get $\hat{\rho}$.
- (b) For each person i with only an A response, impute a B response as

$$\hat{y}_{i,B} = \hat{\theta}_B + (y_{i,A} - \hat{\theta}_A)\hat{\rho}\sqrt{\hat{\sigma}_B^2/\hat{\sigma}_A^2}$$

For each person i with only a B response, impute an A response as

$$\hat{y}_{i,A} = \hat{\theta}_A + (y_{i,B} - \hat{\theta}_B)\hat{\rho}\sqrt{\hat{\sigma}_A^2/\hat{\sigma}_B^2}$$

You now have two “observations” for each individual. Do a paired sample t-test and obtain a 95% confidence interval for $\hat{\theta}_A - \hat{\theta}_B$.

- (c) Use the Jeffrey’s prior for multivariate normal data ($p_J(\theta, \Sigma) \propto |\Sigma|^{-(p+2)/2}$) where p is the dimension of each observable vector \mathbf{y}_i for the parameters, and implement a Gibbs sampler that approximates the joint distribution of the parameters and the missing data. Compute a posterior mean for $\theta_A - \theta_B$ as well as a 95% posterior confidence interval for $\theta_A - \theta_B$. Compare these results with the results from part (b) and discuss.

Note, you may use the following results for the posterior distributions:

$$\begin{aligned}\Sigma | \mathbf{y}_1, \dots, \mathbf{y}_n &\sim \text{Inverse-Wishart}(n-1, S^{-1}) \\ \theta | \Sigma, \mathbf{y}_1, \dots, \mathbf{y}_n &\sim \text{Multivariate Normal}(\bar{\mathbf{y}}, \Sigma/n)\end{aligned}$$

where $S = \sum_{i=1}^n (\mathbf{y}_i - \bar{\mathbf{y}})(\mathbf{y}_i - \bar{\mathbf{y}})^T$.

- (d) Contrast the two expressions for the imputed values $\hat{y}_{i,B} = \hat{\theta}_B$ and $\hat{y}_{i,B} = \hat{\theta}_B + (y_{i,A} - \hat{\theta}_A)\hat{\rho}\sqrt{\hat{\sigma}_B^2/\hat{\sigma}_A^2}$. Explain (in words) the differences in any underlying assumptions between the two expressions.

Solution

(a)

```
rm(list=ls())
data <- read.table("interexp.dat",header=T)
# head(data)
# empirical estimates
theta.A.hat <- mean(data$yA,na.rm=T) # use obs A to estimate missing A
var.A.hat <- var(data$yA,na.rm=T)
theta.B.hat <- mean(data$yB,na.rm=T) # use obs B to estimate missing B
var.B.hat <- var(data$yB,na.rm=T)
rho.hat <- cor(data$yA,data$yB,use="complete.obs") # use only obs
```

```
data.frame(hatTHETA.A=theta.A.hat,hatSIGMA.A=var.A.hat,
           hatTHETA.B=theta.B.hat,hatSIGMA.B=var.B.hat,
           hatRHO=rho.hat)
```

```
##   hatTHETA.A hatSIGMA.A hatTHETA.B hatSIGMA.B   hatRHO
## 1   24.20049    4.0928   24.80535   4.691578 0.6164509
```

The empirical estimates of $\theta_A, \theta_B, \rho, \sigma_A^2, \sigma_B^2$ are shown in the table above, i.e. $\hat{\theta}_A = 24.20049, \hat{\theta}_B = 24.80535, \hat{\sigma}_A^2 = 4.0928, \hat{\sigma}_B^2 = 4.691578, \hat{\rho} = 0.6164509$.

(b)

```
impt <- data
# the implementation of imputation
for (i in 1:nrow(impt)){
  if (is.na(impt[i,1])) {
    impt[i,1] <-
      theta.A.hat+(impt[i,2]-theta.B.hat)*rho.hat*sqrt(var.A.hat/var.B.hat)
  }
  if (is.na(impt[i,2])) {
    impt[i,2] <-
      theta.B.hat+(impt[i,1]-theta.A.hat)*rho.hat*sqrt(var.B.hat/var.A.hat)
  }
}
t.test(impt[,1],impt[,2],paired=T) # paired t-test
```

```
##
## Paired t-test
##
## data: impt[, 1] and impt[, 2]
## t = -3.2807, df = 57, p-value = 0.00177
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.9850730 -0.2383347
## sample estimates:
## mean of the differences
## -0.6117038
```

After the imputation, we get two vectors of “full” data, and paired t-test gives us information about differences: - the mean of the differences is -0.6117038 . - the 95% confidence interval of the difference is $(-0.9850730, -0.2383347)$.

Generally speaking, we can be sure that Group A has smaller values than Group B.

(c)

```
# priors
ybar <- apply(data,2,mean,na.rm=TRUE)
complete <- which(complete.cases(data))
miss.A <- which(is.na(data$yA))
miss.B <- which(is.na(data$yB))
n.complete <- length(complete) # 1:26
S <- data.frame(matrix(rep(0,4),ncol=2)) # 2x2 var-cov matrix
colnames(S) <- c("yA", "yB")
rownames(S) <- c("yA", "yB")
```

```

for (i in 1:n.complete) {
  S <- S+(t(data[i,])-ybar)%*%t(t(data[i,])-ybar)
}
S <- as.matrix(S)
n <- nrow(data)

sims <- 10000
# store the sampled yA and yB in matrices, although not used
# yA.sample <- matrix(nrow=sims,ncol=n)
# yB.sample <- matrix(nrow=sims,ncol=n)
theta.sample <- matrix(nrow=sims,ncol=2)

# initial values
Sigma <- S
theta <- ybar
Y <- as.matrix(impt)

# Gibbs sampler
# Gibbs-sampled parameters and missing data are stored in
# theta.sample, yA.sample, yB.sample
# the sampled sigmaA, sigmaB are not stored, however, "burner sigmas"

for (i in 1:sims) {
  # update theta
  ybar.sample <- apply(Y,2,mean)
  theta <- rmvnorm(1,ybar.sample,Sigma/n)
  theta.sample[i,] <- theta

  # update Sigma
  Sn <- S+t(Y-c(theta))%*%(Y-c(theta))+t(t(c(theta)-ybar.sample))%*%
    t(c(theta)-ybar.sample)
  Sigma <- solve(rWishart(1,n-1,solve(Sn))[,1])
  rho <- cov2cor(Sigma)[1,2]
  sigmaA.sq <- Sigma[1,1]
  sigmaB.sq <- Sigma[2,2]

  # update missing data by case
  # if missing A
  for (j in miss.A) {
    Y[j,1] <- rnorm(1,theta[1]+(rho*sqrt(sigmaA.sq/sigmaB.sq))*
      (Y[j,"yB"]- theta[2]),sqrt(sigmaA.sq*(1-rho)))
  }
  # if missing B
  for (k in miss.B) {
    Y[k,2] <- rnorm(1,theta[2]+(rho*sqrt(sigmaB.sq/sigmaA.sq))*
      (Y[k,"yA"]-theta[1]),sqrt(sigmaB.sq*(1-rho)))
  }
  # yA.sample[i,] <- Y[,1]
  # yB.sample[i,] <- Y[,2]
}

mean(theta.sample[1,]-theta.sample[,2])

## [1] -0.6106031

```

```
quantile(theta.sample[1,]-theta.sample[,2],c(0.025, 0.975))
```

```
##          2.5%          97.5%  
## -1.6457114  0.4216818
```

```
# unit info prior -> posterior mean -0.3665874  
# unit info prior -> posterior ci -1.1753716  0.4294017
```

After conducting a Gibbs sampler approximation, the 95% posterior confidence interval for $\theta_A - \theta_B$ based on a Jeffrey's prior is $(-1.6457114, 0.4216818)$, which is wider than the previous confidence interval. On the other hand, the posterior mean (-0.6106031) stays the almost the same.

(d)

By comparing two imputed values, we easily find out that the second one has one more term $(y_{i,A} - \hat{\theta}_A)\hat{\rho}\sqrt{\hat{\sigma}_B/\hat{\sigma}_A}$ which is mainly controlled by the estimated correlation $\hat{\rho}$ between group A and B . Typically, the idea behind the second imputation is that group A and group B are correlated, while the first imputation ignores such correlation.

Problem 3

A population of 532 women living near Phoenix, Arizona were tested for diabetes. Other information was gathered from these women at the time of testing, including number of pregnancies, glucose level, blood pressure, skin fold thickness, body mass index, diabetes pedigree and age. This information appears in the file `azdiabetes.dat`. Model the joint distribution of these variables for the diabetics and non-diabetics separately, using a multivariate normal distribution:

- (a) For both groups separately, use the following type of unit information prior, where $\hat{\Sigma}$ is the sample covariance matrix.
- (b) $\mu_0 = \bar{\mathbf{y}}, \Lambda_0 = \hat{\Sigma}$
- (ii) $S_0 = \hat{\Sigma}, \nu_0 = p + 2 = 9$

Generate at least 10,000 (Markov Chain) Monte Carlo samples for $\{\theta_d, \Sigma_d\}$ and $\{\theta_n, \Sigma_n\}$, the model parameters for diabetics and non-diabetics respectively. For each of the seven variables $j \in \{1, \dots, 7\}$, compare the marginal posterior distributions of $\theta_{d,j}$ and $\theta_{n,j}$. Which variables seem to differ between the two groups? Also obtain $Pr(\theta_{d,j} > \theta_{n,j} \mid \mathbf{Y})$ for each $j \in \{1, \dots, 7\}$.

- (b) Obtain the posterior means of Σ_d and Σ_n , and plot the entries versus each other. What are the main differences, if any?

Solution

(a)

In the following codes, we use subscript `d` for diabetics group and `n` for non-diabetics group.

Also note that we separate the iterative steps in Gibbs sampler from the for-loop and write them as helper functions `helper.sample.theta` and `helper.sample.Sigma`.

```
rm(list=ls())
data <- read.table("azdiabetes.dat",header=T)

# prior
Y.n <- data[data[,8]=="No",,,-8]
Y.d <- data[data[,8]=="Yes",,,-8]
mu0.n <- theta.n <- apply(Y.n,2,mean)
mu0.d <- theta.d <- apply(Y.d,2,mean)
lambda0.n <- S0.n <- Sigma.n <- cov(Y.n)
lambda0.d <- S0.d <- Sigma.d <- cov(Y.d)
nu0.d <- nu0.n <- 9

sims <- 10000
thetas.d <- matrix(nrow=sims,ncol=length(theta.d))
thetas.n <- matrix(nrow=sims,ncol=length(theta.n))
Sigmas.d <- matrix(nrow=sims,ncol=length(as.vector(Sigma.d)))
Sigmas.n <- matrix(nrow=sims,ncol=length(as.vector(Sigma.n)))

helper.sample.theta <- function(Y,Sigma,mu0,lambda0) {
  y.bar <- apply(Y,2,mean)
  n <- nrow(Y)
  theta.var <- solve(solve(lambda0)+n*solve(Sigma))
  theta.mean <- theta.var%*(solve(lambda0)%*mu0+n*solve(Sigma)%*y.bar)
  return(rmvnorm(1,theta.mean,theta.var))
}
```

```

helper.sample.Sigma <- function(Y,theta,S0,nu0) {
  n <- nrow(Y)
  Sn <- S0+(t(Y)-c(theta))%*%t(t(Y)-c(theta))
  return(solve(rWishart(1,nu0+n,solve(Sn))[, ,1]))
}

for (i in 1:sims) {
  # diabetics
  theta.d <- helper.sample.theta(Y.d,Sigma.d,mu0.d,lambda0.d)
  thetas.d[i,] <- theta.d
  Sigma.d <- helper.sample.Sigma(Y.d,theta.d,S0.d,nu0.d)
  Sigmas.d[i,] <- as.vector(Sigma.d)

  # non-diabetics
  theta.n <- helper.sample.theta(Y.n,Sigma.n,mu0.n,lambda0.n)
  thetas.n[i,] <- theta.n
  Sigma.n <- helper.sample.Sigma(Y.n,theta.n,S0.n,nu0.n)
  Sigmas.n[i,] <- as.vector(Sigma.n)
}

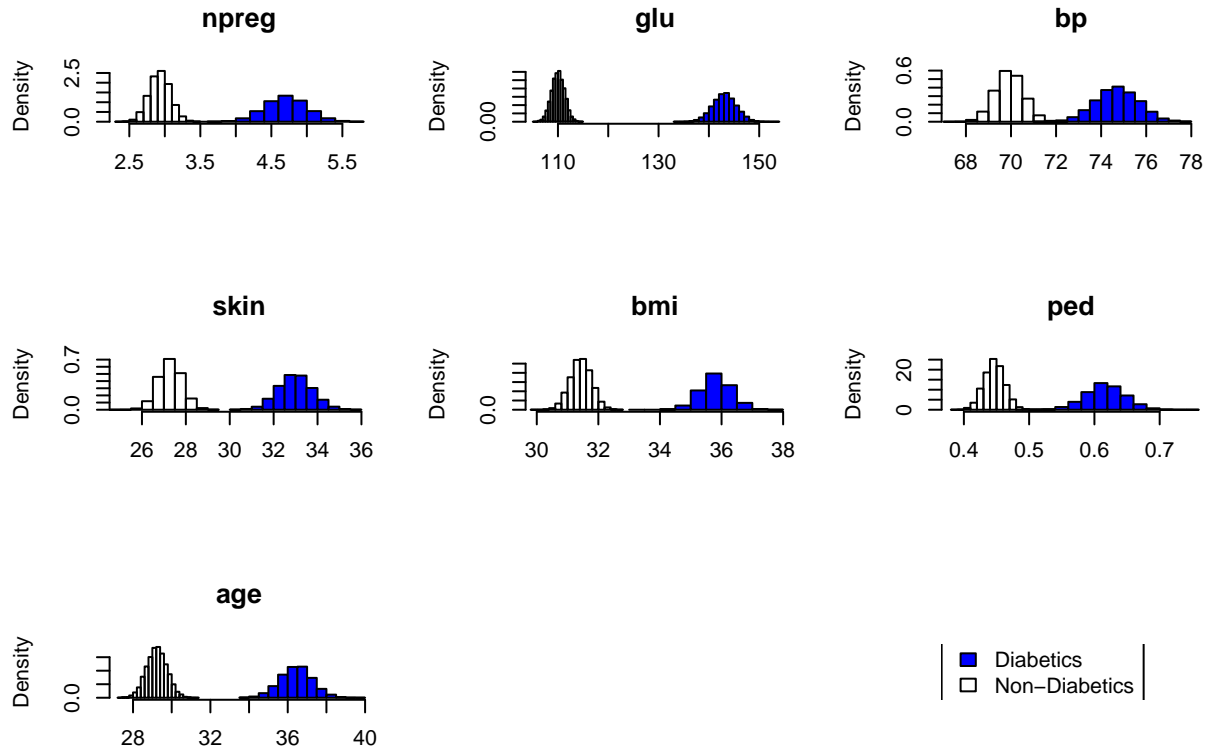
```

After sims=10000 runs, we take a closer look at the sampled parameters stored, and plot each variable by groups (side-by-side) so that direct comparison can be made visually.

```

par(mfrow=c(3,3))
for (j in 1:7) {
  plot.range <- range(cbind(thetas.d[,j],thetas.n[,j]))
  hist(thetas.n[,j],xlim=plot.range,freq=F,
       xlab="",main=names(data)[j])
  hist(thetas.d[,j],xlim=plot.range,freq=F,
       add=T,col="blue")
}
plot(plot.range,plot.range,xlab="",ylab="",type="n",axes=F)
plot(plot.range,plot.range,xlab="",ylab="",type="n",axes=F)
legend("left",fill=c("blue","white"),
      legend=c("Diabetics","Non-Diabetics"))

```

It looks like diabetics group and non-diabetics group behave rather differently in all 7 variables. To confirm, should also check the probability numerically.

```
post.prob <- c()
for (i in 1:7) {
  post.prob <- c(post.prob, mean(thetas.d[,i] > thetas.n[,i]))
}
post.prob
```

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

As we can see

$$Pr(\theta_{d,j} > \theta_{n,j} \mid \mathbf{Y}) = 1, \forall j \in \{1, \dots, 7\},$$

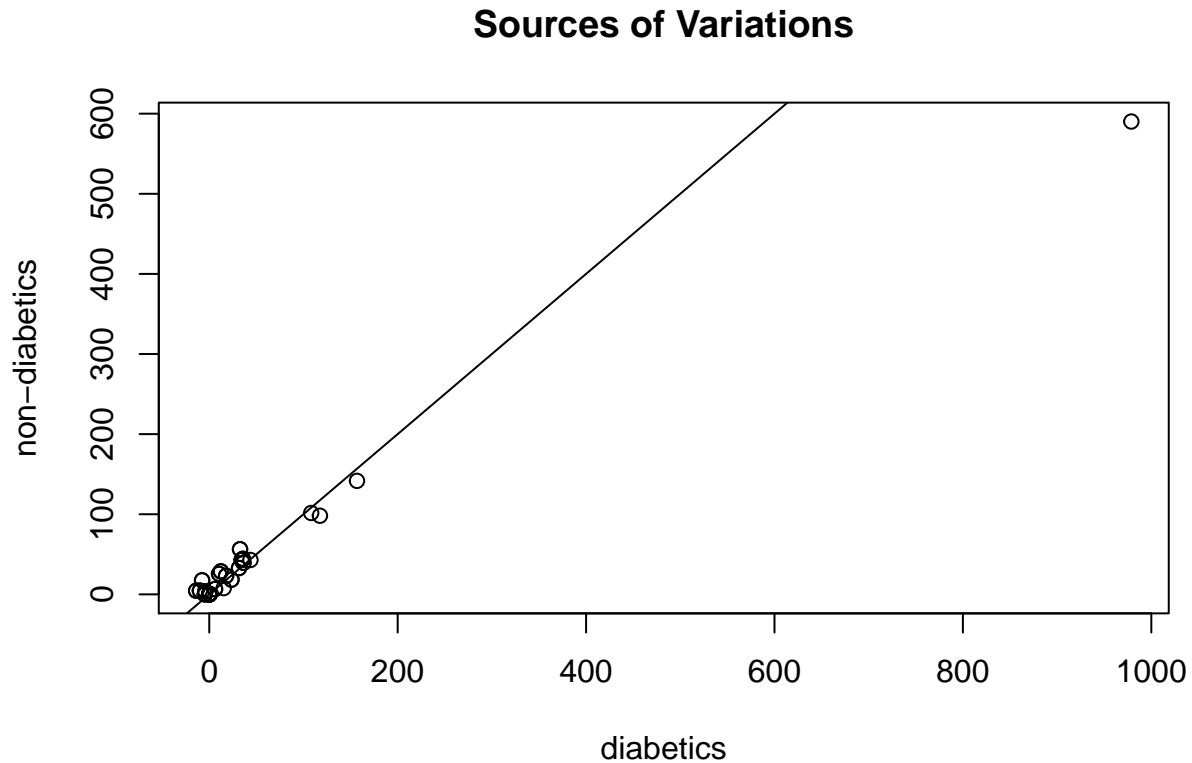
which agrees with the histograms we plotted above that the diabetics group have obviously greater values in all seven variables, i.e. **all variables seem to differ between the two groups**.

(b)

Since we want to find the posterior means of entries in a covariance matrix, we just plot 7×7 pairs of means in a plot with linear line $y = x$ so that any deviations would clearly stand out.

```
par(mfrow = c(1, 1))
# 7x7 var-cov
post.Sigmas.d.mean <- apply(Sigmas.d, 2, mean)
post.Sigmas.n.mean <- apply(Sigmas.n, 2, mean)
# post.Sigmas.d.mean
# post.Sigmas.n.mean
plot(post.Sigmas.d.mean, post.Sigmas.n.mean,
     xlab="diabetics", ylab="non-diabetics",
```

```
main="Sources of Variations")
abline(a=0,b=1)
```



We can definitely use `identify()` to find the outlier. However, we use `which()` function instead for simplicity:

```
which(abs(post.Sigmas.d.mean-post.Sigmas.n.mean)==
      max(abs(post.Sigmas.d.mean-post.Sigmas.n.mean)))
```

```
## [1] 9
```

Observe the covariance matrix and find out that the 9th covariance (actually this is a variance) is `glu` vs `glu`. So this is where the main difference originates.

Problem 4 (Probit regression)

A panel study followed 25 married couples over a period of five years. One item of interest is the relationship between divorce rates and the various characteristics of the couples. For example, the researchers would like to model the probability of divorce as a function of age differential, recorded as the man's age minus the women's age. The data can be found in the file `divorce.dat`. We will model these data with a probit regression, in which a binary variable Y_i is described in terms of an explanatory variable x_i via the following latent variable model:

$$\begin{aligned} Z_i &= \beta x_i + \epsilon_i \\ Y_i &= \delta_{(c, \infty)}(Z_i) \end{aligned}$$

where β and c are unknown coefficients, $\epsilon_1, \dots, \epsilon_n \stackrel{iid}{\sim} N(0, 1)$ and $\delta_{(c, \infty)}(z) = 1$ if $z > c$ and equals zero otherwise.

- Assuming $\beta \sim N(0, \tau_\beta^2)$ obtain the full conditional distribution $p(\beta \mid \mathbf{y}, \mathbf{x}, \mathbf{z}, c)$.
- Assuming $c \sim N(0, \tau_c^2)$, show that $p(c \mid \mathbf{y}, \mathbf{x}, \mathbf{z}, \beta)$ is a constrained normal density, i.e. proportional to a normal density but constrained to lie in an interval. Similarly, show that $p(z_i \mid \mathbf{y}, \mathbf{x}, \mathbf{z}_{-i}, \beta, c)$ is proportional to a normal density but constrained to be either above c or below c , depending on y_i .
- Letting $\tau_\beta^2 = \tau_c^2 = 16$, implement a Gibbs sampling scheme that approximates the joint distribution of \mathbf{Z}, β , and c . Compute the effective sample sizes of all unknown parameters (including all the Z_i 's). Also compute the autocorrelation function of the parameters and discuss the mixing of the Markov chain. (See Hoff Section 12.1.1 for a method on sampling from a constrained normal distribution).
- Obtain a 95% posterior confidence interval for β , as well as $Pr(\beta > 0 \mid \mathbf{y}, \mathbf{x})$.

Solution

(a)

The full conditional distribution $p(\beta \mid \mathbf{y}, \mathbf{x}, \mathbf{z}, c)$ only depends on \mathbf{z} in fact.

$$p(\beta \mid \mathbf{y}, \mathbf{z}, \mathbf{x}, c) = p(\beta \mid \mathbf{z}) \propto p(\beta) \cdot p(\mathbf{z} \mid \beta).$$

We know that a normal prior for β gives a (multivariate) normal posterior for β (as likelihood is multivariate normal). In this case, we have:

$$\begin{aligned} E(\beta \mid \mathbf{z}) &= \mathbf{z}^T \mathbf{x} \frac{\tau_\beta^2}{\tau_\beta^2 \mathbf{x}^T \mathbf{x} + 1} \\ V(\beta \mid \mathbf{z}) &= \frac{\tau_\beta^2}{\tau_\beta^2 \mathbf{x}^T \mathbf{x} + 1}. \end{aligned}$$

$$p(\beta \mid \mathbf{z}) \sim MVN \left(\mathbf{z}^T \mathbf{x} \frac{\tau_\beta^2}{\tau_\beta^2 \mathbf{x}^T \mathbf{x} + 1}, \frac{\tau_\beta^2}{\tau_\beta^2 \mathbf{x}^T \mathbf{x} + 1} \right).$$

(b)

Since $c \sim N(0, \tau_c^2)$, by the definition of probit regression above, there exists a critical value c_k such that:

- c_k is greater than all z_i 's for which $y_i = 0$, and
- c_k is less than all z_i 's for which $y_i = 1$.

Assume $a_k = \max\{z_i : y_i = 0\}$, $b_k = \min\{z_i : y_i = 1\}$, then the full conditional distribution of c is proportional to $p(c)$ but constrained by such set $\{c : a_k < c < b_k\}$.

The conditional distribution of z_i is very similar. Given a value of c , different values of y_i indicates z_i lies in the interval of (a, b) . Actually, such (a, b) interval can be either $(-\infty, c)$ or (c, ∞) . Therefore, the full conditional distribution of z_i given $\mathbf{y}, \mathbf{x}, \mathbf{z}_{-i}, \beta, c$ can be represented by:

$$p(z_i \mid \mathbf{y}, \mathbf{x}, \mathbf{z}_{-i}, \beta, c) \propto p(z_i \mid \beta) \cdot \delta_{(a,b)}(z_i)$$

(c)

Since the posterior is a constrained normal, we can either write an if-statement to decide if the newly sampled parameter is within that range, or we can directly sample from *truncated normal distribution* with function `rtnorm()` from package `MCMCglmm`.

Also note that, in our Gibbs sampler, τ_β^2, τ_c^2 are fixed. We only update β, c, \mathbf{z} iteratively.

```
rm(list=ls())
data <- read.table("divorce.dat")
# head(data)
x <- data$V1
y <- data$V2
n <- nrow(data)

## prior
tau.b2 <- tau.c2 <- 16

## Gibbs sampler
sims <- 25000

## initial values
beta <- rnorm(1,0,sqrt(tau.b2))
c <- rnorm(1,0,sqrt(tau.c2))
z <- NULL
for(j in 1:n) {
  m <- beta*x[j]
  if(y[j]==1) {
    z <- c(z,rtnorm(1,m,1,lower=c))
  } else {
    z <- c(z,rtnorm(1,m,1,upper=c))
  }
}
z <- t(as.matrix(z))

## fixed parameters
kappa <- sum(x^2)+1/tau.b2
ind0 <- which(y==0)
ind1 <- which(y==1)
for(i in 1:sims) {
```

```

## update beta
mu <- sum(z[i,]*x)/kappa
beta <- c(beta,rnorm(1,mu,sqrt(1/kappa)))
## update c
a <- max(z[i,ind0])
b <- min(z[i,ind1])
c <- c(c,rtnorm(1,0,sqrt(tau.c2),lower=a,upper=b))
## update z
z.new <- NULL
for(j in 1:n) {
  m <- beta[i+1]*x[j]
  if(y[j]==1) {
    z.new <- c(z.new,rtnorm(1,m,1,lower=c[i+1]))
  } else {
    z.new <- c(z.new,rtnorm(1,m,1,upper=c[i+1]))
  }
}
z <- rbind(z,z.new)
}

```

The effective sample sizes of all unknown parameters are shown below. Since we have run the Gibbs sampler more than 30000 iterations, the effective sample sizes are all large enough (greater than 1000) in this case, which is pretty good.

```
effectiveSize(beta)
```

```
##      var1
## 1576.87
```

```
effectiveSize(c)
```

```
##      var1
## 1049.373
```

```
effectiveSize(z)
```

```
##      var1      var2      var3      var4      var5      var6      var7
## 5083.168 15835.254 16930.518 8465.288 15433.610 24414.289 3693.242
##      var8      var9      var10     var11     var12     var13     var14
## 2091.845 15618.965 5017.216 25695.933 4941.147 2980.816 4482.329
##      var15     var16     var17     var18     var19     var20     var21
## 3400.137 25234.369 4281.005 4495.017 14660.367 15666.253 4281.526
##      var22     var23     var24     var25
## 14904.676 3458.914 1771.039 9560.599
```

We remove the first element in stored `beta`, `c` and the first row in `z` so that `lag0` is removed and the scaling looks better.

Meanwhile, we randomly choose 2 z_i 's out of all 25 to plot their ACF as plotting all 25 would look like a mess.

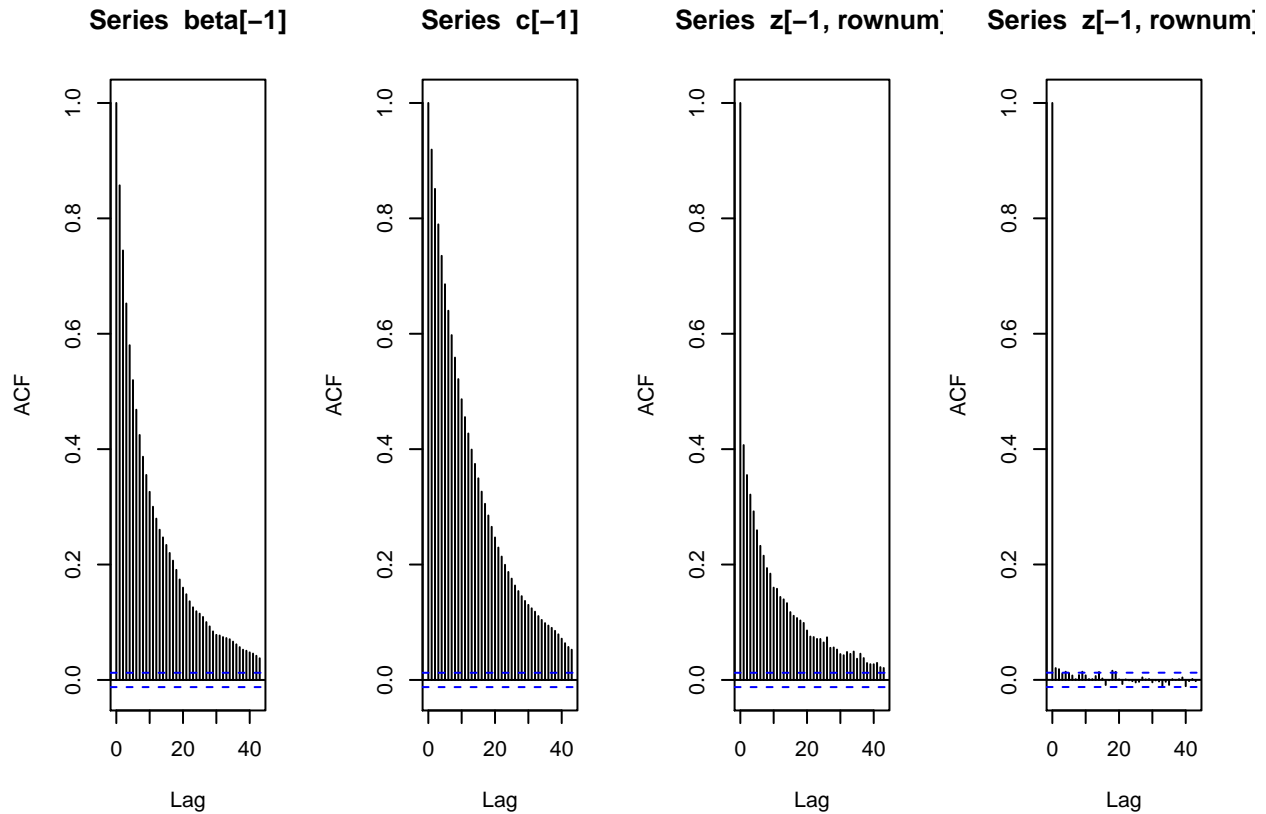
The autocorrelation function plots are plotted below:

```

# ACF
par(mfrow=c(1,4))
acf(beta[-1])
acf(c[-1])
for (rownum in sample(1:n,2,replace=F)) {

```

```
acf(z[-1,rownum])
}
```



ACF plots indicate that β and c are still not in a stable state while \mathbf{z} (in case this z_i and z_j) converges rather quickly. The number of iterations (25000) should be ok, then we intuitively guess that a larger sample size could probably help. This makes sense since the data of this study only contains 25 married couples.

(d)

```
# 95% CI for beta
quantile(beta[-1],c(0.025,0.975))

##          2.5%          97.5%
## 0.09884439 0.65455783

(prob <- length(which(beta[-1]>0))/sims)

## [1] 0.9989667
```

The 95% posterior confidence interval for β is (0.09763168, 0.65841390), and the probability of $\beta > 0$ is $0.9989667 \approx 1$ given \mathbf{y}, \mathbf{x} .

Problem 5

Suppose one observes positive values y_1, \dots, y_n that exhibit some right-skewness, Box and Cox (1964) suggested using the power transformation

$$w_i = \frac{y_i^\lambda - 1}{\lambda}, i = 1, \dots, n$$

such that w_1, \dots, w_n represent a random sample from a normal distribution with mean μ and standard deviation σ . Suppose that the vector of parameter $\theta = (\lambda, \mu, \sigma)$ is assigned the noninformative prior proportional to $1/\sigma$.

- (a) Derive the posterior density of θ up to a proportionality constant.
- (b) Suppose this transformation model is fit to the following survival times of patients in a cancer study:

```
y <- c(13, 52, 6, 40, 10, 7, 66, 10, 10, 14, 16, 4,
       65, 5, 11, 10, 15, 5, 76, 56, 88, 24, 51, 4,
       40, 8, 18, 5, 16, 50, 40, 1, 36, 5, 10, 91,
       18, 1, 18, 6, 1, 23, 15, 18, 12, 12, 17, 3)
```

Write an R function to compute the logarithm of the posterior distribution of $\theta = (\lambda, \mu, \sigma)$.

- (c) Laplace's method is a technique used to approximate integrals (using a Taylor series expansion of the function to integrate, followed by a multivariate normal approximation). The function `laplace` in the R package `LearnBayes` finds the joint posterior mode of the required distribution using the `optim` function and the Nelder-Mead numerical approximation algorithm.

Use the `laplace` function to find the posterior mode of $\theta = (\lambda, \mu, \sigma)$ using an initial starting value of (0.1, 3, 0.5).

- (d) Assume λ is fixed and known. Use a Gibbs sampling algorithm to obtain 10,000 posterior draws of μ and σ , assuming different values of λ . Construct 95% interval estimates of μ and σ , and discuss the sensitivity of your posterior inference to different values of λ .

Solution

(a)

The posterior density of θ is given, up to a proportionality constant, by

$$\begin{aligned} p(\theta \mid \mathbf{y}) &= p(\theta) \cdot p(\mathbf{y} \mid \theta) \cdot \frac{1}{p(\mathbf{y})} \\ &\propto p(\theta) \cdot p(\mathbf{w} \mid \mu, \sigma) \cdot \mathbf{w} \\ &= p(\theta) \cdot \prod_{i=1}^n p(w_i \mid \mu, \sigma) \cdot w_i \\ &\propto \frac{1}{\sigma} \prod_{i=1}^n \left[\phi\left(\frac{y_i^\lambda - 1}{\lambda}; \mu, \sigma\right) y_i^{\lambda-1} \right]. \end{aligned}$$

(b)

Note that we are interested in the logarithm of posterior distribution of θ , so we apply a log-transformation on the posterior distribution in part (a).

```
rm(list=ls())
y <- c(13, 52, 6, 40, 10, 7, 66, 10, 10, 14, 16, 4,
      65, 5, 11, 10, 15, 5, 76, 56, 88, 24, 51, 4,
      40, 8, 18, 5, 16, 50, 40, 1, 36, 5, 10, 91,
      18, 1, 18, 6, 1, 23, 15, 18, 12, 12, 17, 3)

# log posterior of THETA
LPT1 <- function(theta,y){
  lambda <- theta[1]
  mu <- theta[2]
  sigma <- theta[3]
  r <- sum(dnorm((y~lambda-1)/lambda,mu,sigma,log=T)+(lambda-1)*log(y))+
    log(1/sigma)
}
```

(c)

We directly use `laplace()` function from `LearnBayes` package with input parameter `mode = c(0.1,3,0.5)` as initial values.

```
(fit <- laplace(LPT1,mode=c(0.1,3,0.5),y))
```

```
## $mode
## [1] 0.1033001 3.0940929 1.4307691
##
## $var
##          [,1]      [,2]      [,3]
## [1,] 0.01271317 0.06540363 0.04671406
## [2,] 0.06540363 0.37912055 0.24031777
## [3,] 0.04671406 0.24031777 0.19253983
##
## $int
## [1] -201.9249
##
## $converge
## [1] TRUE
```

So the posterior modes for λ, μ, σ are 0.1033001, 3.0940929, 1.4307691 respectively.

(d)

As λ is fixed, we cannot use `gibbs()` function in `LearnBayes` directly. However, we can take a detour by “not updating” `lambda` in each iteration. The log posterior distribution is changed as well.

Here, we pick a sequence of `lambda` values from 0.01 to 1 (by 0.1).

```
# lambdas <- c(0.01,0.05,0.1,0.25,0.5,0.75,1)
lambdas <- seq(from=0.01,to=1,by=0.1)
mu.lower <- mu.upper <- sigma.lower <- sigma.upper <- NULL
for (i in 1:length(lambdas)) {
  lambda <- lambdas[i]

  # log-posterior of theta vector version 2
  LPT2 <- function(theta,y){
    mu <- theta[1]
```



```

sigma <- theta[2]
r <- sum(dnorm((y^lambda-1)/lambda,mu,sigma,log=T)+(lambda-1)*log(y))+
  log(1/sigma)
}

# redo laplace with fixed lambda input
fit2 <- laplace(LPT2,mode=c(3,0.5),y)
# redo Gibbs sampler
gbs <- gibbs(LPT2,start=fit2$mode,m=10000,scale = 2*diag(fit2$var)^0.5, y)
paras <- cbind(gbs$par[,1:2])
cis <- apply(paras,2,quantile,c(0.025, 0.975))
mu.lower <- c(mu.lower,cis[1,1])
mu.upper <- c(mu.upper,cis[2,1])
sigma.lower <- c(sigma.lower,cis[1,2])
sigma.upper <- c(sigma.upper,cis[2,2])
# apply(paras, 2, mean)
}

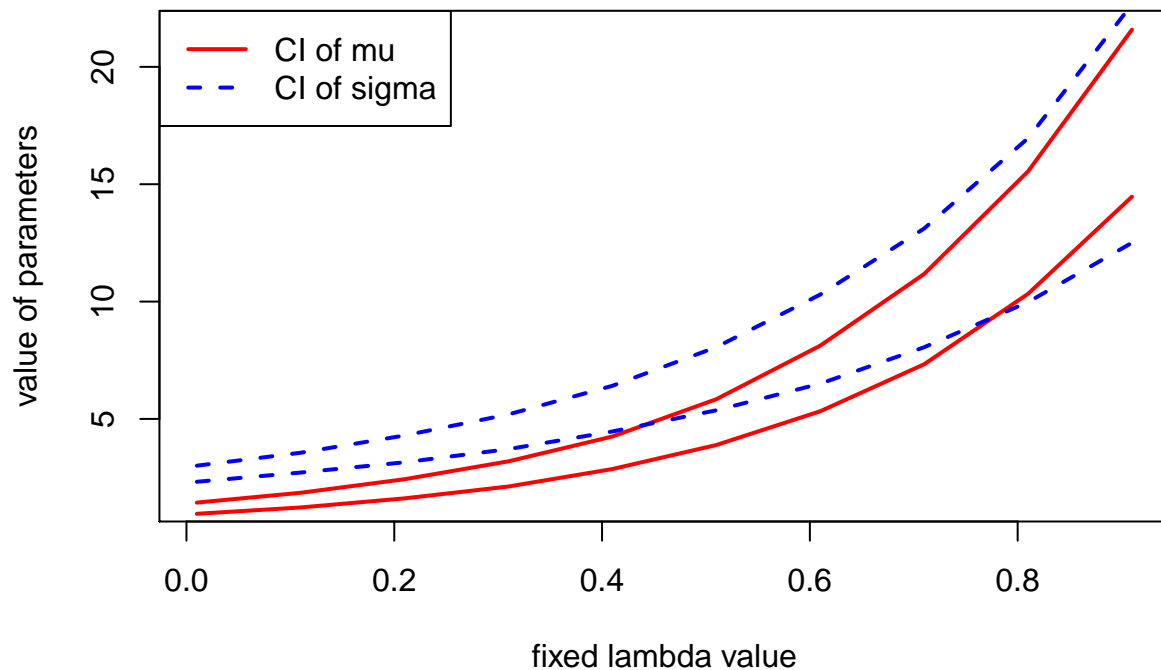
```

Finally, we plot all the collected information in a plot. The x-axis shows the fixed λ values we pick, the y-axis shows values of σ and μ . The two lines with the same colour and line type stand for the confidence interval for that parameter.

```

plot(lambdas,sigma.upper,type="l",lty=1,col="red",lwd=2,
     xlab="fixed lambda value",ylab="value of parameters")
lines(lambdas,sigma.lower,lty=1,col="red",lwd=2)
lines(lambdas,mu.upper,lty=2,col="blue",lwd=2)
lines(lambdas,mu.lower,lty=2,col="blue",lwd=2)
legend("topleft",legend=c("CI of mu","CI of sigma"),
      lty=c(1,2),col=c("red","blue"),lwd=c(2,2))

```



By observation, we notice:

- μ has lower values of confidence intervals but increases faster than σ .

- but the span of σ is larger than that of μ .

So we can conclude that as fixed λ gets larger, μ is more sensitive in the sense of **increasing**, while σ is more sensitive in the sense of **variations**.

References

- *Bayesian Data Analysis*, Third Edition (Chapman & Hall/CRC Texts in Statistical Science) (2014) by Andrew Gelman, John B. Carlin, Hal S. Stern, Donald B. Rubin
- *Bayesian Computation with R (Use R)* (2009) by Jim Albert.
- *A First Course in Bayesian Statistical Methods* by Peter D. Hoff, Chapter 12.1.1 Probit Regression.
- *Truncated normal distribution*, https://en.wikipedia.org/wiki/Truncated_normal_distribution.
- *Cross Validated - Omit 0 lag order in ACF plot*, <https://stats.stackexchange.com/questions/57573/omit-0-lag-order-in-acf-plot>