

CHL5223 A2

Belina Jang

Question 1

(a) Use R to run a Gibbs sampler algorithm, a Markov chain Monte Carlo algorithm, on this data. (DO NOT USE WinBugs, Openbugs, Jags or any of their relatives. The purpose is for you to do some basic programming.) You only need to run the MCMC for about 20,000 iterations for this example. Note, that for full credit, you need to submit your code as part of the homework.

```
library(dplyr)
library(kableExtra)
set.seed(0)

# load the data
q1data <- read.csv("brnbdy.csv")

# Y=log(brain weight)
Y <- log(q1data$brain)
# X=log(body weight)
X <- log(q1data$body)
# n=number of data points
n <- length(Y)

# given values
mu_alpha <- 0
mu_beta <- 0
tau_alpha <- 0.0001
tau_beta <- 0.0001
a_tau <- 0.0001
b_tau <- 0.0001
```

```

# gibbs sampler step

# set up
alpha_samples <- numeric(20000)
beta_samples <- numeric(20000)
tau_samples <- numeric(20000)

# initial values
alpha <- rnorm(1, mean = 0, sd = 1)
beta <- rnorm(1, mean = 0, sd = 1)
tau <- rgamma(1, shape = 1, rate = 1)

# Gibbs Sampler Loop
for (i in 1:20000) {
  # sample alpha
  tau_alpha_star <- n*tau + tau_alpha
  mu_alpha_star <- (tau*sum(Y-beta*X) + tau_alpha*mu_alpha) / tau_alpha_star
  # var = 1/prec
  alpha <- rnorm(1, mean=mu_alpha_star, sd=sqrt(1/tau_alpha_star))

  # sample beta
  tau_beta_star <- tau*sum(X^2) + tau_beta
  mu_beta_star <- (tau*sum(X*(Y-alpha)) + tau_beta*mu_beta) / tau_beta_star
  beta <- rnorm(1, mean=mu_beta_star, sd=sqrt(1/tau_beta_star))

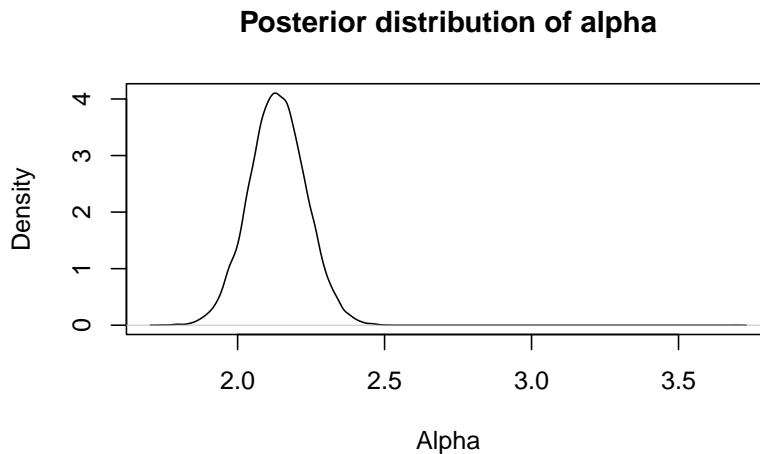
  # sample tau
  a_tau_star <- a_tau + n/2
  b_tau_star <- b_tau + 0.5*sum((Y-(alpha+beta*X))^2)
  tau <- rgamma(1, shape=a_tau_star, rate=b_tau_star)

  # save samples
  alpha_samples[i] <- alpha
  beta_samples[i] <- beta
  tau_samples[i] <- tau
}

```

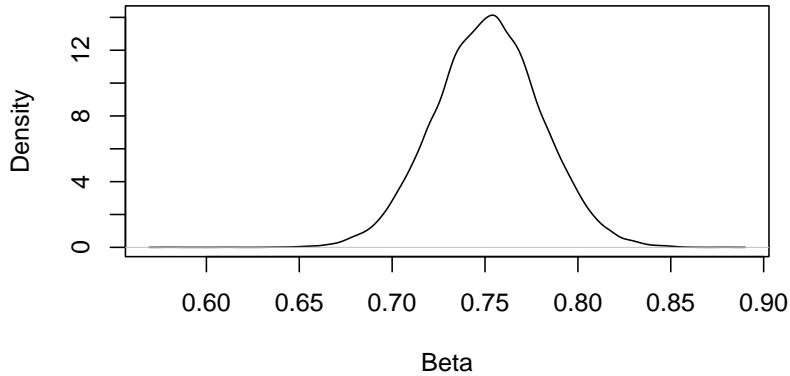
(b) Please give the posterior distribution (a plot and the summary statistics for the posterior distribution) for the parameters α , β , and τ . For the summary statistics of these posterior distributions it is sufficient to give the mean, standard deviation, and some type of 95% credible region.

```
set.seed(0)
# posterior distributions
plot(density(alpha_samples), main = "Posterior distribution of alpha", xlab =
  "Alpha")
```



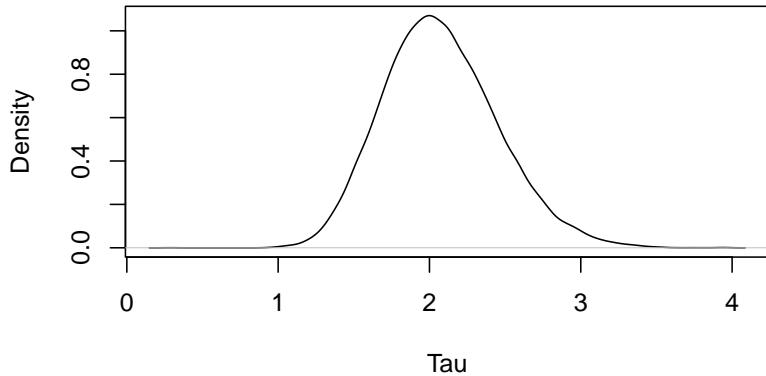
```
plot(density(beta_samples), main = "Posterior distribution of beta", xlab =
  "Beta")
```

Posterior distribution of beta



```
plot(density(tau_samples), main = "Posterior distribution of tau", xlab =
  ↪ "Tau")
```

Posterior distribution of tau



```
# posterior summaries
alpha_summary <- c(mean = signif(mean(alpha_samples),5), sd =
  ↪ signif(sd(alpha_samples),5),
  ↪ cr=paste0("(", signif(quantile(alpha_samples, 0.025),5),
  ↪ " ", ", signif(quantile(alpha_samples, 0.975),5), ")"))

beta_summary <- c(mean = signif(mean(beta_samples),5), sd =
  ↪ signif(sd(beta_samples),5),
```

Table 1: Summary statistics for the posterior distribution

	alpha	beta	tau
mean	2.1358	0.75151	2.0744
sd	0.097334	0.028763	0.37844
cr	(1.9469, 2.3274)	(0.69495, 0.8079)	(1.4038, 2.888)

```

cr=paste0("(, signif(quantile(beta_samples, 0.025),5), ",
           " , signif(quantile(beta_samples, 0.975),5), ")"))

tau_summary <- c(mean = signif(mean(tau_samples),5), sd =
  signif(sd(tau_samples),5),
  cr=paste0("(, signif(quantile(tau_samples, 0.025),5), ", " ,
            signif(quantile(tau_samples, 0.975),5), ")"))

posterior_summary <- data.frame(alpha = alpha_summary, beta = beta_summary,
  tau = tau_summary)

posterior_summary %>%
  kable(format="latex",
        caption = "Summary statistics for the posterior distribution")

```

(c) For an animal species with an average body weight of 55, provide the distribution which represents your belief of the brain weight. (Note, one might consider two different values here. Basically, I want the estimate of brain weight as a function of $\{\alpha, \beta, \text{body weight} = 55\}$). Since this is a function of the random variables α and β , the estimate is also a random variable.) Provide the distribution by giving an estimate of the density (a plot) and summary statistics of the distribution (mean, standard deviation, and credible interval).

```

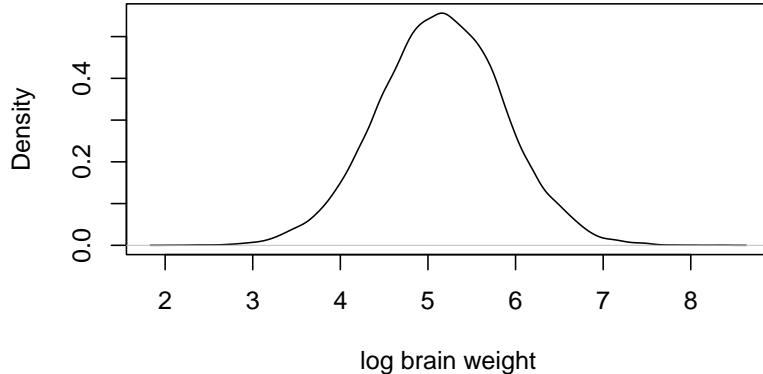
set.seed(0)

# predicted brain weight for X = log(body weight) = log(55)
new_X <- log(55)
pred_brain_log <- rnorm(length(alpha_samples), mean = alpha_samples +
  beta_samples * new_X, sd = 1/sqrt(tau_samples))
pred_brain <- exp(pred_brain_log)

plot(density(pred_brain_log), main = "Predicted log brain weight for body
  weight = 55", xlab = "log brain weight")

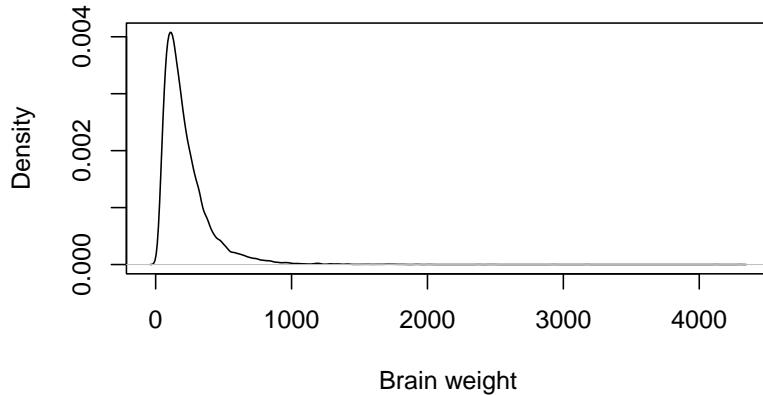
```

Predicted log brain weight for body weight = 55



```
plot(density(pred_brain), main = "Predicted brain weight for body weight =  
~ 55", xlab = "Brain weight")
```

Predicted brain weight for body weight = 55



```
# posterior summary  
brain_summary <- c(mean = signif(mean(pred_brain),5), sd =  
~ signif(sd(pred_brain),5),  
cr=paste0("(", signif(quantile(pred_brain, 0.025),5), ",  
~ ", signif(quantile(pred_brain, 0.975),5), ")"))
```

Table 2: Summary statistics for the predicted brain weight

	Brain weight
mean	222.71
sd	188.52
cr	(41.782, 702.87)

```
brain_summary %>%
  kable(format="latex", col.names = "Brain weight",
        caption = "Summary statistics for the predicted brain weight")
```

Question 2

(a) Do a bayesian Poisson regression on this data. Model the logarithm of the rate of deaths (per person-years) as a linear function of age and smoking category. Model both the smoking category and the age groupings as discrete variables (that is as categorical variables). Please give the OpenBugs/ Jags code for this data. Provide the univariate posterior distributions of each level of smoking and age and also give the posterior distribution of the precision used in your model. (If you wish, you may report the “standard deviations” instead of the precision.)

```
# Load the data
q2data <- read.csv("SmokeAgeDeath.csv")

# to get prior values
poisson_model <- glm(death ~ factor(age) + factor(smoke), offset =
  log(pyyears), family = poisson, data = q2data)

summary(poisson_model)
```

Call:
`glm(formula = death ~ factor(age) + factor(smoke), family = poisson,
 data = q2data, offset = log(pyyears))`

Coefficients:

```

Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.1949    0.2038 -45.111 < 2e-16 ***
factor(age)2  0.6431    0.2510   2.562  0.01040 *
factor(age)3  1.3469    0.2323   5.798  6.7e-09 ***
factor(age)4  2.0498    0.2257   9.082 < 2e-16 ***
factor(age)5  2.7697    0.2090  13.250 < 2e-16 ***
factor(smoke)2 -0.0740   0.1211  -0.611  0.54115
factor(smoke)3  0.5554    0.1774   3.130  0.00175 **
factor(smoke)4  1.4160    0.1401  10.104 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

(Dispersion parameter for poisson family taken to be 1)

```

Null deviance: 467.2046 on 19 degrees of freedom
Residual deviance: 4.7798 on 12 degrees of freedom
AIC: 109.56

```

Number of Fisher Scoring iterations: 4

Y_i : number of deaths in the age group i ($\text{age}=i$) E_i : person-years of exposure in the age group i N_i : number of people in the age group i λ_i : rate of deaths in the age group i

$$Y_i \sim \text{Poisson}(\lambda_i E_i)$$

Poisson regression model $\log(\lambda_i)$ (rate of deaths) as a linear function of age and smoking category:

$$\log(\lambda_i) = \alpha + \beta_1(\text{Age}) + \beta_2(\text{Smoking Category})$$

where $\alpha \sim N(-8, 20)$, $\beta_{\text{age}} \sim N(0, \tau_{\text{age}})$, $\beta_{\text{smoke}} \sim N(0, \tau_{\text{smoke}})$ and $\tau_{\text{age}} \sim \text{Gamma}(15, 1)$. and $\tau_{\text{smoke}} \sim \text{Gamma}(30, 1)$.

```

set.seed(0)

library(rjags)
library(tidyverse)

age_groups <- c("<45", "45-54", "55-64", "65-74", ">74")
smoking_categories <- c("Never", "Past", "<=20/day", ">20/day")

q2data$age <- factor(q2data$age)
q2data$smoke <- factor(q2data$smoke)

```

```

deaths <- q2data$death
person_years <- q2data$pyears
age_index <- q2data$age
smoking_index <- q2data$smoke

jags_model <- "
model {
  for (i in 1:N) {
    deaths[i] ~ dpois(lambda[i] * person_years[i])
    log(lambda[i]) <- alpha + beta_age[age_index[i]] +
      beta_smoke[smoking_index[i]]
  }

  # Priors
  alpha ~ dnorm(-8, 20)

  for (j in 1:5) {
    beta_age[j] ~ dnorm(0, tau_age)
  }

  # mean beta_age
  mean_beta_age <- sum(beta_age[]) / 4

  # beta_age adjusted
  for (j in 1:5) {
    beta_age_adj[j] <- beta_age[j] - mean_beta_age
  }

  for (k in 1:4) {
    beta_smoke[k] ~ dnorm(0, tau_smoke)
  }

  # mean beta_smoke
  mean_beta_smoke <- sum(beta_smoke[]) / 4

  # beta_smoke adjusted
  for (k in 1:4) {
    beta_smoke_adj[k] <- beta_smoke[k] - mean_beta_smoke
  }

  # alpha adjusted
  alpha_adj <- alpha + 5*mean_beta_age + 4*mean_beta_smoke
}

```

```

    tau_age ~ dgamma(15,1)
    tau_smoke ~ dgamma(30,1)
}
"

# data list
jags_data <- list(
  N = length(deaths),
  deaths = as.numeric(deaths),
  person_years = as.numeric(person_years),
  age_index = age_index,
  smoking_index = smoking_index
)

# initial list
initslist <- list(alpha = 0, beta_age = rep(0, 5), beta_smoke = rep(0, 4),
                     tau_age = 1, tau_smoke = 1)

# fit the model
model <- jags.model(textConnection(jags_model), data=jags_data, n.chains=3,
                     inits = initslist)

```

Compiling model graph
 Resolving undeclared variables
 Allocating nodes
 Graph information:
 Observed stochastic nodes: 20
 Unobserved stochastic nodes: 12
 Total graph size: 180

Initializing model

```

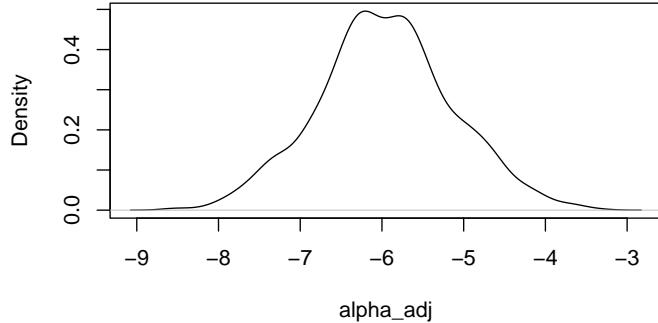
update(model, iter=5000)
samples <- coda.samples(model, variable.names = c("alpha_adj", "beta_age_adj",
  "beta_smoke_adj", "tau_age", "tau_smoke"), n.iter=10000, thin = 10)

# posterior distributions
# using first chain for density plot

plot(density(samples[[1]][,"alpha_adj"]), main="Posterior of alpha_adj",
  xlab="alpha_adj")

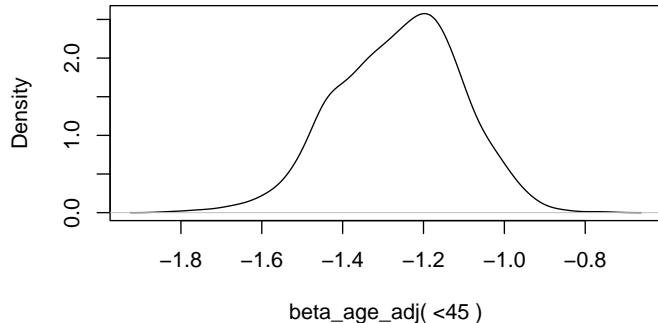
```

Posterior of alpha_adj

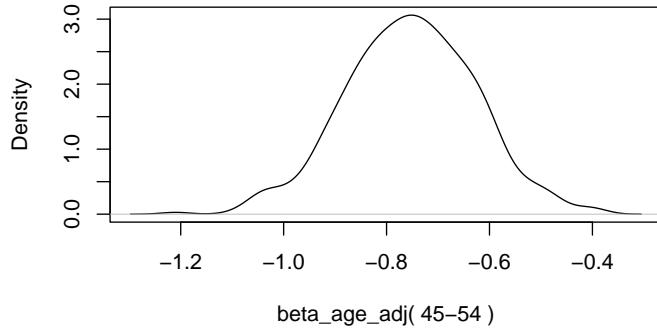


```
for (i in 1:5) {  
  plot(density(samples[[1]][, paste0("beta_age_adj[", i, "]")]), main =  
    paste("Posterior of beta_age_adj(", age_groups[i], ")"), xlab =  
    paste("beta_age_adj(", age_groups[i], ")"))  
}
```

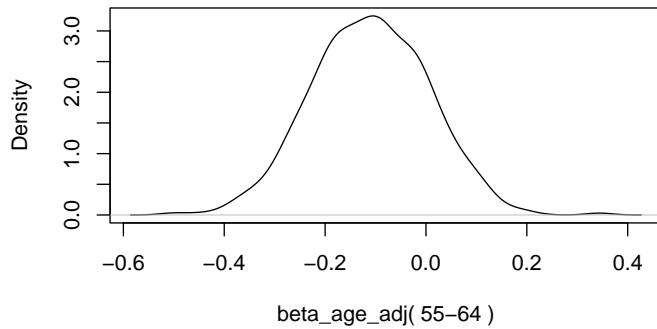
Posterior of beta_age_adj(<45)



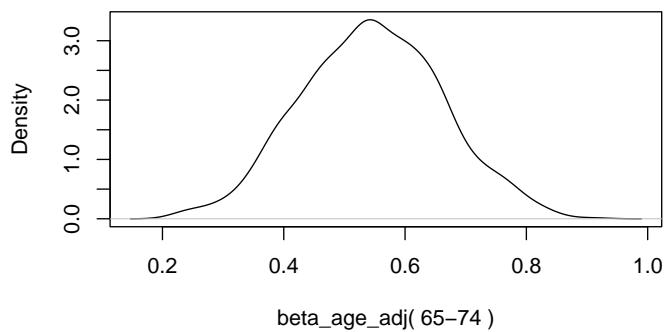
Posterior of $\text{beta_age_adj(45–54)}$



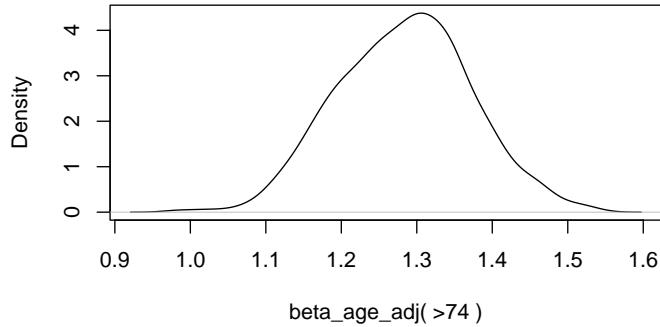
Posterior of $\text{beta_age_adj(55–64)}$



Posterior of $\text{beta_age_adj(65–74)}$

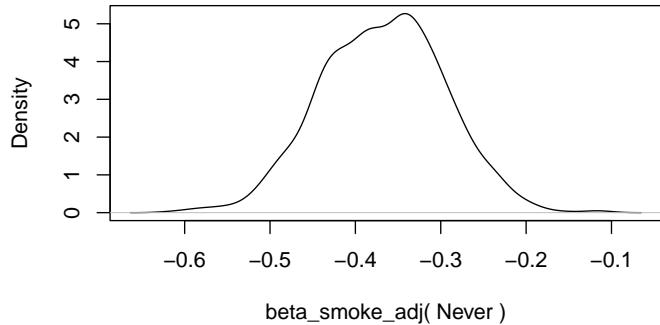


Posterior of beta_age_adj(>74)

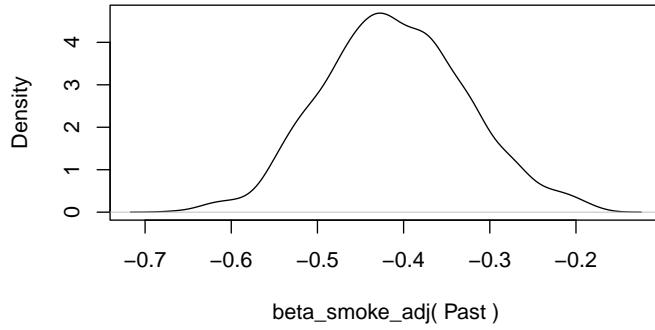


```
for (j in 1:4) {  
  plot(density(samples[[1]][, paste0("beta_smoke_adj[", j, "]")]), main =  
    paste("Posterior of beta_smoke_adj(", smoking_categories[j], ")"), xlab  
    = paste("beta_smoke_adj(", smoking_categories[j], ")"))  
}
```

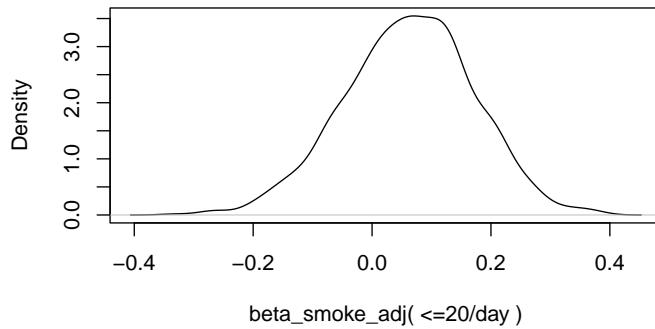
Posterior of beta_smoke_adj(Never)



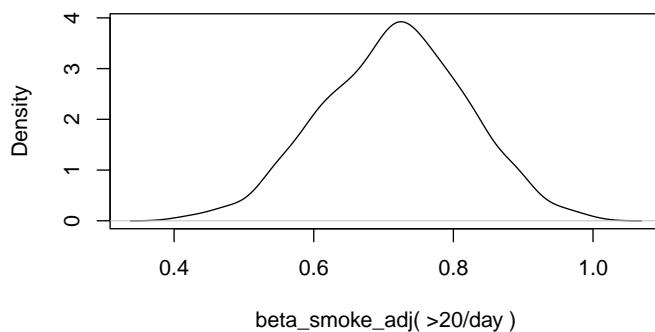
Posterior of $\beta_{\text{smoke_adj}}(\text{Past})$



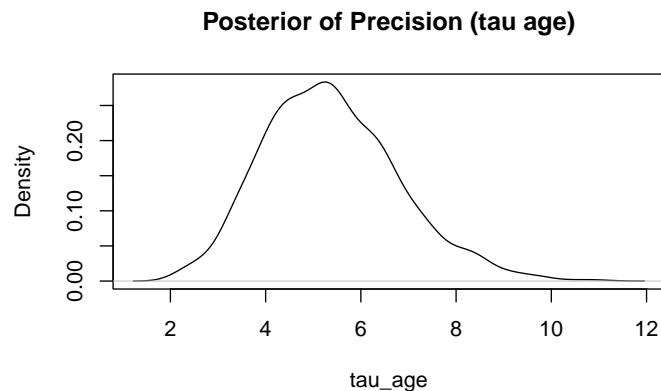
Posterior of $\beta_{\text{smoke_adj}}(\leq 20/\text{day})$



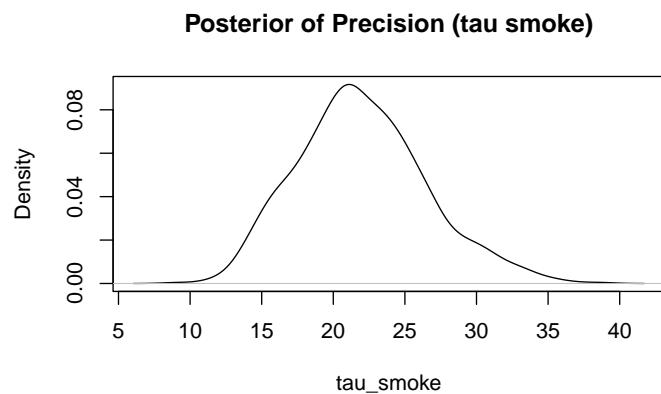
Posterior of $\beta_{\text{smoke_adj}}(> 20/\text{day})$



```
# posterior for precision (tau)
plot(density(samples[[1]][,"tau_age"]), main="Posterior of Precision (tau
  ↵ age)", xlab="tau_age")
```

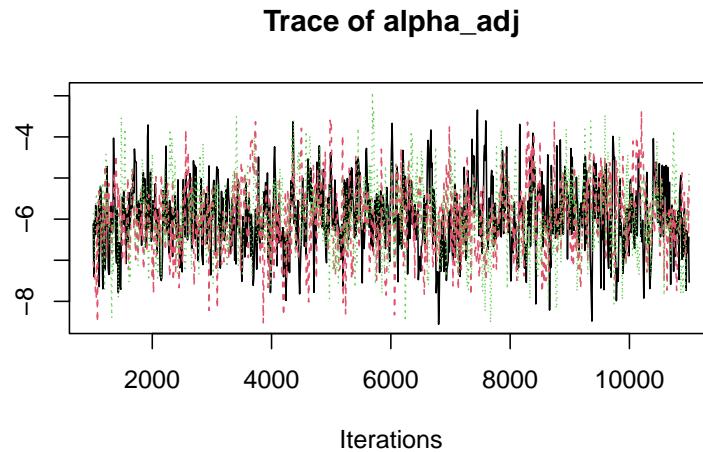


```
plot(density(samples[[1]][,"tau_smoke"]), main="Posterior of Precision (tau
  ↵ smoke)", xlab="tau_smoke")
```

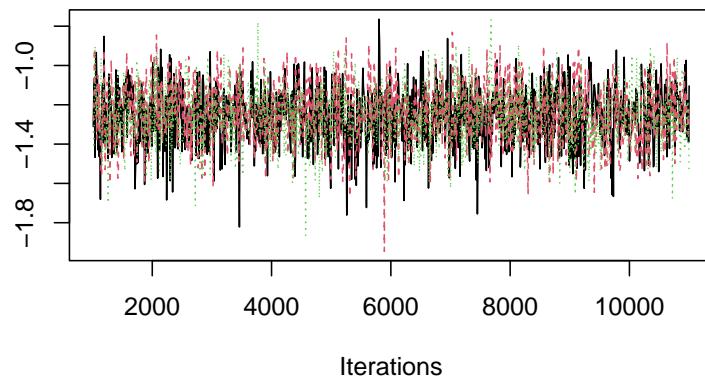


(b) Provide a brief justification that your model has converged and that you have “burnedin” your simulation enough. (Note: you may give “thinned” plots of your simulations to avoid printing out plots which are suppose to represent several thousands or hundreds of thousand points. For the purpose of this homework assignment question, you can provide simple diagnostics like trace/history plots and autocorrelation plots.) Also, for the purpose of this homework, just provide these diagnostics for only 3 parameters in the model.

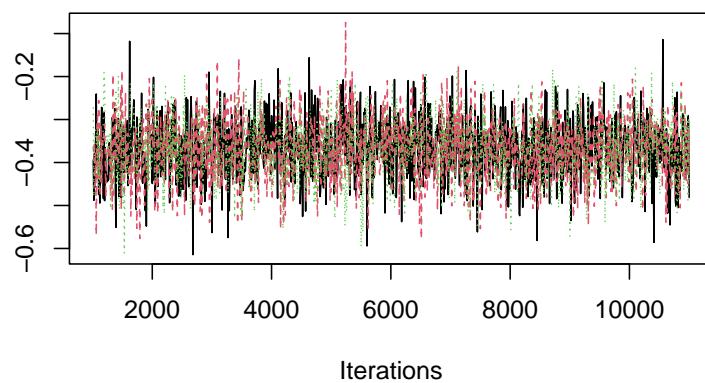
```
set.seed(0)
# convergence checking
# trace plot
traceplot(samples[, c("alpha_adj", "beta_age_adj[1]", "beta_smoke_adj[1]",
                     "tau_smoke")])
```



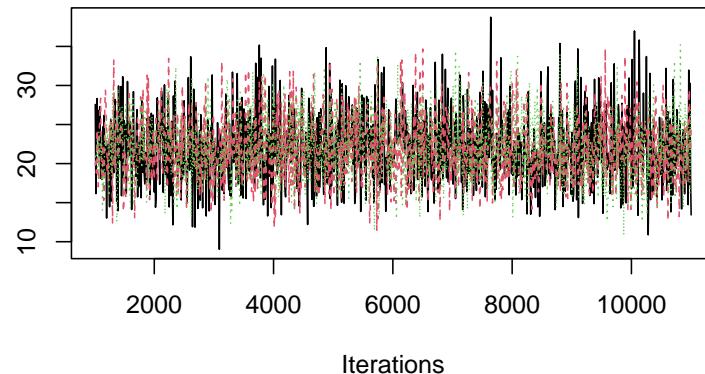
Trace of beta_age_adj[1]



Trace of beta_smoke_adj[1]

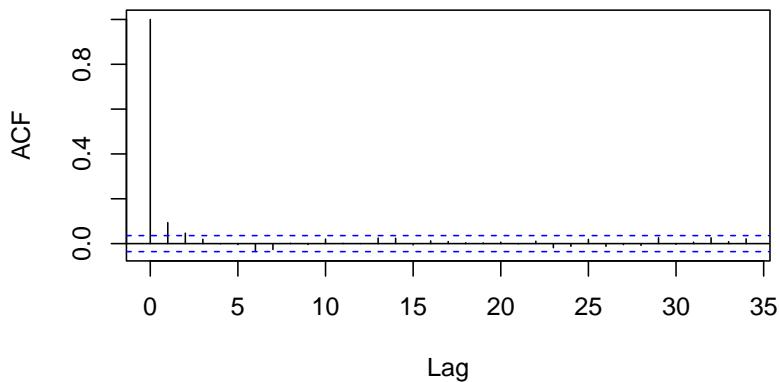


Trace of tau_smoke



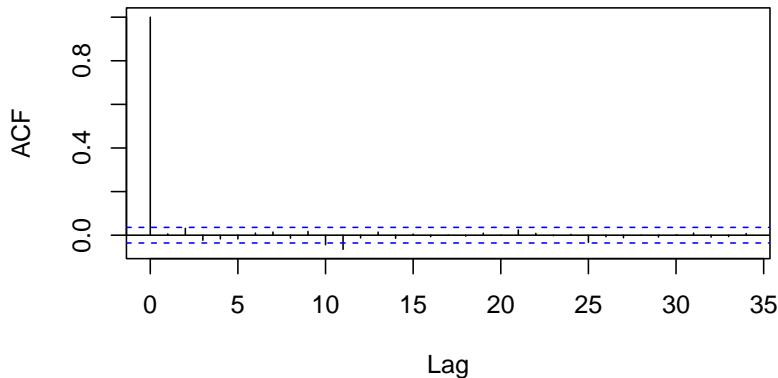
```
# acf  
acf(as.matrix(samples)[, "beta_age_adj[1]"], main="ACF of beta_age_adj(<45)")
```

ACF of beta_age_adj(<45)



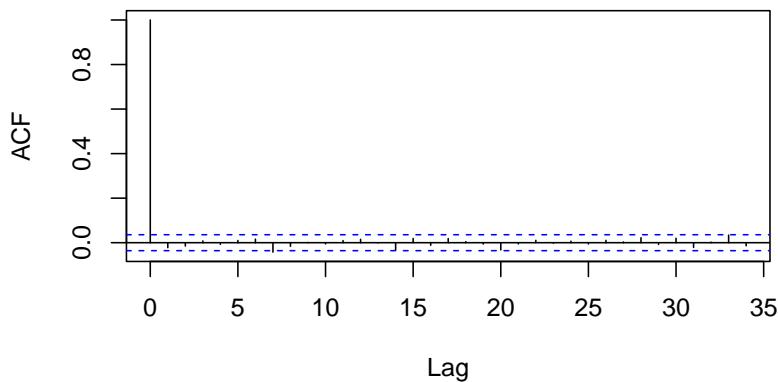
```
acf(as.matrix(samples)[, "beta_smoke_adj[1]"], main="ACF of  
← beta_smoke_adj(Never)")
```

ACF of beta_smoke_adj(Never)



```
acf(as.matrix(samples)[, "tau_smoke"], main="ACF of tau age")
```

ACF of tau age



For all three parameters, the trace plots show that the chains have converged since they are mixed in well with no obvious trends, and the autocorrelation plots also show that all the autocorrelation reached zero quickly (within 5-10 lags).

Table 3: Summary statistics for the risk ratio of heavy smokers vs. non-smokers

	Risk Ratio
mean	2.9871
sd	0.42773
cr	(2.2207, 3.9315)

(c) Provide your belief as to the increase probability of death for someone who smokes >20 cigarettes per day versus a nonsmoker. (That is, you would say that you believe that a smoker who smokes > 20 cigarettes per day has an increase risk of xxx times the risk of a nonsmoker.) Please, include both a “point” estimate of this increase risk and also state some interval estimate. Please identify the type of point and interval estimate that you are using.

```
set.seed(0)

library(kableExtra)
# risk ratio of heavy smokers vs. non-smokers using first chain
beta_smoke_samples <- samples[[1]][,"beta_smoke_adj[4]"] -
  ↪ samples[[1]][,"beta_smoke_adj[1]"]
risk_ratio <- exp(beta_smoke_samples)

risk_summary <- c(mean=signif(mean(risk_ratio),5),
  ↪ sd=signif(sd(risk_ratio),5),
  ↪ cr=paste0("(", signif(quantile(risk_ratio, 0.025),5), ", ",
  ↪ signif(quantile(risk_ratio, 0.975),5), ")"))

risk_summary %>%
  kable(format="latex", col.names = "Risk Ratio",
    caption = "Summary statistics for the risk ratio of heavy smokers vs.
  ↪ non-smokers")
```

Based on the bayesuan poisson regression model, a smoker who smokes more than 20 cigarettes per day has an increased risk of death by approximately 3 times the risk of a nonsmoker (posterior mean and the 95% credible interval shown in Table 3; didn't report the exact number in the paragraph since jags produce slightly different number everytime).

Question 3

(a)

For the i th rat, the weight at week j is

$$Y_{ij} = \beta_{0i} + \beta_{1i} * \text{week}_j + \epsilon_{ij},$$

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

Linear regression model of the rat's dose level using hierarchical model:

$$\beta_{0i} \sim N(\delta_{00} + \delta_{01} \cdot \text{dose}_i, \sigma_0^2)$$

$$\beta_{1i} \sim N(\delta_{10} + \delta_{11} \cdot \text{dose}_i, \sigma_1^2)$$

Priors for the group means:

$$\delta_{00} \sim N(100, 0.00001), \quad \delta_{10} \sim N(0, 0.0001), \quad \delta_{01} \sim N(0, 0.0001), \quad \delta_{11} \sim N(0, 0.0001)$$

Note:

- $\tau = \frac{1}{\sigma^2}$
- $\tau_{\beta_0} = \frac{1}{\sigma_0^2}$
- $\tau_{\beta_1} = \frac{1}{\sigma_1^2}$

```
set.seed(0)

data <- read.csv("ratDoseData.csv", header=TRUE)

dose <- data$doses

df_long <- data %>% pivot_longer(cols=wk1:wk11, names_to="week",
  values_to="Y") %>%
  mutate(week = as.numeric(gsub("wk", "", week)))

N <- length(unique(df_long$rat))
J <- max(df_long$week)

jags_model <- "
model {
  for (i in 1:N) {
```

```

for (j in 1:J) {
  # i: regression line
  Y[i, j] ~ dnorm(mu[i, j], tau)
  mu[i, j] <- beta0[i] + beta1[i] * j
}

# ii: linear regression model of the rat's dose level.
beta0[i] ~ dnorm(delta00 + delta01 * dose[i], tau_beta0)
beta1[i] ~ dnorm(delta10 + delta11 * dose[i], tau_beta1)
}

# iii: weak priors; not restrictive
delta00 ~ dnorm(100, 0.00001)
delta01 ~ dnorm(0, 0.0001)
delta10 ~ dnorm(0, 0.0001)
delta11 ~ dnorm(0, 0.0001)

# not conjugate; flat priors
sau ~ dunif(0, 250)
sigma_beta0 ~ dunif(0, 250)
sigma_beta1 ~ dunif(0, 250)
tau <- pow(sau, -2)
tau_beta0 <- pow(sigma_beta0, -2)
tau_beta1 <- pow(sigma_beta1, -2)
}

"
jags_data <- list(
  N = N,
  J = J,
  Y = matrix(df_long$Y, nrow=N, byrow=TRUE),
  dose = df_long$dose
)

model <- jags.model(textConnection(jags_model), data=jags_data, n.chains=3)

```

```

Compiling model graph
Resolving undeclared variables
Allocating nodes
Graph information:
  Observed stochastic nodes: 110
  Unobserved stochastic nodes: 27
  Total graph size: 1373

```

Initializing model

```
# iv.individual rate model only report for rat 1: beta01, beta11,
# and for the variance parameter parameter sigma^2=1/tau
# for the y-intercept parameter model for the parameters delta00, delta01,
# and sigma0^2=1/tau_beta0
# for the dose rate model delta10, delta11, and sigma1^2=1/tau_beta1

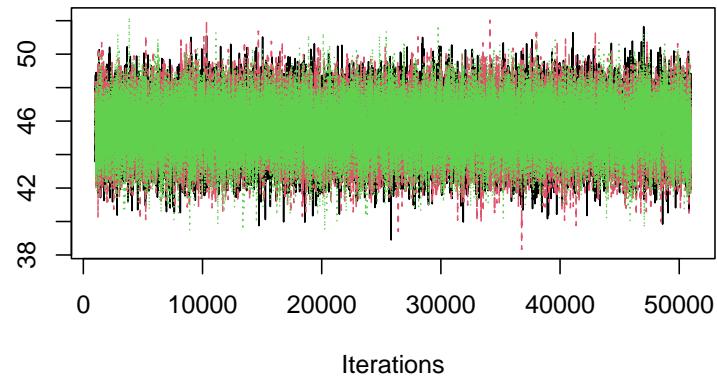
samples <- coda.samples(model,
                         variable.names=c("beta0[1]", "beta1[1]", "tau",
                                           "delta00", "delta01", "tau_beta0",
                                           "delta10", "delta11", "tau_beta1"),
                         n.iter=50000, n.burnin=10000)
```

(b) Provide some preliminary evidence that the model looks like it converged. It is sufficient to show trace plots and some auto correlation values. (Don't do this for all individual rat's parameters. (See the note above.)

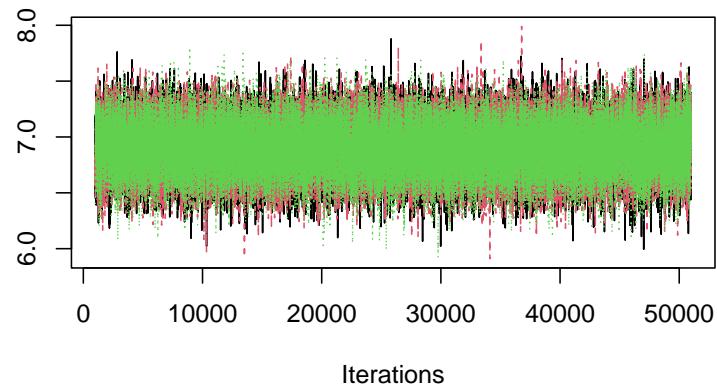
```
# create sigma squares
#as.matrix(samples)[,"sigma2"] <- 1 / as.matrix(samples)[,"tau"]
#as.matrix(samples)[,"sigma2_0"] <- 1 / as.matrix(samples)[,"tau_beta0"]
#as.matrix(samples)[,"sigma2_1"] <- 1 / as.matrix(samples)[,"tau_beta1"]

# trace plot
traceplot(samples[, c("beta0[1]", "beta1[1]", "tau", "delta00", "delta01",
                     "tau_beta0", "delta10", "delta11", "tau_beta1")])
```

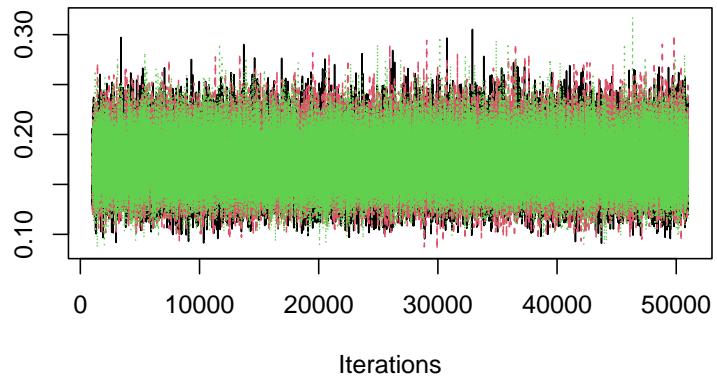
Trace of beta0[1]



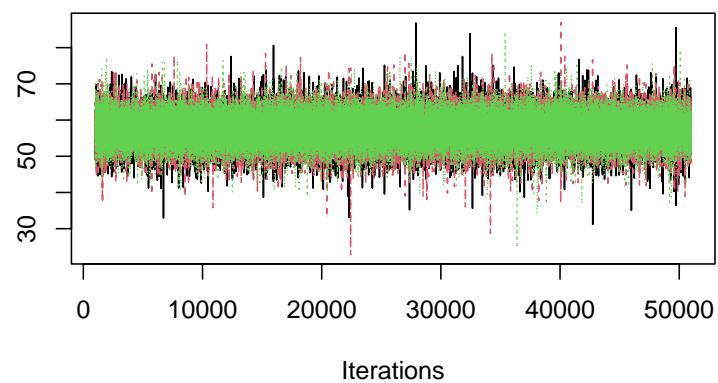
Trace of beta1[1]



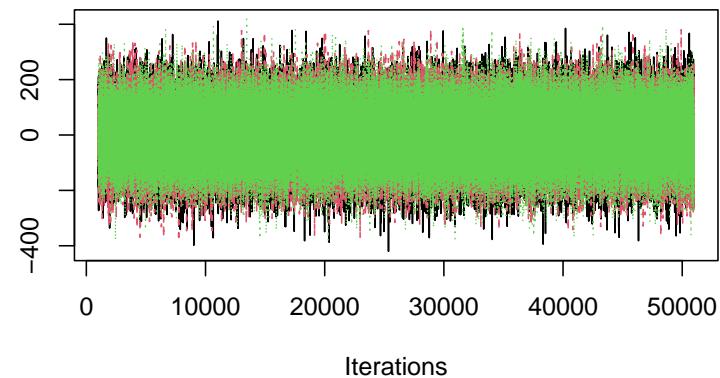
Trace of tau



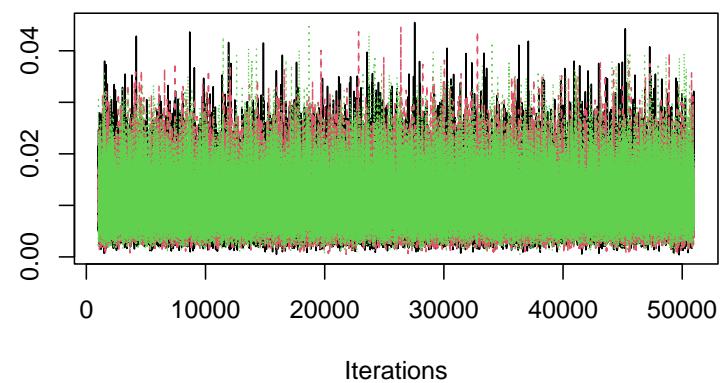
Trace of delta00



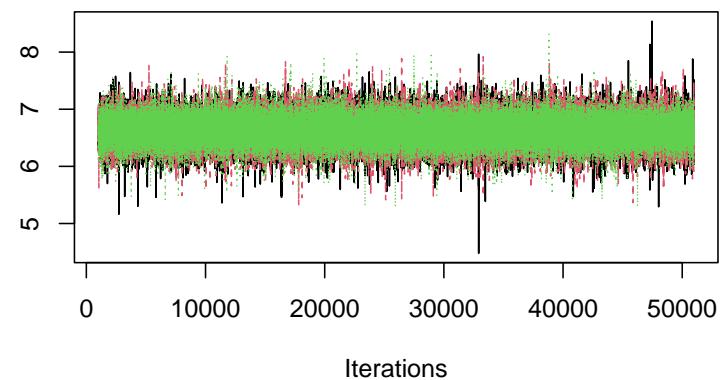
Trace of delta01



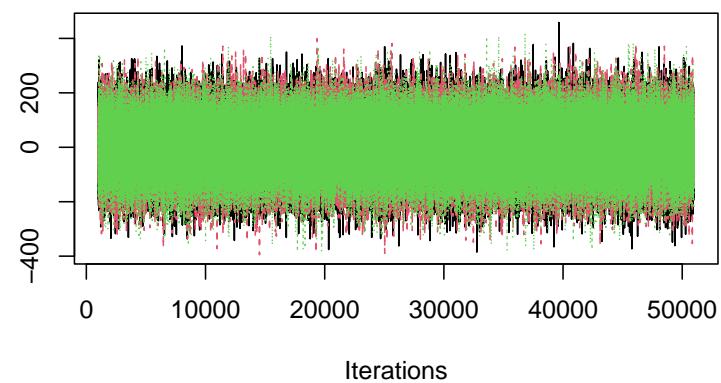
Trace of tau_beta0



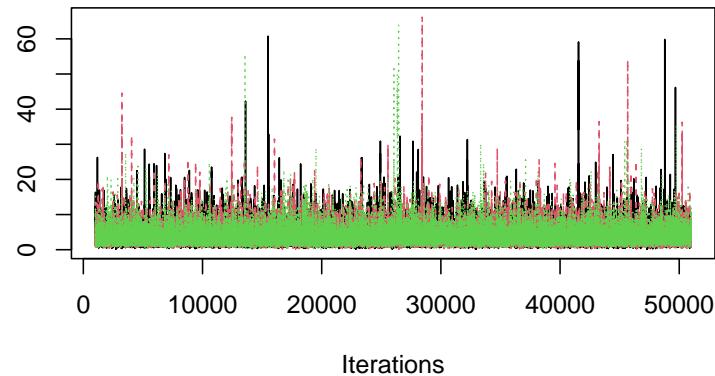
Trace of delta10



Trace of delta11

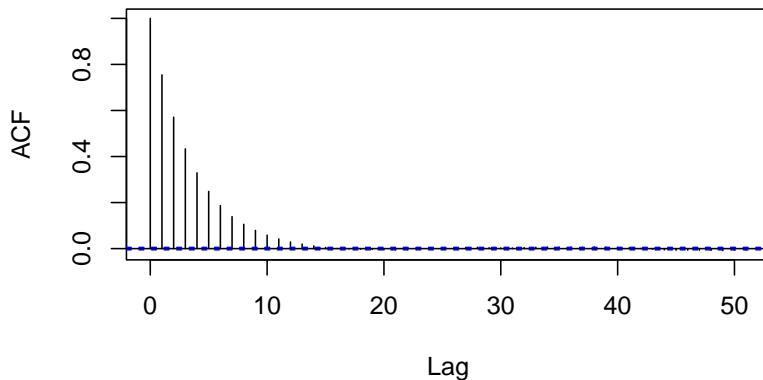


Trace of tau_beta1



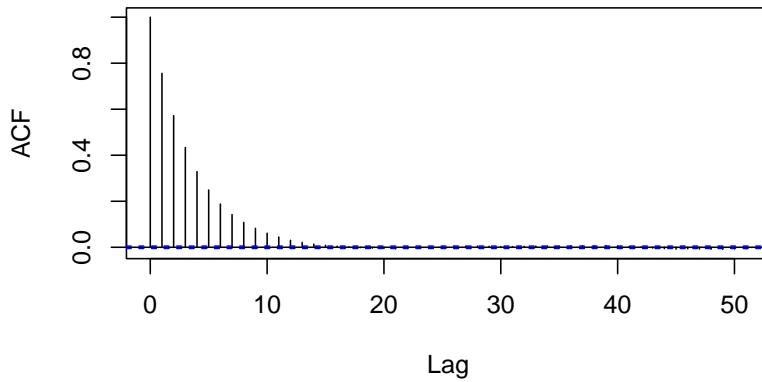
```
# acf: autocorrelation  
  
# for rat 1  
acf(as.matrix(samples)[, "beta0[1]"], main="ACF of beta0[1]")
```

ACF of beta0[1]



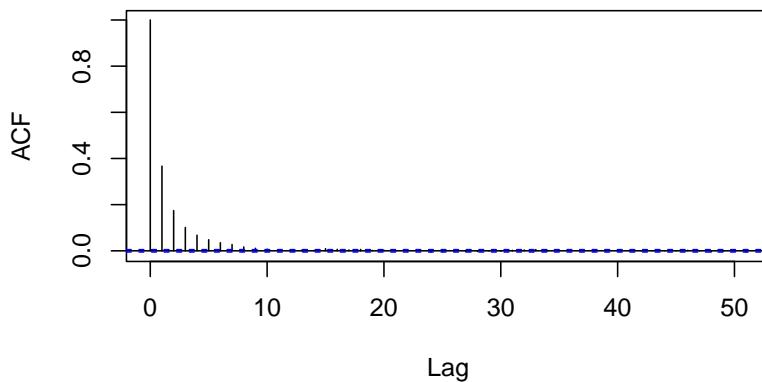
```
acf(as.matrix(samples)[, "beta1[1]"], main="ACF of beta1[1]")
```

ACF of beta1[1]



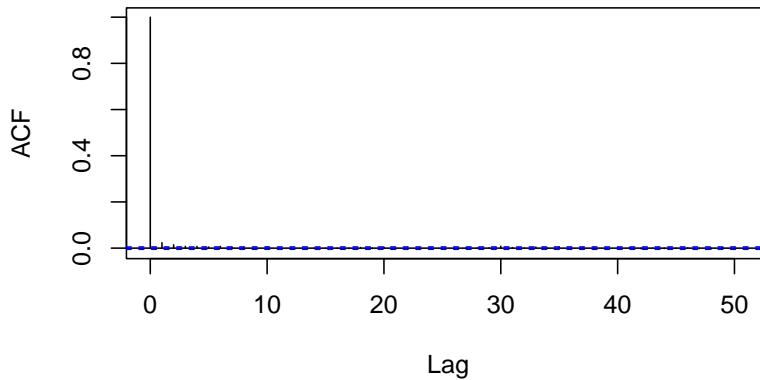
```
acf(as.matrix(samples)[, "tau"], main="ACF of tau")
```

ACF of tau



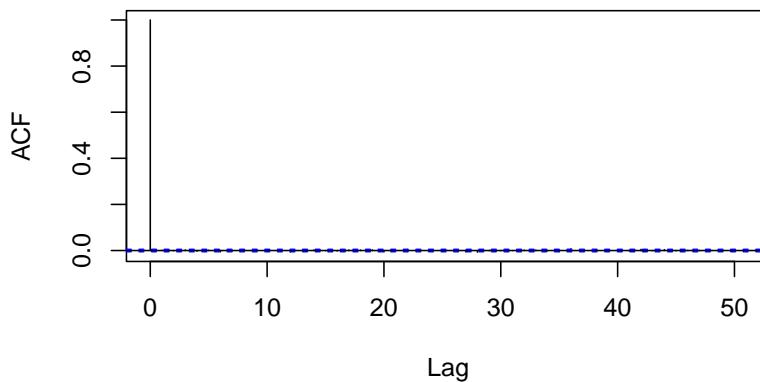
```
# for the y-intercept parameter  
acf(as.matrix(samples)[, "delta00"], main="ACF of delta00")
```

ACF of delta00

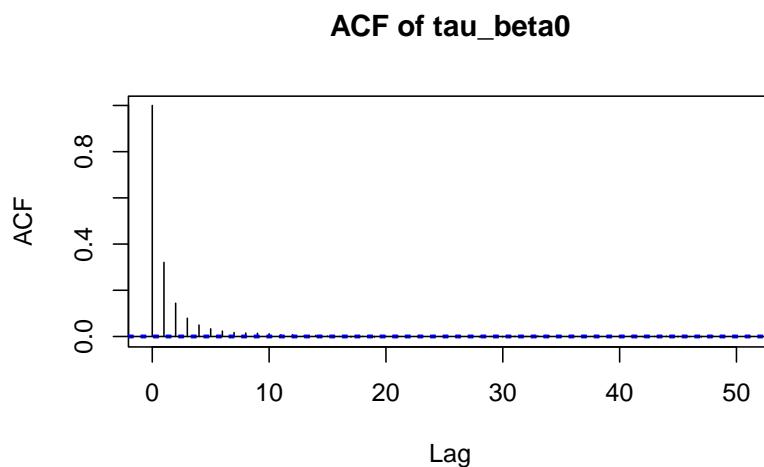


```
acf(as.matrix(samples)[, "delta01"], main="ACF of delta01")
```

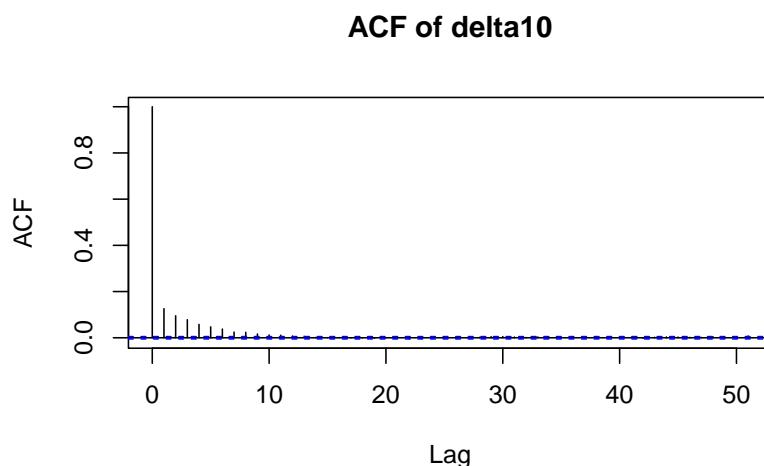
ACF of delta01



```
acf(as.matrix(samples)[, "tau_beta0"], main="ACF of tau_beta0")
```

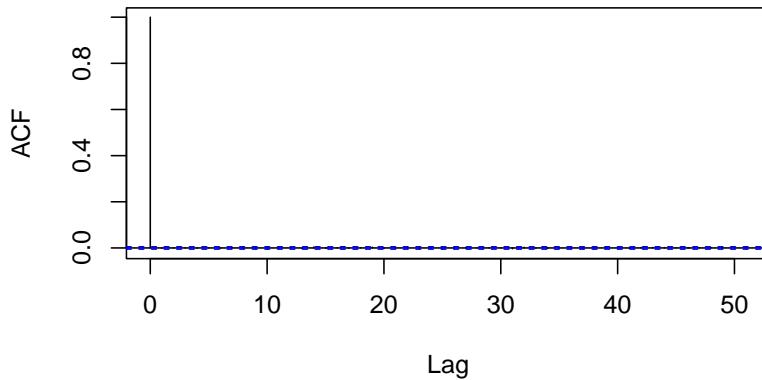


```
# for the dose rate parameter  
acf(as.matrix(samples)[, "delta10"], main="ACF of delta10")
```



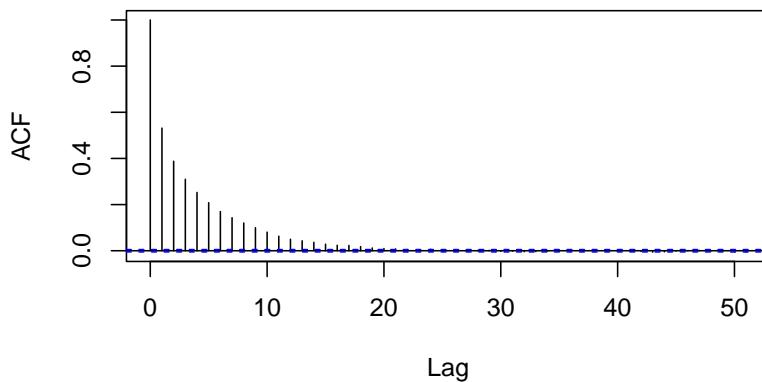
```
acf(as.matrix(samples)[, "delta11"], main="ACF of delta11")
```

ACF of delta11



```
acf(as.matrix(samples)[, "tau_beta1"], main="ACF of tau_beta1")
```

ACF of tau_beta1

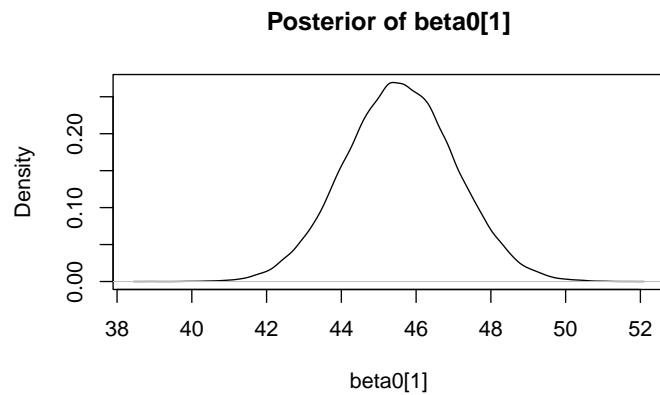


Based on the trace plots, the chains have converged since they are mixed in well with no obvious trends. The autocorrelation plots also show that all the autocorrelation reached zero quickly (within 20 lags).

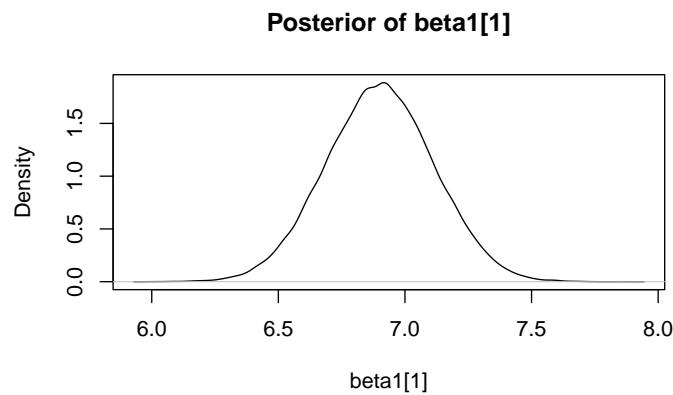
(c) Provide the posterior distribution for the parameters of the distribution of the 's. (See the comment above about which parameters to pick.)

```
# using first chain for density plot
sigma_sq <- 1 / samples[[1]][,"tau"]
sigma_beta0_sq <- 1 / samples[[1]][,"tau_beta0"]
sigma_beta1_sq <- 1 / samples[[1]][,"tau_beta1"]

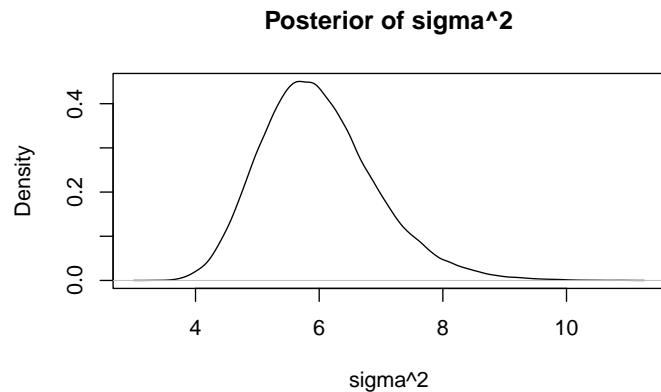
# individual rat model
plot(density(samples[[1]][,"beta0[1]"]), main="Posterior of beta0[1]",
      xlab="beta0[1]")
```



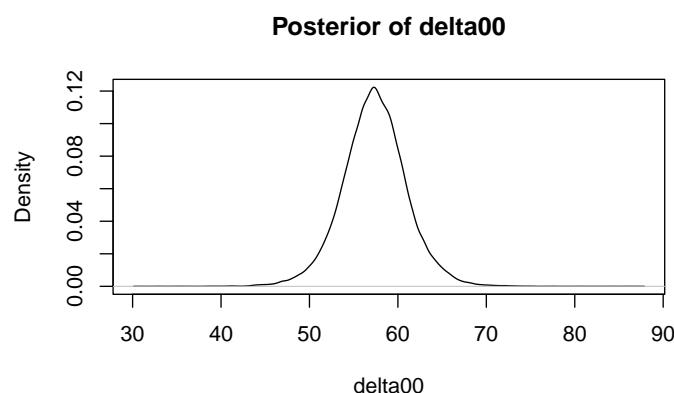
```
plot(density(samples[[1]][,"beta1[1]"]), main="Posterior of beta1[1]",
      xlab="beta1[1]")
```



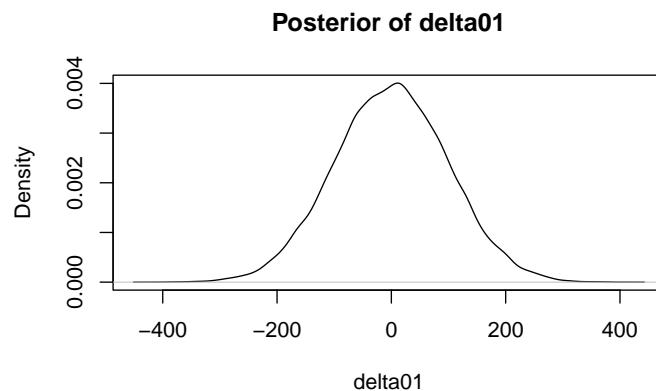
```
plot(density(sigma_sq), main="Posterior of sigma^2", xlab="sigma^2")
```



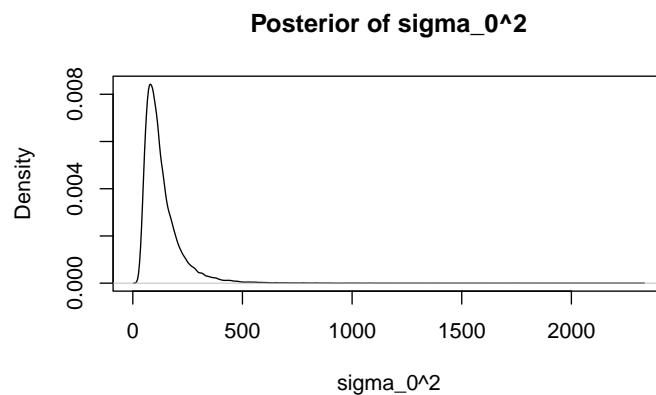
```
# parameters of the distribution of beta01  
plot(density(samples[[1]][,"delta00"]), main="Posterior of delta00",  
      xlab="delta00")
```



```
plot(density(samples[[1]][,"delta01"]), main="Posterior of delta01",  
      xlab="delta01")
```

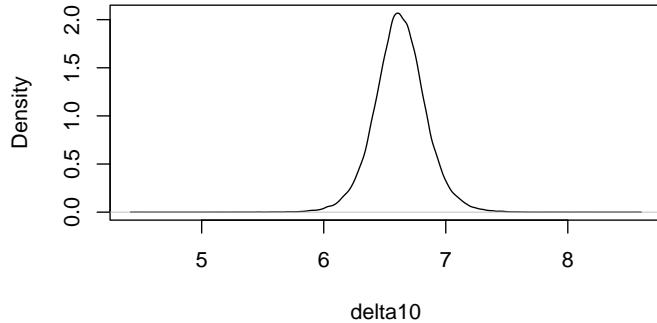


```
#plot(density(samples[[1]][,"tau_beta0"]), main="Posterior of Precision
← (tau_beta0)", xlab="tau_beta0")
plot(density(sigma_beta0_sq), main="Posterior of sigma_0^2",
← xlab="sigma_0^2")
```



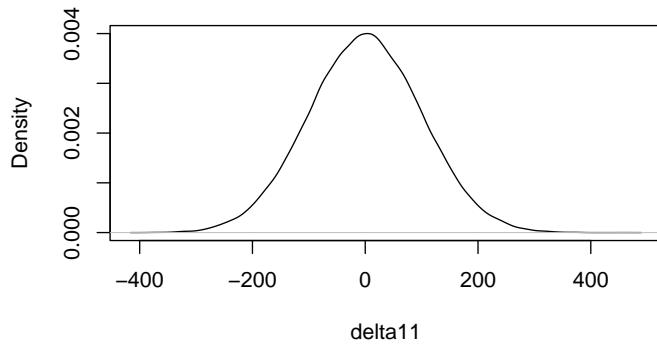
```
# parameters of the distribution of beta11
plot(density(samples[[1]][,"delta10"]), main="Posterior of delta10",
← xlab="delta10")
```

Posterior of delta10

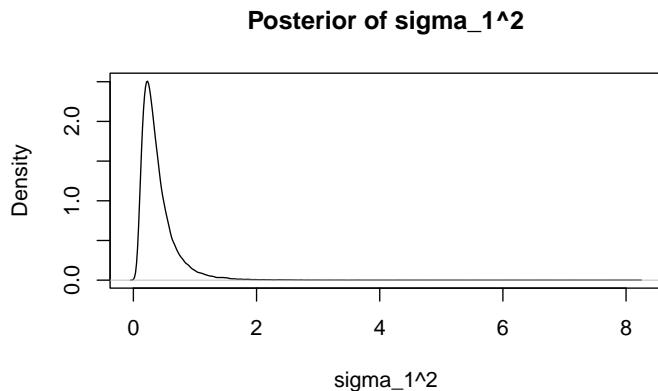


```
plot(density(samples[[1]][,"delta11"]), main="Posterior of delta11",
      xlab="delta11")
```

Posterior of delta11



```
#plot(density(samples[[1]][,"tau_beta1"]), main="Posterior of Precision
      (tau_beta1)", xlab="tau_beta1")
plot(density(sigma_beta1_sq), main="Posterior of sigma_1^2",
      xlab="sigma_1^2")
```



(d) Is there a difference due to drug in this model. That is using the regression models described in step (a.ii) to make inferences about the effect of the drug levels on the y-intercept and the growth rate of the rats. Please provide your posterior belief of this effect. In justifying your answer, you should include the appropriate posterior distribution.

From step (a.ii), we know

$$\beta_{0i} \sim N(\delta_{00} + \delta_{01} \cdot \text{dose}_i, \sigma_0^2)$$

$$\beta_{1i} \sim N(\delta_{10} + \delta_{11} \cdot \text{dose}_i, \sigma_1^2)$$

If δ_{00} and δ_{10} are significantly different from 0, then there is a difference due to drug in the model.

```
# using first chain
summary(samples[[1]][,"delta01"])
```

```
Iterations = 1001:51000
Thinning interval = 1
Number of chains = 1
Sample size per chain = 50000
```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

Mean	SD	Naive SE	Time-series SE
-0.3091	99.7892	0.4463	0.4463

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
-194.9645	-67.5265	0.1769	67.1087	195.9975	

```
summary(samples[[1]][,"delta11"])
```

```
Iterations = 1001:51000
Thinning interval = 1
Number of chains = 1
Sample size per chain = 50000
```

1. Empirical mean and standard deviation for each variable,
plus standard error of the mean:

Mean	SD	Naive SE	Time-series SE
0.1238	100.0373	0.4474	0.4474

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
-195.4093	-67.6138	0.1544	67.7466	196.2742	

Since the 95% credible intervals for δ_{01} and δ_{11} contain 0 and mean values are not significantly different from 0, we can conclude that there is little to no evidence of difference due to drug in the model.

(e) In a short paragraph, summarize your belief in the effect of the different dose levels of the drug. In this summary, you should state what the strength of the evidence between the two different dose levels. You should describe this difference using values from the posterior distribution. (Note, for previous answer, you should concentrate on the “mathematical” values. For this answer, you should explain things in “English” so that your non-statistical colleague would understand the answer.)

In this analysis, we wanted to study whether giving different amount of the drug will change how fast they grow by measuring their weights over time. While we did observe some differences between the weights of rats with different dose levels, the difference was small and not found to be significant. This means that the difference could be due to a real effect of different level of doses or just a random error. Our study result suggests that there is not enough statistical evidence that different dose of drug has a significant effect on how fast they grow.