Dr Byron J. Gajewski
BIOS 902

Guanlin Zhang
Fall '18

Question 1: Chapter 5 Question 3 on page 134:

Solution 1. For part (a), *We first reproduce all the graphs of the example as in Section 5.5.*

To get Figure 5.5, we use the result from section 5.4 *for the posterior marginal distribution of* τ*:*

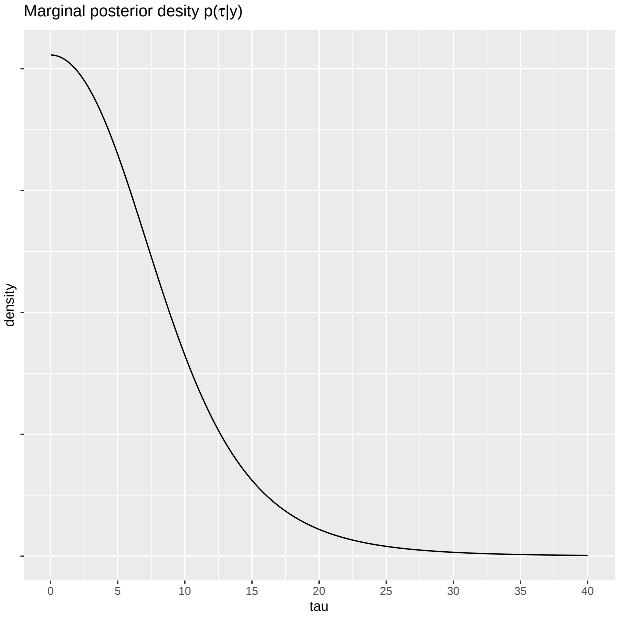
$$\begin{aligned} p(\tau|y) &\propto \frac{p(\tau)\prod_{j=1}^{J} N(\bar{y}_{.j}|\hat{\mu}, \sigma_j^2+\tau^2)}{N(\hat{\mu}|\hat{\mu}, V_\mu)} \\ &\propto p(\tau)V_\mu^{1/2}\prod_{j=1}^{J}(\sigma_j^2+\tau^2)^{-1/2}\exp\Big(-\frac{(\bar{y}_{.j}-\hat{\mu})^2}{2(\sigma_j^2+\tau^2)}\Big) \end{aligned}$$

and we use the uniform prior distribution $p(\tau) \propto 1$.

The following R code take evaluations of the density for τ *on the range* $(0, 40)$*:*

```
library(ggplot2)
library(tidyr)
library(gridExtra)
library(scales)
library(directlabels)

#Input the data
school <- c("A", "B", "C", "D", "E", "F", "G", "H")
effect <- c(28, 8, -3, 7, -1, 1, 18, 12) # those y_j's
sigma <- c(15, 10, 16, 11, 9, 11, 10, 18) # those sigma_j's
data <- data.frame(school = school, effect = effect, sigma = sigma)

#preliminary quantities for computing posterior inferences
J <- length(school) #number of theta

tau <- seq(0.01, 40, 0.01) #set a range of tau's.
                        #Start from 0.01 to avoid singularity

n <- length(tau) #just an intermediate value

hat_mu <- rep(NA, n)
InvV_mu <- rep(NA, n) #initialize the two values in equation 5.20

for (i in 1:n){
 hat_mu[i] <- sum(effect/(sigma^2 + (tau[i])^2))/sum(1/(sigma^2 + (tau[i])^2))
 InvV_mu[i] <- sum(1/(sigma^2 + (tau[i])^2))
}#define values from equation (5.20), for computation of posterior inference later

margin_post_tau <- rep(NA, n)
for (i in 1:n){
```

```
margin_post_tau[i] <- sqrt(1/InvV_mu[i])*prod(exp(-0.5*(effect - hat_mu[i])^2/
                      (sigma^2 + (tau[i])^2))/sqrt(sigma^2 + (tau[i])^2))
} #compute marginal posterior density for tau given data y

#store the results in data frame
inference <- data.frame(tau = tau, margin_post_tau = margin_post_tau)

#plot the posterior marginal density p(tau|y) (Figure 5.5)
ggplot(data = inference, aes(x=tau, y=margin_post_tau)) +
geom_line() +
theme(
  axis.text.y = element_blank()) + #hide the unnormalized values
scale_x_continuous(breaks = seq(0, 40, 5))  + #set the steps size to be 5
labs(title=expression(paste("Marginal posterior desity"," p(",tau,"|y)")))+
labs(y = "density")
```



We then plot Figure 5.6, which is the conditional posterior means of treatment effects, $E(\theta_j|\tau, y)$ *as a function of the between school standard deviation* τ.

The strategy is this:

Keep in mind we need to average on μ, *and since* $\mu|\tau, y \sim N(\hat{\mu}, V_\mu)$ *with*

$$\hat{\mu} = \frac{\sum_{j=1}^{J} \frac{1}{\sigma_j^2+\tau^2}\bar{y}_{\cdot j}}{\sum_{j=1}^{J} \frac{1}{\sigma_j^2+\tau^2}} \text{ and } V_\mu^{-1} = \sum_{j=1}^{J} \frac{1}{\sigma_j^2 + \tau^2}$$

we can plug $\hat{\mu}$ *into the posterior of* $\theta_j : \theta_j|\mu, \tau, y$ *to essentially get* $\theta_j|\tau, y$.

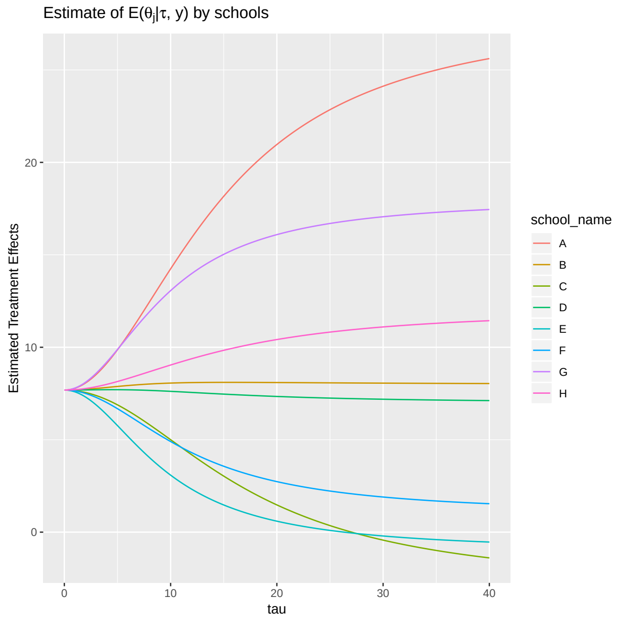
With the knowledge that $\theta_j|\mu, \tau, y \sim N(\hat{\theta}_j, V_j)$ *where*

$$\hat{\theta}_j = \frac{\frac{1}{\sigma_j^2}\bar{y}_{.j} + \frac{1}{\tau^2}\mu}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}} \text{ and } V_j = \frac{1}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}}$$

We can estimate $E[\theta_j|\tau, y]$ *by computing* $\hat{\theta}_j$ *using above equation but with* μ *replaced by* $\hat{\mu}$*.*

The following R code does the job above to plot Figure 5.6:

```
# figure 5.6
theta <- matrix(NA, n, J) #initialize the theta values
for (i in 1:n){
for (j in 1:J){
  theta[i, j] <- (effect[j]/(sigma[j])^2 +
                    hat_mu[i]/(tau[i])^2)/(1/(sigma[j])^2 + 1/(tau[i])^2)
}
} #fill in the corresponding value of theta_hat for different grid nubmers of tau
theta <- cbind(theta, tau) #bind y value with x value
colnames(theta) <- c(school, "tau") #name the variables
theta <- as.data.frame(theta) #make it a data frame
new_theta <- gather(theta, school_name, trt_effect,
                  A:H, factor_key=TRUE) #change data to long format

ggplot(data = new_theta, aes(x = tau, y = trt_effect, colour = school_name)) +
geom_line() +
labs(title = expression(paste("Estimate of E("
    ,theta[j],"|",tau,", y) by schools")),y =
      "Estimated Treatment Effects") # plot figure 5.6
```



Now to plot Figure 5.7, the conditional posterior standard deviations of treatment effects, $sd(\theta_j|\tau, y)$, as functions of the between-school standard deviation τ:

Look at the equation again that for $\theta_j|\mu, \tau, y$ we have the variance

$$V_j = \frac{1}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}}$$

and for $\mu|\tau, y$ we have the variance

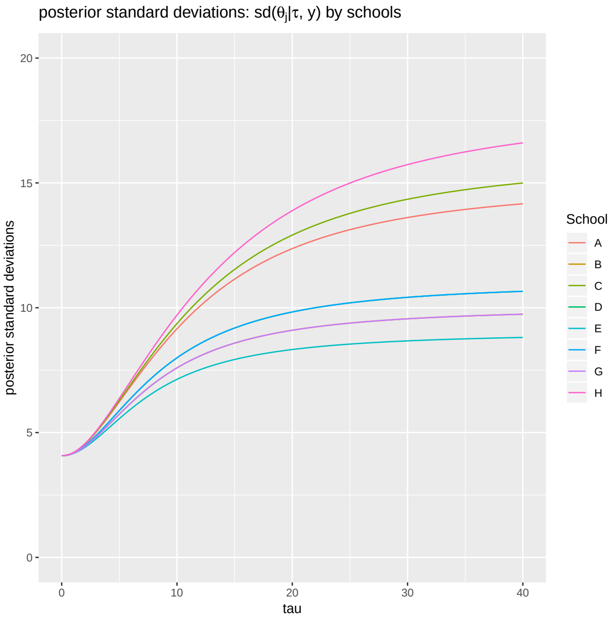
$$V_\mu^{-1} = \sum_{j=1}^{J} \frac{1}{\sigma_j^2 + \tau^2}$$

Then we have

$$\begin{aligned} Var(\theta_j|\tau, y) &= Var(\theta_j|\mu, \tau, y) + Var(\mu|\tau, y) \\ &= V_j + V_\mu \\ &= \frac{1}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}} + \frac{1}{\sum_{j=1}^{J} \frac{1}{\sigma_j^2 + \tau^2}} \end{aligned}$$

We can the take square root to acquire the standard deviation. The following R code makes the plot of Figure 5.7 by using above computation:

```
#figure 5.7
sd_theta <- matrix(NA, n, J) #initialize the theta values
for (i in 1:n){
for (j in 1:J){
  sd_theta[i, j] <- sqrt((sigma[j]^2/(sigma[j]^2 +
                          tau[i]^2))^2/(sum(1/(tau[i]^2 + sigma^2))) +
                            1.0/(1.0/(sigma[j]^2) + 1.0/(tau[i]^2)))
}
} #fill in the corresponding value of sd_theta for different grid nubmers of tau
sd_theta <- cbind(sd_theta, tau) #bind y value with x value
colnames(sd_theta) <- c(school, "tau") #name the variables
sd_theta <- as.data.frame(sd_theta) #make it a data frame
new_sd_theta <- gather(sd_theta, school_name, post_sd,
                  A:H, factor_key=TRUE) #change data to long format
ggplot(data = new_sd_theta, aes(x = tau, y = post_sd, colour = school_name)) +
geom_line() +
labs(title = expression(paste("posterior standard deviations: sd("
    ,theta[j],"|",tau,", y) by schools")),y =
       "posterior standard deviations", colour = "School") +
ylim (0, 20)#Figure 5.7
```



Now as is done in the textbook, we also draw 200 *simulations of* θ_j *by first drawing* τ *from* $p(\tau|y)$*, then drawing* μ *from* $p(\mu|\tau, y)$*, and finally drawing* θ *from* $p(\theta|\mu, \tau, y)$*.*

The following R code execute this process. To make results replicable, we set the seed number as 34*:*

```
#draw posterior tau, mu, and theta sequentially
set.seed(34) #set seed number
size <- 200 #set simulation size
post_tau <- sample(tau, size = size, replace = TRUE,
     prob = margin_post_tau)#draw posterior tau
hat_mu_post <- rep(NA, size) #initialize mean for posterior mu
sd_mu_post <- rep(NA, size) #initialize sd for posterior mu
for (i in 1:size){
hat_mu_post[i] <- sum(effect/(sigma^2 +
              (post_tau[i])^2))/sum(1/(sigma^2 + (post_tau[i])^2))
sd_mu_post[i] <- 1/sqrt(sum(1/(sigma^2 + (post_tau[i])^2)))
}#generate posterior mean and sd for mu
post_mu <- rnorm(n= size, mean= hat_mu_post,
               sd = sd_mu_post) #generate posterior mu
hat_theta_post <- matrix(NA, size, J) #initialize mean for posterior theta
sd_theta_post <- matrix(NA, size, J) #initialize sd for posterior theta
for (j in 1:J){
for (i in 1:size){
  hat_theta_post[i, j] <- (effect[j]/sigma[j]^2 + post_mu[i]/post_tau[i]^2)/
    (1/sigma[j]^2 + 1/post_tau[i]^2)
  sd_theta_post[i, j] <- sqrt(1/(1/sigma[j]^2 + 1/post_tau[i]^2))
}
}
post_theta <- matrix(NA, size, J) #initialize sample for posterior theta
for (j in 1:J){
post_theta[, j] <- rnorm(n = size, mean = hat_theta_post[, j],
                        sd = sd_theta_post[, j])
} #generate posteior theta for each school
```

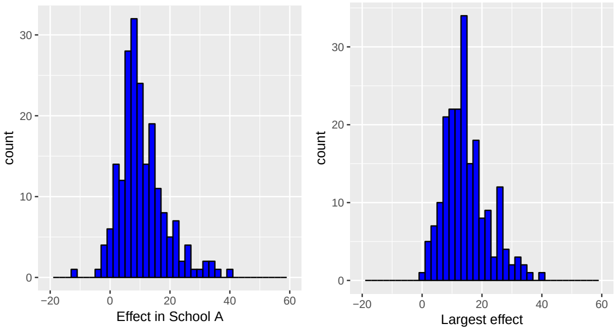
We then generate Table 5.3 to look at the posterior quantiles for each school:

```
post_quant <- matrix(NA, J, 5) #initialize for quantile summary
for (j in 1:J){
post_quant[j, ] <- quantile(x = post_theta[, j], probs =
            c(0.025, 0.25, 0.5, 0.75, 0.975))
} #compute posterior quantiles for theta in different schools
post_quant <- cbind(school, round(post_quant, digits = 1)) #get school names
post_quant <- as.data.frame(post_quant) #convert to data.frame
colnames(post_quant) <- c("School", "2.5%", "25%", "median", "75%", "97.5%")
print(post_quant) #show table 5.3

##   School 2.5% 25% median  75% 97.5%
## 1      A   -1 5.7    9.2 14.4  32.8
## 2      B -3.7 3.5    7.4 11.4  19.1
## 3      C -7.7 2.6    7.3   11  18.3
## 4      D -5.5 4.4    7.4 11.5  20.8
## 5      E -5.6 1.4    5.3  9.5  16.8
## 6      F -5.8 2.6      7   11  16.7
## 7      G -0.5 6.1   10.2 13.2  25.2
## 8      H   -5 4.4    7.9 12.7  26.1
```

Since we probably used different seeds than the book, our simulation has different values. However the general relationships of those quantiles between and within schools are similar to the one from the book.

We also generate Figure 5.8 to look at histograms of the effects for school A and largest effect:

```
#check the effect in school A and the largest effect
post_theta <- as.data.frame(post_theta) #convert posterior
                                      #simulation to data.frame
colnames(post_theta) <- school
post_theta$max <- apply(post_theta, 1, max)

hist_A <-ggplot(data = post_theta,
              aes(x=A)) + #histogram of posterior effect for school A
labs(x = "Effect in School A")+
xlim(c(-20, 60))+
coord_fixed(ratio=2.7) +
geom_histogram(binwidth = 2, color = "black", fill = "blue")
hist_max <-ggplot(data = post_theta,
                  aes(x=max)) + #histogram of posterior effect for school A
labs(x = "Largest effect")+
xlim(c(-20, 60))+
coord_fixed(ratio=2.6) +
geom_histogram(binwidth = 2, color = "black", fill = "blue")
grid.arrange(hist_A, hist_max, nrow = 1)
```



We now answer the questions in part (a):

(i): for each school j, the probability that its coaching program is the best of the eight. We do this by couting the number of observations in our simulation where school j has best effect, and then compute the percentage of these observations out of 200.

```
#compute probability of school j is the best
best_prob <- as.data.frame(matrix(ncol= J, nrow=0)) #assign space
#for storing results
for (j in 1:J){
best_prob[1, j] <- percent(mean(post_theta[, j] == post_theta[, 9]))
} #compute the probability of each school has best effects
colnames(best_prob) <- school #retrieve school names
best_prob

##       A     B     C     D     E     F     G     H
## 1 26.5% 10.0% 7.00% 11.5% 6.00% 6.50% 20.0% 12.5%
```

Not surprisingly, with the given the data, school with the highest score y_j gets the highest chance to have best effect, and the order goes on in accordance with the rest of the scores. This makes sense consider we started with a non-informative prior on τ. So the data is dominant here.

(ii): For each pair of j and k, the probability that the coaching program in school j is better than that in school k.

Again we compare the scores between j ad k on each observation of the simulation, and the cmopute the percentage out of 200.

```
pair_prob <- as.data.frame(matrix(ncol= J, nrow = J)) #assign space
        #for storing results
for (j in 1:J){
for(i in 1:J){
  pair_prob[i, j] <- percent(mean(post_theta[, j] > post_theta[, i]))
}
}
colnames(pair_prob) <- c("A >", "B >", "C >", "D >", "E >", "F >", "G >", "H >")
row.names(pair_prob) <- school
pair_prob #pairwise comparison on probabilities which school has better effects

##     A >   B >   C >   D >   E >   F >   G >   H >
## A    0% 36.5% 30.5% 32.5% 25.0% 30.0% 50.5% 35.5%
## B 63.5%    0% 50.5% 51.5% 38.5% 46.0% 63.5% 52.0%
## C 69.5% 49.5%    0% 53.5% 43.0% 48.5% 69.5% 56.5%
## D 67.5% 48.5% 46.5%    0% 44.5% 48.0% 63.5% 54.0%
## E 75.0% 61.5% 57.0% 55.5%    0% 56.5% 72.5% 63.0%
## F 70.0% 54.0% 51.5% 52.0% 43.5%    0% 69.0% 58.0%
## G 49.5% 36.5% 30.5% 36.5% 27.5% 31.0%    0% 41.5%
## H 64.5% 48.0% 43.5% 46.0% 37.0% 42.0% 58.5%    0%
```

The table above shows pairwise comparison results. Particularly, school A has an over 50% *chance of having better effect than all other schools, which makes sense consider that from the data school A has the highest score.*

part b*:*

When $\tau = +\infty$, we have $\theta_j|\mu, \infty, y \sim N(\hat{\theta}_j, V_j)$ where

$$\hat{\theta}_j = \frac{\frac{1}{\sigma_j^2}y_j + 0}{\frac{1}{\sigma_j^2} + 0} = y_j, \qquad V_j = \frac{1}{\frac{1}{\sigma_j^2} + 0} = \sigma_j^2$$

what it says here is that when the hyperparameter $\tau = +\infty$, we do not get any information from the prior and our posterior distribution for θ_j is completely depending on the data y_j.

Now to answer the same question as in part (i), we first want to explain how it is manually doable, although we are not actually going to do it (becuase for 8 schools, there will be $\binom{8}{2} = 28$ pairwise comparisons we need to compute).

Remember that $\theta_j|\mu, +\infty, y$ are independent from each other following normal distributions with the mean and variance given above. So the difference of $\theta_i - \theta_j$ is also normal. For example,

$$\begin{aligned}\theta_1 - \theta_2|\mu, +\infty, y \sim N(\hat{\theta}_1 - \hat{\theta}_2, V_1 + V_2) &= N(y_1 - y_2, \sigma_1^2 + \sigma_2^2)\\ &= N(28 - 8, 15^2 + 10^2) = N(20, 325)\end{aligned}$$

So

$$\begin{aligned}P(\theta_1 - \theta_2 > 0) = P(N(20, 325) > 0) &= P(\frac{N(20, 325) - 20}{\sqrt{325}} > \frac{0 - 20}{\sqrt{325}})\\ &= P(N(0, 1) > -\frac{20}{\sqrt{325}}) = 86.6\%\end{aligned}$$

which is higher than 63.5% as we have acquired in part (a). This is understandable since we are only relying on our observed data here instead of also having part of prior information.

To compute the probability of a school having the best effect, take school A for example, then we have:

$$\begin{aligned}P(\theta_1 = \max_{1 \le i \le 8} \theta_i|\mu, +\infty, y) &= P(\theta_1 > \theta_j, j = 2, \ldots, 8|\mu, +\infty, y)\\ &= \prod_{j=2}^{8} P(\theta_1 > \theta_j|\mu, \tau = +\infty, y)\end{aligned}$$

The last equality is due to again that these events are independent.

But as we can see that it would be tedious to compare each pair manually, we use the following R code to do it:

```r
set.seed(34)
post_theta_b <- as.data.frame(matrix(ncol= J, nrow = size))
for (j in 1:J){
post_theta_b[, j] <- rnorm(n = size, mean = effect[j],
                         sd = sigma[j])
} #generate posteior theta for each school, in part b
post_theta_b$max <- apply(post_theta_b, 1, max)

#compute probability of school j is the best
best_prob_b <- as.data.frame(matrix(ncol= J,
                    nrow=0)) #assign space for storing results
for (j in 1:J){
best_prob_b[1, j] <- percent(mean(post_theta_b[, j] == post_theta_b[, J + 1]))
} #compute the probability of each school has best effects
colnames(best_prob_b) <- school #retrieve school names
best_prob_b

##       A     B     C     D  E     F     G     H
## 1 59.5% 2.50% 1.00% 3.50% 0% 1.00% 17.0% 15.5%
```

```
pair_prob_b <- as.data.frame(matrix(ncol= J,
                          nrow = J)) #assign space for storing results
for (j in 1:J){
for(i in 1:J){
  pair_prob_b[i, j] <- percent(mean(post_theta_b[, j] > post_theta_b[, i]))
}
}
colnames(pair_prob_b) <- c("A >", "B >", "C >", "D >", "E >", "F >", "G >", "H >")
row.names(pair_prob_b) <- school
pair_prob_b #pairwise comparison on probabilities which school has better effects

##     A >   B >   C >   D >   E >   F >   G >   H >
## A    0% 14.0% 6.00% 10.5% 5.00% 5.50% 27.5% 23.5%
## B 86.0%    0% 33.0% 44.5% 28.5% 33.5% 76.0% 60.5%
## C 94.0% 67.0%    0% 73.0% 57.0% 57.5% 89.0% 80.5%
## D 89.5% 55.5% 27.0%    0% 29.0% 37.5% 77.5% 61.5%
## E 95.0% 71.5% 43.0% 71.0%    0% 54.0% 91.0% 76.5%
## F 94.5% 66.5% 42.5% 62.5% 46.0%    0% 85.0% 73.0%
## G 72.5% 24.0% 11.0% 22.5% 9.00% 15.0%    0% 38.0%
## H 76.5% 39.5% 19.5% 38.5% 23.5% 27.0% 62.0%    0%
```

As we can see that our simulation gives 86% chance that school A has better effect than school B, which is the same as our manual computation earlier.

For part *(c):*

To see the difference between (a) and (b), which we already briefly mentioned in our solutions above, that the difference is caused by our different assumptino on τ. In (a) our τ follows a uniform prior $p(\tau) \propto 1$, and hence the marginal posterior $p(\tau|y)$ and the conditional posteriors $p(\mu|\tau, y)$ and $p(\theta|\mu, \tau, y)$ all have non-trival forms that includes the informatin of τ, which is the prior standard deviation of θ_j.

However for (b) since we assume $\tau = +\infty$, we no longer have a hierarchical model instead we just have a regular Bayesian model with constant priors θ_j and normal posteriors $\theta_j|y \sim N(y_j, \sigma_j^2)$ where y_j, σ_j^2 known.

We can see from the pairwise comparison table that part (b) is impacted more by the data without considering the variability of θ_j in the prior.

For part (d)

If we assue $\tau = 0$, then we are assuming that all $\theta_j = \mu$ a priori. This is to say

$$p(\theta_j|\mu) = \begin{cases} 1 & \textit{if } \theta_j = \mu \\ 0 & \textit{otherwise} \end{cases} \qquad\qquad p(y_j|\theta_j) \sim N(\theta_j, \sigma_j^2)$$

and we will have complete conditional posterior as:

$$\theta_j|\mu, \tau(=0), y = \hat{\theta}_j = \mu$$

This is to say all schools have the same posterior mean effect μ, regardless what prior we choose for μ. So for any school j

$$P\Big(\theta_j = \max\{\theta_i, 1 \le i \le 8\}|\mu, y\Big) = P(\theta_j = \mu|\mu, y) = 1$$

i.e., every school has 100% to be the best school in terms of the treatment effect (but not really better than the other schools since very school gets the same posterior effect).

Also for school j and k, we have

$$P(\theta_j > \theta_k|\mu, y) = P(\mu > \mu|\mu, y) = 0$$

Question 2: Chapter 5 Question 14.

Solution 2. for part (a)*:*

We have $\theta_j|\alpha, \beta \sim$ Gamma(α, β) and $y_j|\theta_j \sim$ Poisson(θ_j), so the joint posterior distribution can be represented as:

$$\begin{aligned}
p(\theta, \alpha, \beta|y) &\propto \underbrace{p(\alpha, \beta)}_{\textit{hyper prior}} \cdot \underbrace{p(\theta|\alpha, \beta)}_{\textit{population}} \cdot \underbrace{p(y|\theta, \alpha, \beta)}_{\textit{likelihood}} \\
&= p(\alpha, \beta) \cdot p(\theta|\alpha, \beta) \cdot p(y|\theta) \\
&= p(\alpha, \beta) \cdot \Big(\prod_{j=1}^{J} \frac{\beta^\alpha}{\Gamma(\alpha)} \theta_j^{\alpha-1} \cdot e^{-\beta\theta_j}\Big) \cdot \Big(\sum_{j=1}^{J} \frac{\theta_j^{y_j}}{y_j!} \cdot e^{-\theta_j}\Big)
\end{aligned}$$

For noninformative prior, we may choose $p(\alpha, \beta) \propto 1$.

for part (b)*:*

To compute the marginal posterior density of the hyperparameters, notice that:

$$p(\alpha, \beta|y) \cdot p(\theta|\alpha, \beta, y) = p(\theta, \alpha, \beta|y)$$

So

$$p(\alpha, \beta|y) = \frac{p(\theta, \alpha, \beta|y)}{p(\theta|\alpha, \beta, y)}$$

We already computed the numerator in **part(a)***, now for the denominator, notice that*

$$\begin{aligned}
p(\theta|\alpha, \beta, y) = \frac{p(\theta, y|\alpha, \beta)}{p(y|\alpha, \beta)} &= \frac{p(\theta|\alpha, \beta) \cdot p(y|\theta, \alpha, \beta)}{p(y|\alpha, \beta)} \\
&\propto p(\theta|\alpha, \beta) \cdot p(y|\theta) = \Big(\prod_{j=1}^{J} p(\theta_j|\alpha, \beta)\Big)\Big(\prod_{j=1}^{J} p(y_j|\theta_j)\Big) \\
&\propto \Big(\prod_{j=1}^{J} \theta_j^{\alpha-1} \cdot e^{-\beta\theta_j}\Big)\Big(\prod_{j=1}^{J} \theta_j^{y_j} e^{-\theta_j}\Big) = \prod_{j=1}^{J} \theta_j^{\alpha+y_j-1} e^{-(\beta+1)\theta_j} \sim \prod_{j=1}^{J} \Gamma(\alpha + y_j, \beta + 1)
\end{aligned}$$

so we have found that

$$p(\theta|\alpha,\beta,y)=\prod_{j=1}^{J}\frac{(\beta+1)^{\alpha+y_j}e^{-(\beta+1)\theta_j}\theta_j^{\alpha+y_j-1}}{\Gamma(\alpha+y_j)}$$

Plug this into the second equation of **part (b)** *we have:*

$$\begin{aligned}p(\alpha,\beta|y)&=\frac{p(\theta,\alpha,\beta|y)}{p(\theta|\alpha,\beta,y)}\\&=\frac{p(\alpha,\beta)\cdot\left(\prod_{j=1}^{J}\frac{\beta^\alpha}{\Gamma(\alpha)}\theta_j^{\alpha-1}\cdot e^{-\beta\theta_j}\right)\cdot\left(\sum_{j=1}^{J}\frac{\theta_j^{y_j}}{y_j!}\cdot e^{-\theta_j}\right)}{\prod_{j=1}^{J}\frac{(\beta+1)^{\alpha+y_j}e^{-(\beta+1)\theta_j}\theta_j^{\alpha+y_j-1}}{\Gamma(\alpha+y_j)}}\\&\propto p(\alpha,\beta)\prod_{j=1}^{J}\frac{\beta^\alpha\times\Gamma(\alpha+y_j)}{(\beta+1)^{\alpha+y_j}\times\Gamma(\alpha)}\end{aligned}$$

Notice that in **part (c)** *and* **part(d)** *we will be asked to check if our posterior density for hyperparameters is integrable or not, particularly if it is not integrable, we need to alter it and repeat this part, so we might just as well take a look for now what prior $p(\alpha,\beta)$ we should choose in order to make sure that we will have an integrable posterior distribution.*

A necessary condition for $p(\alpha,\beta|y)$ to be integrable is that:

$$\lim_{(\alpha,\beta)\to(\alpha,+\infty)}p(\alpha,\beta|y)=\lim_{(\alpha,\beta)\to(+\infty,\beta)}p(\alpha,\beta|y)=0$$

Here $(\alpha,\beta)\to(\alpha,+\infty)$ means fixing α when letting $\beta\to+\infty$. The other one is similar.

Observe

$$\prod_{j=1}^{J}\frac{\beta^\alpha\times\Gamma(\alpha+y_j)}{(\beta+1)^{\alpha+y_j}\times\Gamma(\alpha)}$$

When $(\alpha,\beta)\to(\alpha,+\infty)$,

$$\prod_{j=1}^{J}\frac{\beta^\alpha\times\Gamma(\alpha+y_j)}{(\beta+1)^{\alpha+y_j}\times\Gamma(\alpha)}\simeq Constant\cdot\beta^{-\sum_{j=1}^{J}y_j}\text{ when }\beta\text{ is large}$$

Since all the y_j are pretty big positive number, so β has a large negative power, hence for β alone we could have a very loose restriction on the form of the prior $p(\alpha,\beta)$.

On the other hand, when $(\alpha,\beta)\to(+\infty,\beta)$,

$$\prod_{j=1}^{J}\frac{\beta^\alpha\times\Gamma(\alpha+y_j)}{(\beta+1)^{\alpha+y_j}\times\Gamma(\alpha)}\simeq Constant\cdot(1+\frac{1}{\beta})^{-J\alpha}\cdot\alpha^{\sum_{j=1}^{J}y_j}\text{ when }\alpha\text{ is large}$$

Since the exponential function $(1+\frac{1}{\beta})^{-J\alpha}$ dominate power function $\alpha^{\sum_{j=1}^{J}y_j}$ at large value, so we can also be comfortable to say that the above term goes to 0 when $\alpha\to+\infty$ while fixing β.

In summary, any proper marginal prior $p(\alpha, \beta)$ can be chosen to guarantee integrability of our posterior. We can also even choose improper prior, particularly including the usual noninformative prior uniform distribution.

Suppose we choose the following prior such that

$$p(\log \frac{\alpha}{\beta}, \log \beta) \propto 1$$

notice that

$$\alpha = e^{\log \frac{\alpha}{\beta} + \log \beta} \text{ and } \beta = e^{\log \beta}$$

then the Jacobian transform matrix is:

$$J = \begin{bmatrix} \frac{\partial \alpha}{\partial \log \frac{\alpha}{\beta}} & \frac{\partial \alpha}{\partial \log \beta} \\ \frac{\partial \beta}{\partial \log \frac{\alpha}{\beta}} & \frac{\partial \beta}{\partial \log \beta} \end{bmatrix} = \begin{bmatrix} \alpha & \alpha \\ 0 & \beta \end{bmatrix}$$

So we have $|J| = \alpha\beta$ and

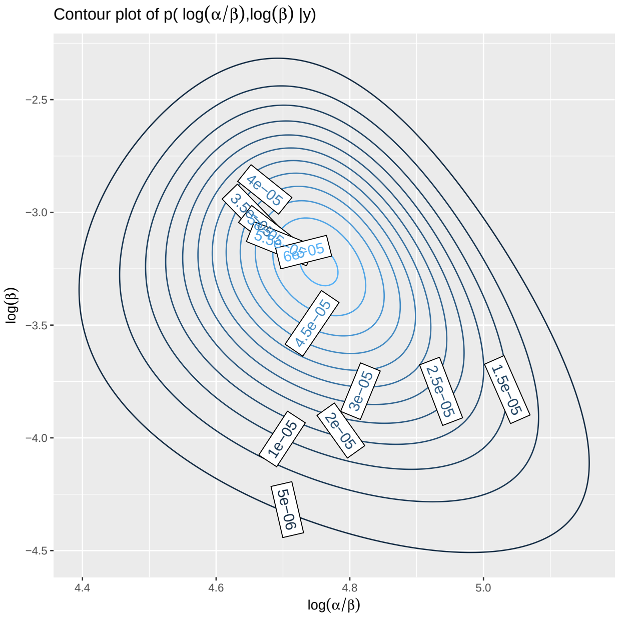
$$\begin{aligned} p(\alpha, \beta) &\propto (\alpha\beta)^{-1} \\ p(\log \frac{\alpha}{\beta}, \log \beta | y) = |J| \cdot p(\alpha, \beta | y) &\propto \alpha\beta \cdot p(\alpha, \beta) \prod_{j=1}^{J} \frac{\beta^{\alpha} \times \Gamma(\alpha + y_j)}{(\beta + 1)^{\alpha + y_j} \times \Gamma(\alpha)} \\ &\propto \prod_{j=1}^{J} \frac{\beta^{\alpha} \times \Gamma(\alpha + y_j)}{(\beta + 1)^{\alpha + y_j} \times \Gamma(\alpha)} \end{aligned}$$

Hence

$$\log \Big(p(\log \frac{\alpha}{\beta}, \log \beta | y) \Big) \propto \sum_{j=1}^{J} \Big[\log \Gamma(\alpha + y_j) + \alpha \log \beta - \log \Gamma(\alpha) - (\alpha + y_j) \log(\beta + 1) \Big]$$

The following code makes a contour plot of $p(log(\alpha/\beta), log(\beta))$:

```r
#create the data
y <- c(16+58,9+90,10+48,13+57,19+103,20+57,18+86,17+112,35+273,55+64)

#create contour plot for p(log(alpha/beta), log(beta))
a <- seq(3,7,4/1000) #grid value for log(alpha/beta)
b <- seq(-8,-1,7/1000) #grid value for log(beta)
z <- matrix(0,length(a),length(b)) #matrix to store density values at grids
size <- length(a)

for (i in 1:size){
   for (j in 1:size){
 t1<-exp(a[i]+b[j]) #alpha
 t2<-exp(b[j]) #beta
z[i,j]<-
   sum(lgamma(t1+y)+log(t2^t1)-lgamma(t1)-log((t2+1)^(t1+y)))
   }
 }#compute unnormalized density values at grid points
 z<-z-max(z)#these two lines are just for
 z<-exp(z) #numerical stableness

post_hyper <- data.frame(alpha = numeric(1001^2),
        beta= numeric(1001^2),
        post_density = numeric(1001^2))#create dataframe for contour map
#preparing data for ggplot
post_hyper$alpha <- rep(a, each = size)
post_hyper$beta <- rep(b, size)
post_hyper$post_density <- as.vector(t(z))

#normalize the grid values so it become probability density
post_hyper$post_density <- post_hyper$post_density/sum(post_hyper$post_density)

#make contour plot
hyper_contour <- ggplot(data = post_hyper, aes(post_hyper$alpha,
    post_hyper$beta,
    z = post_hyper$post_density)) +
  geom_contour(aes(colour = ..level..)) +
  labs(title = expression(paste("Contour plot of p( ",log(alpha/beta),
  ",",log(beta)," |y)")))+
  xlab(expression(log(alpha/beta)))+
  ylab(expression(log(beta)))

direct.label(hyper_contour, list("angled.boxes"))
```



Now to simulate random draws from $p(\alpha, \beta|y)$*. The following R code process this with successive substitution sampling.*

To further explain, it is like simulating a random walk with the following algorithm:

(1) we first pick a row number t *based on the marginal posterior probability* $p(\log(\frac{\alpha}{\beta})|y)$*, which helps us find a value of* $\log(\frac{\alpha}{\beta})$*,*

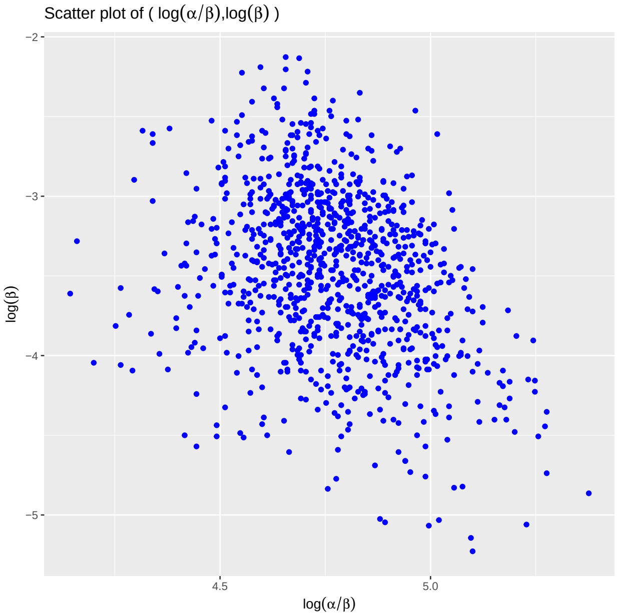
(2) we then condition on this row number t*, draw a column number* s *based on the conditional posterior distribution* $p(\log(\beta)|\log(\frac{\alpha}{\beta}), y)$*,*

(3) similar to step 2, now we condition on this column number s*, update the row number* t *based on the conditional posterior distribution* $p(\log(\frac{\alpha}{\beta})|\log(\beta), y)$*, which can point us to the next value of* $\log(\frac{\alpha}{\beta})$*,*

(4) repeat step (2), and so on ...

```
#simulate random draw of (log(alpha/beta), log(beta))
prob<-apply(z,1,sum)
prob<-prob/sum(prob)#normalize the value of z matrix

#create two vectors to store the simulation result.
aa <- rep(0,1100)#the first 100 is for burning out
bb <- rep(0,1100)

#successive subsitution sampling
#the line start with s <- and t <- shows successive substitution process

t <- sample(size,1,prob=prob)#draw a row number from marginal posterior
```

```r
for (i in 1:1100){
aa[i] <- a[t]#find the log(alpha/beta) value corresponding to that row

#draw a column number, conditioned on the previous row
s <- sample(size, 1, prob = z[t, ])

bb[i] <- b[s]#find the log(beta) value corresponding to that column

#draw a row number, conditioned on the previous column.
t <- sample(size, 1, prob = z[, s])
}

aa <- aa[101:1100]#throw away the burning out part
bb <- bb[101:1100]

#make scatter plot
scatter_data <-data.frame(aa = numeric(1000), bb = numeric(1000))
scatter_data$aa <- aa #fill in x value
scatter_data$bb <- bb #fill in y value

ggplot(data= scatter_data, aes(x = aa, y = bb)) +
geom_point(colour = "blue") +
labs(title = expression(paste("Scatter plot of ( ",log(alpha/beta),
                              ",",log(beta)," )")))+
xlab(expression(log(alpha/beta)))+
ylab(expression(log(beta)))
```



As we can see that the scatter plot also reflects same pattern as is shown by the contour plot.

For part (c):

To see if the posterior density $p(\alpha, \beta|y)$ is integrable or not, we already did some analytical observation in **part (b)***, that by our choice of the prior, we should be getting an integrable kernel.*

In the mean time, if we check on the contour line from our controu plot, we find that towards boundary of the area, the posterior density is really appraochin 0 *(on the* 10^{-6} *scale), so* $p(\alpha, \beta|y)$ *should be integrable. Thus there is no need to alter and repeat the steps. Hence we skip* **part (d)**.

For part (e):

We have shown in **part (b)** *that*

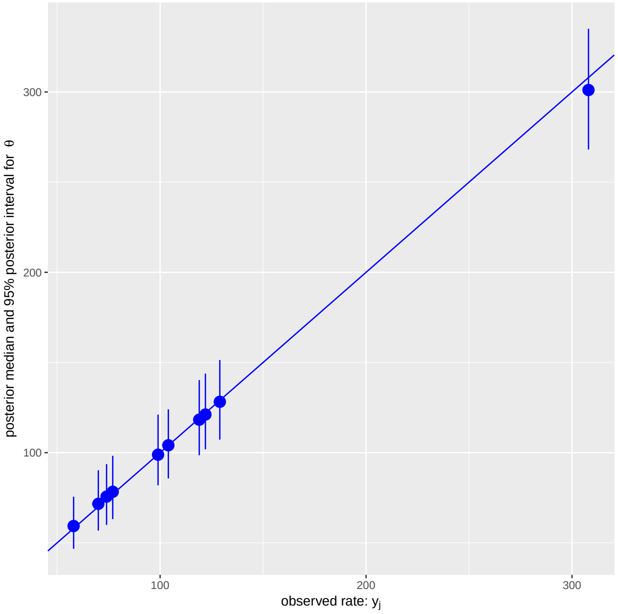
$$p(\theta_j|\alpha, \beta, y) \sim \Gamma(\alpha + y_j, \beta + 1)$$

Since we already drew $(\log(\frac{\alpha}{\beta}), \log(\beta))$ *in part (b), here we will just transform them back to* (α, β)*, and then draw* θ *based on the conditional posterior distribution above.*

```r
#draw samples for theta_j
alpha<-exp(aa+bb) #transform to alpha
beta<-exp(bb) #transform to beta
theta<-matrix(0,1000,10) #draw 1000 posterior sample for each site
for(i in 1:1000){
  for(j in 1:10){
    theta[i,j]<-rgamma(1,alpha[i]+y[j],beta[i]+1)
  }
}
```

To observe our draws, we plot the median of the 1000 posterior sample for θ *on each site, as well as the* 95% *confidence intervals (by connecting the* 2.5% *and* 97.5% *quantile of the sample). We also sketch the* 45 *degree reference line.*

```r
theta_plot <- data.frame(site = y, median = numeric(10),
              lquan = numeric(10), uquan = numeric(10))
for(j in 1:10){
  theta_plot$median[j] <- median(theta[,j])
  theta_plot$lquan[j] <- quantile(theta[, j], 0.025)
  theta_plot$uquan[j] <- quantile(theta[, j], 0.975)
}#compute median, lower and upper quantile for each site

ggplot(data = theta_plot, aes(group = site)) +
geom_point(aes(x = site, y = median), colour = "blue", size = 4) +
geom_segment(aes(x = site, y = lquan,
                 xend = site, yend = uquan), colour = "blue")+
geom_abline(intercept = 0, slope = 1, colour = "blue")+
labs(x = expression(paste("observed rate: ",y[j])),
y= expression(paste("posterior median and 95% posterior interval for  ",theta))
```



As we can see the median lies pretty close to the reference line.