

Tutorial on Bayesian Statistics. Homework from BDA3

Fernando Hoces de la Guardia

Guided by: Susan Paddock

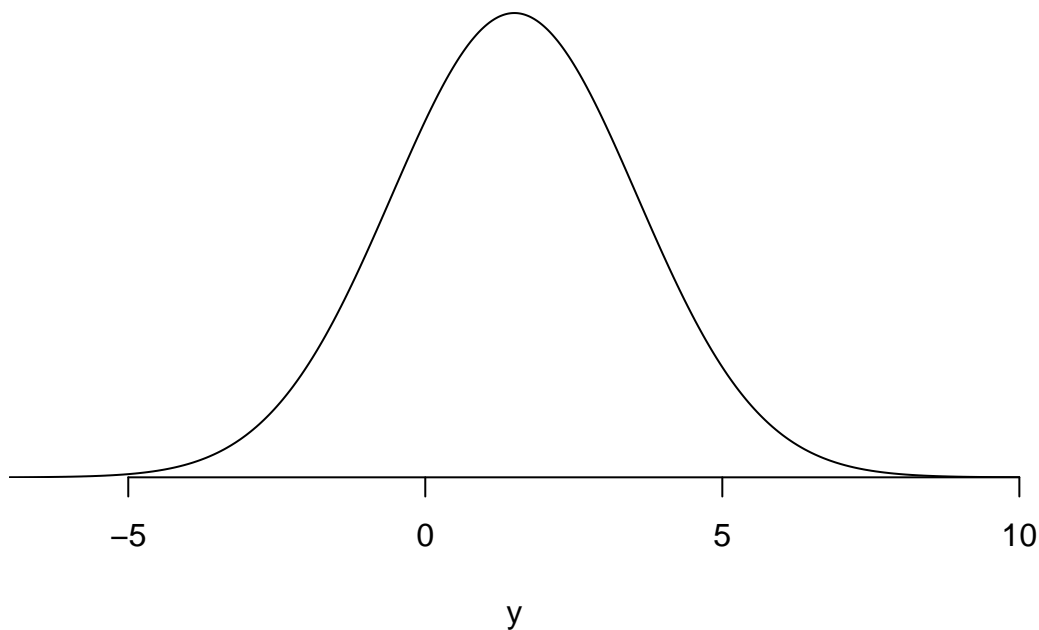
Chapter 1

1.1

1a:

$$\begin{aligned} p(y) &= \frac{1}{2} (p(y|\theta = 1) + p(y|\theta = 2)) \\ &= \frac{1}{2} (N(y|1, 2^2) + N(y|2, 2^2)) \end{aligned}$$

```
domain      <- seq(-7,10,.02)
dens        <- 0.5*dnorm(domain,1,2) + 0.5*dnorm(domain,2,2)
plot (domain, dens, ylim=c(0,1.1*max(dens)),
     type="l", xlab="y", ylab="", xaxs="i",
     yaxs="i", yaxt="n", bty="n", cex=2)
```



1b:

$$p(\theta = 1|y = 1) = \frac{p(\theta = 1)p(y = 1|\theta = 1)}{\sum_{i=1}^2 p(\theta = i)p(y = 1|\theta = i)}$$

$$= \frac{0.5N(1|1, 4)}{\sum_{i=1}^2 0.5N(1|i, 4)}$$

```
p.theta.1 <- function(sigma) {
  res1 <- (0.5*dnorm(1,1,sigma)) / (sum(0.5*dnorm(1,c(1,2),sigma)))
  return(res1)
}
```

Evaluating the last expression in the respective cumulative distribution function we get:0.5312094. **Note: even though we are adding “discrete” number of probabilities, we are still in the continuous space (but for $y = 1$) and should evaluate the probabilities in the density function.**

1c:

Table 1: Posterior probability of $\theta = 1$, as a function of σ

σ	$p(\theta = 1 y = 1)$
0.25	0.9996646
0.5	0.8807971
1	0.6224593
2	0.5312094
4	0.5078119
8	0.5019531

1.3 [NEW]

First: compute posterior of probability of having Xx genes knowing that parents have brown eyes and individual has brown eyes:

$$p(Xx|\text{Person and parents have brown eyes}) =$$

$$\frac{Pr(Xx|(XX, XX))Pr((XX, XX)) + Pr(Xx|(Xx, XX))Pr((Xx, XX)) + Pr(Xx|(Xx, Xx))Pr((Xx, Xx))}{Pr(Brown|(XX, XX))Pr((XX, XX)) + Pr(Brown|(Xx, XX))Pr((Xx, XX)) + Pr(Brown|(Xx, Xx))Pr((Xx, Xx))}$$

$$= \frac{0 \cdot (1-p)^4 + 1/2 \cdot 4p(1-p)^3 + 1/2 \cdot 4p(1-p)^2}{1 \cdot (1-p)^4 + 1 \cdot 4p(1-p)^3 + (1 - Pr(Blue|(Xx, Xx))) \cdot 4p(1-p)^2}$$

$$= \frac{0 \cdot (1-p)^4 + 1/2 \cdot 4p(1-p)^3 + 1/2 \cdot 4p(1-p)^2}{1 \cdot (1-p)^4 + 1 \cdot 4p(1-p)^3 + 3/4 \cdot 4p(1-p)^2}$$

$$= \frac{2p}{1 + 2p}$$

This posterior ($p(Xx|\text{Person and parents have brown eyes})$) will be the new prior when computing the

probability that Judy's is Xx given that all her kids are brown eyed (husband is Xx).

$$\begin{aligned}
p(\text{Judy} = Xx | \text{all n kids brown eyes (AKBE)} + \text{previous info.}) &= \\
&= \frac{P(\text{Judy} = Xx)P(\text{AKBE} | \text{Judy} = Xx)}{P(\text{Judy} = Xx)P(\text{AKBE} | \text{Judy} = Xx) + P(\text{Judy} \neq Xx)P(\text{AKBE} | \text{Judy} \neq Xx)} \\
&= \frac{\frac{2p}{1+2p}(\frac{3}{4})^n}{\frac{2p}{1+2p}(\frac{3}{4})^n + \frac{1}{1+2p}}
\end{aligned}$$

Now we want to compute the probability that the first grandchildren will have blue eyes. We know that all Judy's children have brown eyes, hence have genes Xx or XX . It follows that the grandkids will have blue eyes only if the kids have Xx genes. A kid will have Xx if Judy has Xx and she and her husband provide Xx ($pr = 2/3$, remember that the kid has brown eyes) or if she has XX and her husband provide Xx ($pr = 1/2$). Hence the probability of any kid to be Xx is:

$$\begin{aligned}
P(\text{kid} = Xx | \text{all info}) &= P(\text{kid} = Xx | \text{all info} + \text{Judy} = Xx)P(\text{Judy} = Xx) + \\
&P(\text{kid} = Xx | \text{all info} + \text{Judy} \neq Xx)P(\text{Judy} \neq Xx) \\
&= \left(\frac{2}{3}\right) \frac{\frac{2p}{1+2p}(\frac{3}{4})^n}{\frac{2p}{1+2p}(\frac{3}{4})^n + \frac{1}{1+2p}} + \left(\frac{1}{2}\right) \frac{\frac{1}{1+2p}(\frac{3}{4})^n}{\frac{2p}{1+2p}(\frac{3}{4})^n + \frac{1}{1+2p}}
\end{aligned}$$

Conditional on a kid having Xx the probability of having a grandchild with blue eyes is $0, 1/4$ and $1/2$ if the spouse is XX , Xx and xx respectively. Each spouse type has a probability of $(1-p)^2$, $2p(1-p)$ and p^2 respectively. Hence, the probability of a grandkid with blue eyes is:

$$\begin{aligned}
P(\text{Grandchildren} = \text{Blue} | \text{all info}) &= \frac{\frac{2}{3} \frac{2p}{1+2p} + \frac{1}{2} \frac{1}{1+2p}}{\frac{2p}{1+2p}(\frac{3}{4})^n + \frac{1}{1+2p}} \left(\frac{1}{4} 2p(1-p) + \frac{1}{2} p^2 \right) \\
&= \frac{\frac{2}{3} \frac{2p}{1+2p} + \frac{1}{2} \frac{1}{1+2p}}{\frac{2p}{1+2p}(\frac{3}{4})^n + \frac{1}{1+2p}} \left(\frac{1}{2} p \right)
\end{aligned}$$

Note: This solution was reverse engineered from [here](#).. A few additional lines were added.

1.6 [NEW]

The trick here is that the probability of having a fraternal twin birth with two boys is $1/4 \times 1/125$ and a identical twin birth of boys is $1/2 \times 1/300$. Hence the probability that Elvis had an identical twin is $5/11$.

1.7 Let's Make a Deal

Calculate the probability of winning for each box after one of the empty boxes has been revealed and is not a winning box.

Lets define the following events:

- * A : The participant chose the right box at the beginning.
- * B : The host opens a particular box, among the unchosen ones, such that is empty.
- * C : Among the unchosen boxes the host chooses a empty box.

And let's compute the probabilities of each of this events.

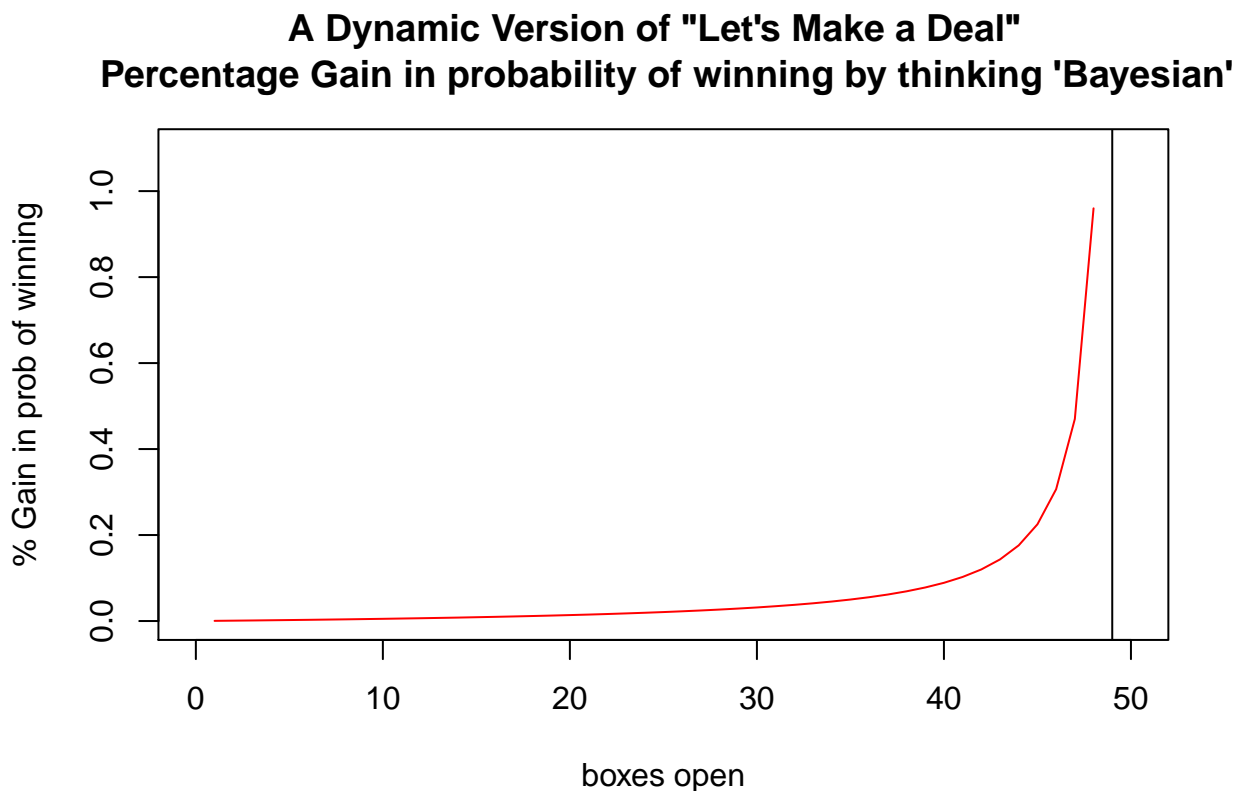
$$\begin{aligned}
Pr(A) &= 1/3 \\
Pr(C) &= 1/2 \\
Pr(B) &= Pr(B|A)Pr(A) + Pr(B|\neg A)Pr(\neg A) = (1/2) * (1/3) + Pr(B|\neg A) * (2/3) \\
&= 1/6 + 2/3 * (Pr(B|\neg A, C)Pr(C) + Pr(B|\neg A, \neg C)Pr(\neg C)) \\
&= 1/6 + 2/3 * (1 * (1/2) + 0 * (1/2)) = 1/2
\end{aligned}$$

Using Bayes' theorem we have that the probability of choosing the right box from the beginning, conditional on a unchosen box being revealed as a losing one is:

$$Pr(A|B) = \frac{Pr(A)Pr(B|A)}{Pr(B)} = \frac{(1/3) * (1/2)}{1/2} = \frac{1}{3}$$

The participant's chances are not equal across remaining boxes! She is worst of staying with her original choice (33% probability of winning instead of 50%).

More generally if there were n boxes in total and i boxes were revealed, we have that the wrong way of updating the probabilities ($1/(n-i)$) and the Bayesian update ($\frac{i+n*(n-1-i)}{n*(n-i)*(n-i-1)}$) differ significantly as $i \rightarrow n$. For example the following graph plots both probabilities of winning in a contest with 50 boxes as the host opens i boxes.



Looking at the graph it seems that the advantages of thinking in a Bayesian fashion are certainly parameter-specific. Also notice that the player here chooses a “stubborn” strategy, I suspect that if she changes boxes in an optimal way the improvement in her chances will be slightly less. Maybe that is the reason why we don't think in a Bayesian fashion all the time.

Chapter 2

2.1

$$\begin{aligned}
 P(\theta) &= \text{Beta}(4, 4) \\
 P(y|\theta) &= \text{Bin}(y|n, \theta) \\
 \Rightarrow P(\theta|y) &= \text{Beta}(4 + y, 4 + (n - y))
 \end{aligned}$$

The **wrong** way to answer the question would be:

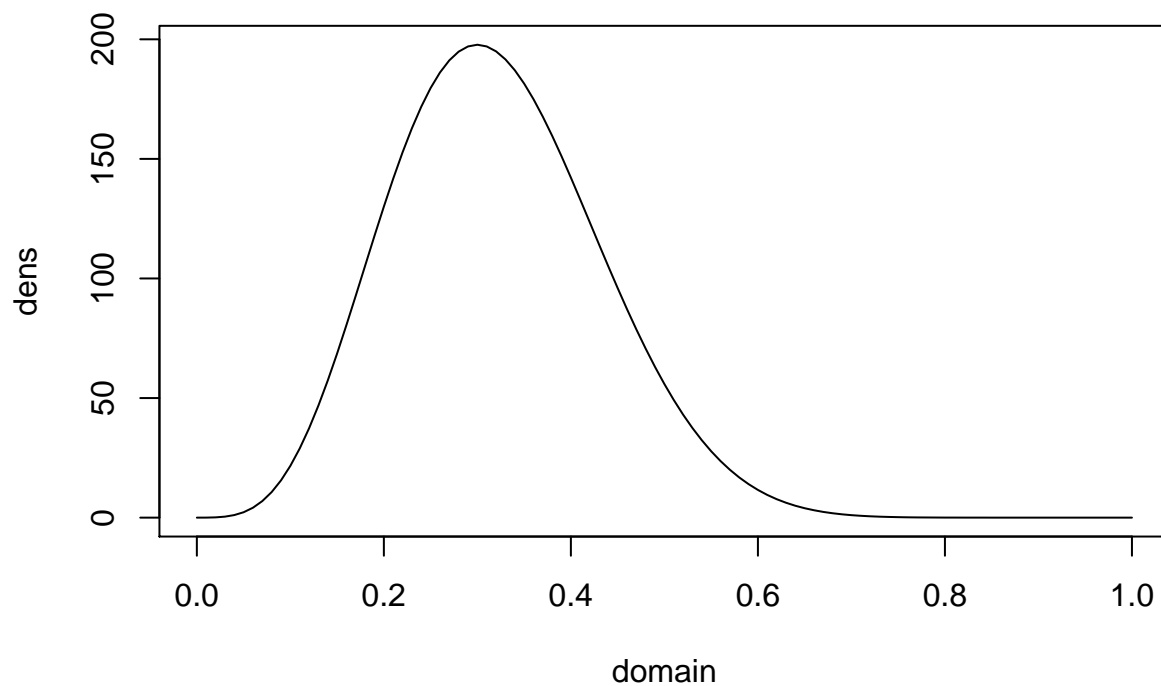
$$P(\theta|y < 3) \propto \sum_{i=0}^2 \text{Beta}(4+i, 4+(n-i))$$

The **right** way to answer the question would be:

$$\begin{aligned} P(y < 3|\theta) &= \sum_{i=0}^2 \text{Bin}(i|n, \theta) \\ \Rightarrow P(\theta|y) &\propto \sum_{i=0}^2 \binom{n}{i} \text{Beta}(4+i, 4+(n-i)) \end{aligned}$$

In this case some part of the proportionality constant *does* matter.

```
domain <- seq(0,1,.01)
dens = apply(sapply(0:2,function(x) choose(10,x)*dbeta(domain,4+x,4+10-x)),1,sum)
plot(domain, dens, type="l")
```



2.14

2.14a

Deriving the posterior for a normal likelihood with known variance, unknown mean, and using a normal prior.
[Slide 15 here](#)

Note: a good reminder of the main conjugacy relationships can be found [here](#)

2.16a [NEW]

Suppose $y \sim \text{Bin}(n, \theta)$, with $\theta \sim \text{Beta}(\alpha, \beta)$. Derive marginal distribution of y (unconditional on θ)

$$\begin{aligned}
P(y = k) &= \int_0^1 p(y, \theta) d\theta \\
&= \int_0^1 p(y|\theta)p(\theta) d\theta \\
&= \int_0^1 \binom{n}{k} \theta^k (1 - \theta)^{n-k} \times \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta^{\alpha-1} (1 - \theta)^{\beta-1} d\theta \\
&= \binom{n}{k} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \int_0^1 \theta^{k+\alpha-1} (1 - \theta)^{n-k+\beta-1} d\theta \\
&= \binom{n}{k} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \int_0^1 \frac{\Gamma(\alpha + k)\Gamma(\beta + n - k)}{\Gamma(\alpha + \beta + n)} \frac{\Gamma(\alpha + \beta + n)}{\Gamma(\alpha + k)\Gamma(\beta + n - k)} \theta^{k+\alpha-1} (1 - \theta)^{n-k+\beta-1} d\theta \\
&= \binom{n}{k} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \frac{\Gamma(\alpha + k)\Gamma(\beta + n - k)}{\Gamma(\alpha + \beta + n)} \int_0^1 \frac{\Gamma(\alpha + \beta + n)}{\Gamma(\alpha + k)\Gamma(\beta + n - k)} \theta^{k+\alpha-1} (1 - \theta)^{n-k+\beta-1} d\theta \\
&= \binom{n}{k} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \frac{\Gamma(\alpha + k)\Gamma(\beta + n - k)}{\Gamma(\alpha + \beta + n)}
\end{aligned}$$

Where the second to last line follows from integrating the density of a $\text{Beta}(k + \alpha, n - k + \beta)$

2.19 [NEW] Exponential model with conjugate prior distribution:

2.19a

Prove conjugacy for $y_i|\theta \sim \text{Exp}(\theta)$ for $i = 1, \dots, n$, iid, and $\theta \sim \text{Gamma}(\alpha, \beta)$.

$$\begin{aligned}
p(\theta|\mathbf{y}) &\propto p(\mathbf{y}|\theta)p(\theta) \\
&= p(\theta) \prod_{i=1}^n p(y_i|\theta) \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)} \theta^{\alpha-1} \exp(-\beta\theta) \times \prod_{i=1}^n \theta \exp(-\theta y_i) \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)} \theta^{\alpha-1} \exp(-\beta\theta) \times \prod_{i=1}^n \theta \exp(-\theta y_i) \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)} \times \theta^{\alpha+n-1} \exp(-\theta(\beta + \sum y_i)) \\
&\propto \frac{(\beta + \sum y_i)^{\alpha+n}}{\Gamma(\alpha + n)} \theta^{\alpha+n-1} \exp(-\theta(\beta + \sum y_i)) \\
&= \text{Gamma}(\alpha + n, \beta + \sum y_i)
\end{aligned}$$

2.19b

Show that the equivalent prior specification for the mean $\phi = 1/\theta$, is inverse-gamma.

Solution: We will use the transformation method. For this purpose we define: $\phi = g(\theta) = 1/\theta$, This implies that $g^{-1}(\phi) = 1/\phi$ and $\frac{d}{d\phi}g^{-1}(\phi) = -1/\phi^2$. Now:

$$\begin{aligned} f_{\phi}(\phi) &= f_{\theta}(g^{-1}(\phi)) \left| \frac{d}{d\phi}g^{-1}(\phi) \right| \\ &= \frac{\beta^{\alpha}}{\Gamma(\alpha)} \frac{1}{\phi^{(\alpha-1)}} \exp(-\beta/\phi) \frac{1}{\phi^2} \\ &= \frac{\beta^{\alpha}}{\Gamma(\alpha)} \phi^{-(\alpha+1)} \exp(-\beta/\phi) \\ &= \text{Inv-Gamma}(\phi) \end{aligned}$$

2.19c

The length of life (y_i) of a light bulb is distributed $\text{Exp}(\theta)$, with $\theta \sim \text{Gamma}(\alpha, \beta)$ and Coefficient of Variation ($\sqrt{\text{Var}(\theta)}/E(\theta)$) equal to 0.5. How many observation do we need to reduce the posterior CV to 0.1?

$$\begin{aligned} CV(\theta) &= \frac{\sqrt{\alpha}/\beta}{\alpha/\beta} = \frac{1}{\sqrt{\alpha}} = 0.5 \implies \alpha = 4 \\ CV(\theta|y) &= \frac{\sqrt{4+n}/(\beta + \sum y_i)}{(4+n)/(\beta + \sum y_i)} = \frac{1}{\sqrt{4+n}} = 0.1 \\ \implies n &= 96 \end{aligned}$$

2.19d

How would your answer change if the CV refers to ϕ instead of θ ? For this exercise we will use the fact that Inverse-Gamma distributions are conjugate with exponentials, where the resulting posterior has parameters $IG(\alpha + n, \beta + \sum y_i)$. Following the same steps as in (c), but for the mean and variance of the IG, we get $\alpha = 6$. Solving for the CV of the posterior we get $n = 96$.

Chapter 3

3.7 [NEW]

Show that the likelihood of two independent poisson $v \sim \text{Pois}(\theta_v), b \sim \text{Pois}(\theta_b)$ is the same as the likelihood of a $b|v+b \sim \text{Bin}(v+b, \frac{\theta_b}{\theta_b+\theta_v})$.

Solution: For this proof I follow [Blitztein & Hwang - Ch4 - p166-167](#):

- First, we get the distribution of $y = v + b$ (conditioning by b and using the law of total probability):

$$\begin{aligned}
P(y = k) &= P(v + b = k) = \sum_{j=1}^k P(v + b = k | b = j) P(b = j) \\
&= \sum_{j=1}^k P(v = k - j) P(b = j) \\
&= \sum_{j=1}^k \frac{e^{-\theta_v} \theta_v^{k-j}}{(k-j)!} \frac{e^{-\theta_b} \theta_b^j}{j!} \\
&= \sum_{j=1}^k \frac{e^{-(\theta_v + \theta_b)} \theta_v^{k-j} \theta_b^j}{(k-j)! j!} \frac{k!}{k!} \\
&= \frac{e^{-(\theta_v + \theta_b)}}{k!} \sum_{j=1}^k \binom{k}{j} \theta_v^{k-j} \theta_b^j \\
&= \frac{e^{-(\theta_v + \theta_b)} (\theta_v + \theta_b)^k}{k!} \\
&= \text{Pois}(\theta_v + \theta_b)
\end{aligned}$$

- Second, we obtain the distribution of $b | v + b = n$:

$$\begin{aligned}
P(b = k | v + b = n) &= \frac{P(v + b = n | b = k) P(b = k)}{P(v + b = n)} \\
&= \frac{P(v = n - k) P(b = k)}{P(v + b = n)} \quad \text{using previous result:} \\
&= \frac{\text{Pois}(n - k | \theta_v) \text{Pois}(k | \theta_b)}{\text{Pois}(n | \theta_v + \theta_b)} \\
&= \frac{\frac{e^{-\theta_v} \theta_v^{n-k}}{(n-k)!} \frac{e^{-\theta_b} \theta_b^k}{k!}}{\frac{e^{-(\theta_v + \theta_b)} (\theta_v + \theta_b)^n}{n!}} \\
&= \binom{n}{k} \frac{\theta_v^{n-k} \theta_b^k}{(\theta_v + \theta_b)^n} \\
&= \binom{n}{k} \left(\frac{\theta_b}{\theta_v + \theta_b} \right)^k \left(\frac{\theta_v}{\theta_v + \theta_b} \right)^{n-k} \\
&= \text{Bin} \left(n = b + v, \frac{\theta_b}{\theta_v + \theta_b} \right)
\end{aligned}$$

Chapter 5

5.3 Reproducing results of section 5.5

```

#Data:
school.id    <- LETTERS[1:8]
effect       <- c(28,8,-3,7,-1,1,18,12)
se.effect    <- c(15,10,16,11,9,11,10,18)

```



```
pool.est      <- sum(effect*se.effect^-2)/sum(se.effect^-2)
pool.var      <- sum(se.effect^-2)^-1
pool.ci       <- c(-1.96,1.96)*pool.var^.5 + pool.est
```

The pooled estimated effect and variance are 7.69 and 16.58, with a 95% CI of [-0.3, 15.67].

Posterior simulation under the hierarchical model

Using the identity:

$$p(\theta, \mu, \tau | y) = p(\tau | y) p(\mu | \tau, y) p(\theta | \mu, \tau, y)$$

And the results from BDA in equation 5.17, 5.20 and 5.21 we code the posteriors $p(\theta_j | \tau, \mu, y)$, $p(\mu | \tau, y)$ and $p(\tau | y)$. **Important note:** for this exercise we could have follow *exactly* the same steps as the last exercise. Given some properties of the N-N model, we took a different path and derive an analytic formula for $p(\mu | \tau, y)$ and $p(\tau | y)$ (instead of sampling from the joint posterior density).

```
# Eqn 5.17 of BDA3
post.theta.j <- function(mu,tau,j) {
  ( effect[j] / ( se.effect[j]^2 ) + mu / ( tau^2 ) ) /
  ( 1 / ( se.effect[j]^2 ) + 1 / ( tau^2 ) )
}

post.v.theta.j <- function(tau,j) {
  1 / ( 1 / ( se.effect[j]^2 ) + 1 / ( tau^2 ) )
}

# Eqn 5.20 of BDA3
post.mu.hat <- function(tau) {
  sum( effect * 1 / ( se.effect^2 + tau^2 ) ) /
  sum( 1 / ( se.effect^2 + tau^2 ) )
}

post.v.mu <- function(tau) {
  ( sum( 1 / ( se.effect^2 + tau^2 ) ) )^-1
}

# Eqn 5.21 of BDA3
marginal.tau <- function(tau) {
  hyper.prior(tau) *
  ( post.v.mu(tau)^.5 ) *
  prod( ( ( se.effect^2 + tau^2 )^(-.5) ) *
        exp( - ( ( effect - post.mu.hat( tau ) )^2 )
              / ( 2 * ( se.effect^2 + tau^2 ) ) ) )
}
```

Define a hyper-prior and draw 200 samples from each distribution (for all 8 schools).

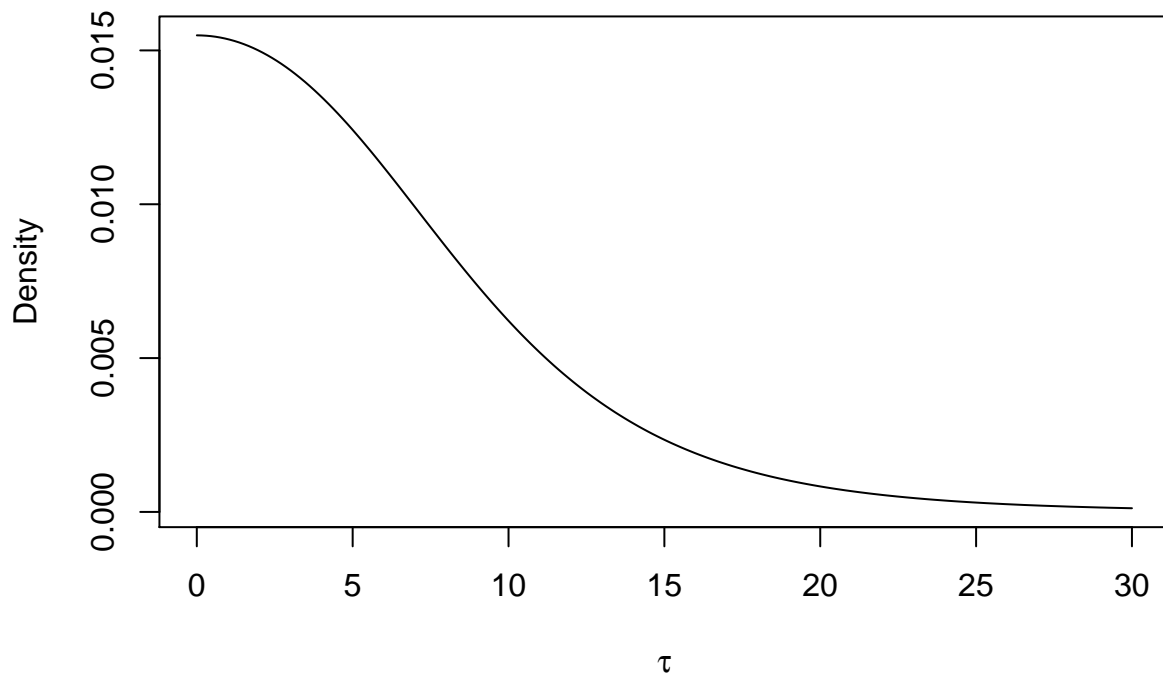
```
samps <- 200

hyper.prior <- function(tau) 1
tau.grid <- seq(0.001,30, length=samps)
pdf.tau <- sapply(tau.grid,function(x) marginal.tau(x))
pdf.tau <- pdf.tau/sum(pdf.tau)

plot(tau.grid,pdf.tau, type="l",
```

```
main="Figure 5.5 from BDA3",
xlab=expression(tau),
ylab="Density")
```

Figure 5.5 from BDA3



The sampling method in BDA3 suggest to apply the inverse method from the posterior of τ . I don't do this for two reasons: (i) I'm not sure the posterior has a closed for solution for its inverse, and (ii) given that I already have the density, I can directly draw from that distribution sampling using the `sample` command (which leads me to think that this command applies the inverse method).

```
# Sampling
s.tau      <- sample(tau.grid,samps,prob=pdf.tau, replace=TRUE)
s.mu       <- sapply(s.tau,function(x) rnorm(1,post.mu.hat(x),(post.v.mu(x))^0.5))
s.theta    <- NULL
for (j in 1:length(school.id)) {
  s.theta[[j]] <- sapply(1:samps,
    function(x)
      rnorm(1,
        post.theta.j(s.mu[x],s.tau[x],j),
        (post.v.theta.j(s.tau[x],j))^0.5
      )
  )
}
```

The following figures replicate the figures in pg 122 in BDA. Before doing the plots we need to 'average over μ '

$$\begin{aligned}
E(\theta_j|\tau, y) &= E_\mu [E(\theta_j|\tau, y, \mu)|\tau, y] \\
&= E_\mu \left[\frac{\frac{1}{\sigma_j^2} y_j + \frac{1}{\tau^2} \mu}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}} | \tau, y \right] = \frac{\frac{1}{\sigma_j^2} y_j + \frac{1}{\tau^2} \hat{\mu}}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}}
\end{aligned}$$

$$\begin{aligned}
Var(\theta_j|\tau, y) &= E_\mu [Var(\theta_j|\tau, y, \mu)|\tau, y] + Var_\mu [E(\theta_j|\tau, y, \mu)|\tau, y] \\
&= \frac{1}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}} + V_\mu \left(\frac{\frac{1}{\tau^2}}{\frac{1}{\sigma_j^2} + \frac{1}{\tau^2}} \right)
\end{aligned}$$

Where V_μ and $\hat{\mu}$ correspond to the expressions defined in Eq 5.20 of BDA3. Below is the code and plot of both equations.

```

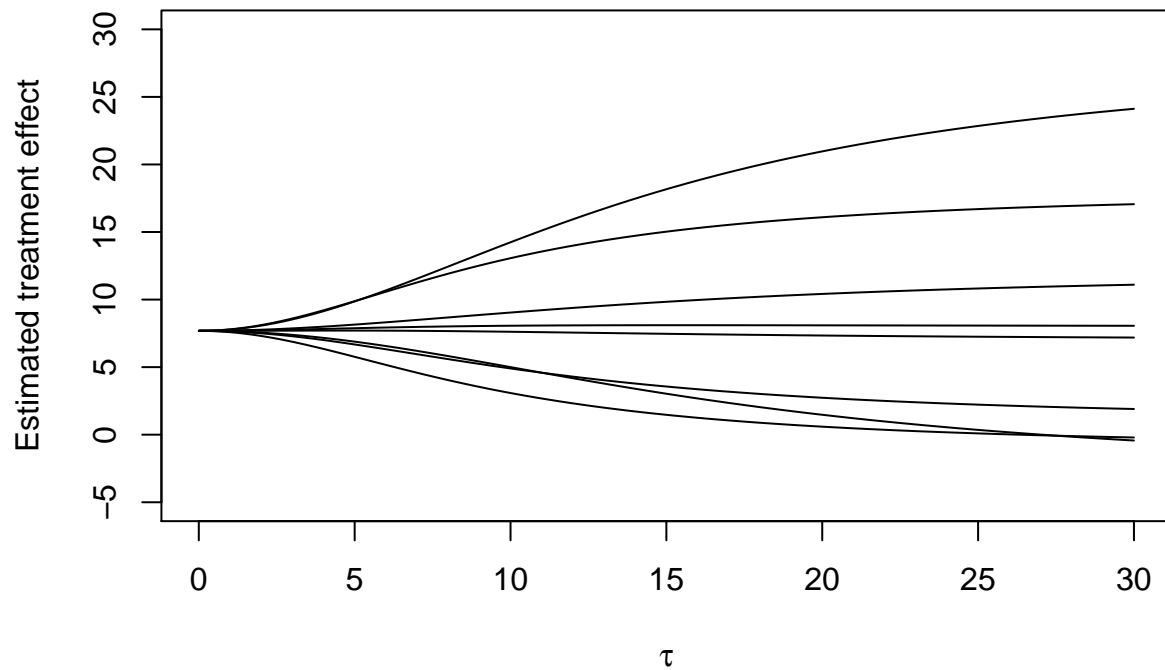
post.theta.j.no.mu    <- function(tau,j) post.theta.j( post.mu.hat( tau ), tau, j )
post.se.theta.j.no.mu <- function(tau,j) sqrt(
  ( post.v.theta.j( tau, j ) ) * ( 1 + post.v.mu( tau ) * tau^(-2) ) )

plot( tau.grid, sapply(tau.grid, function(x) post.theta.j.no.mu(x,1) ),
      type="l", ylim=c(-5,30),
      xlab="", ylab="")
title(main="Figure 5.6 from BDA3",
      xlab=expression(tau),
      ylab="Estimated treatment effect")

for (j in 2:8) {
  lines( tau.grid,
        sapply( tau.grid, function(x) post.theta.j.no.mu(x,j) ) )
}

```

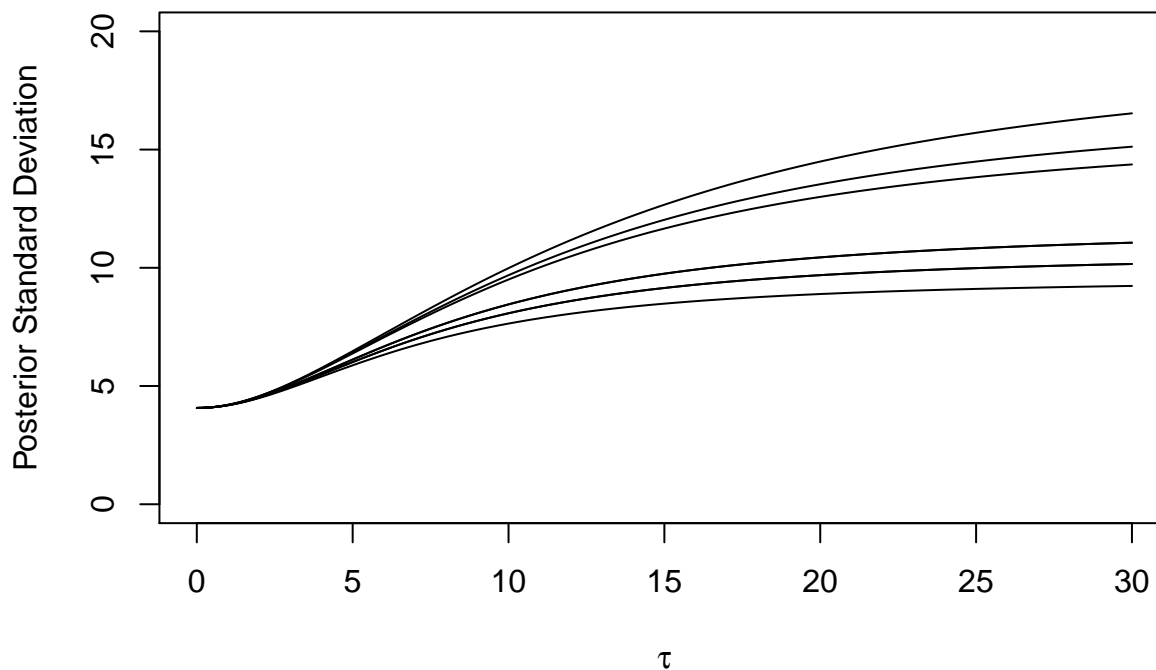
Figure 5.6 from BDA3



```
plot( tau.grid, sapply(tau.grid, function(x) post.se.theta.j.no.mu(x,1) ),
      type="l", ylim=c(0,20),
      xlab="", ylab="")
title(main="Figure 5.7 from BDA3",
      xlab=expression(tau),
      ylab="Posterior Standard Deviation")

for (j in 2:8) {
  lines( tau.grid,
        sapply( tau.grid, function(x) post.se.theta.j.no.mu(x,j) ) )
}
```

Figure 5.7 from BDA3



```
s.theta      <- matrix(unlist(s.theta), ncol = 8, byrow = FALSE)
s.theta.sort <- apply(s.theta, 2, sort)
p            <- t( apply( s.theta.sort, 2, function(x)
                        quantile( x, c( .025, .25, .5, .75, .975), type=1 ) ) ) )
p            <- round(p,3)
#Bug in line below
#knitr::kable(p, caption="Misclassification and Reliability for Different Thresholds")
```

Table 5.3 from BDA3:

Now we can use the simulated θ_s to described the estimated effects in each school.

School	2.5%	25%	median	75%	97.5%
A	-1.287	6.127	9.702	14.719	32.959
B	-5.205	3.904	7.958	11.727	22.622
C	-11.529	1.932	7.345	11.119	21.305
D	-4.456	3.535	7.369	11.092	19.771
E	-11.326	0.956	6.303	9.364	18.245
F	-5.954	2.578	6.999	10.396	17.554
G	-0.544	7.096	10.007	14.477	26.717
H	-6.289	3.54	7.725	11.798	22.223

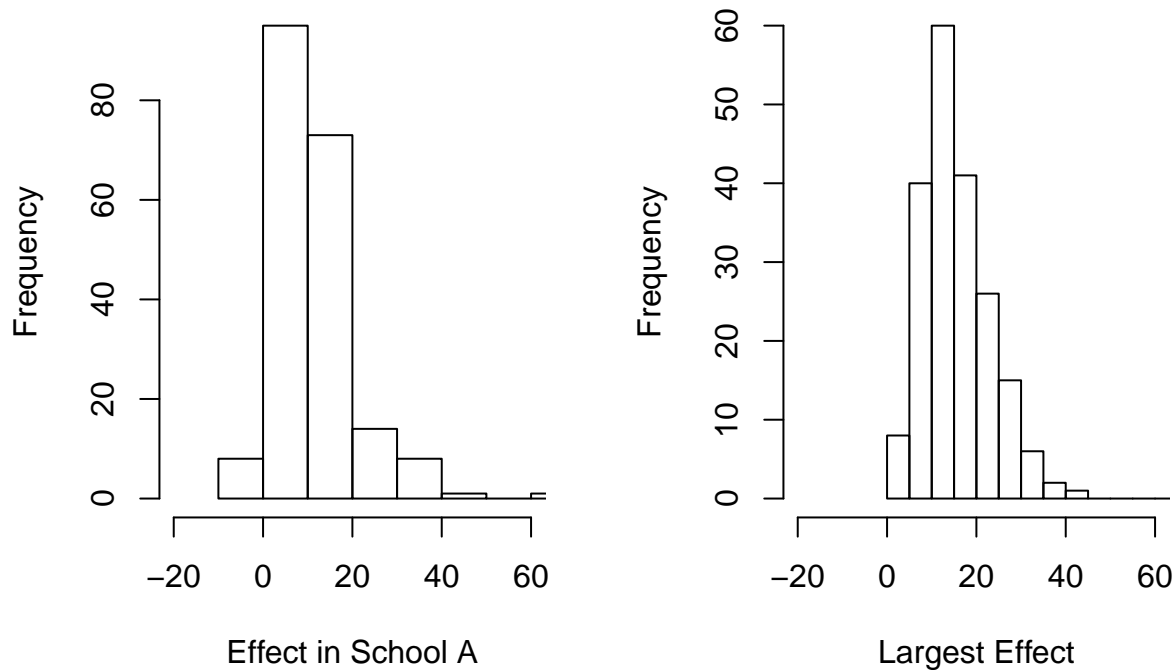
Here we reproduce figure 5.8

```

par(mfrow=c(1,2))
domain      <- c(-20,60)
hist(s.theta[,1], breaks=10, xlab="Effect in School A", main="", xlim=domain)
hist(apply(s.theta,1,max), breaks=10, xlab="Largest Effect", main="", xlim=domain)
title(main="Figure 5.8 from BDA3")

```

Figure 5.8 from BDA3



This last figure (“largest effect”) is a good example of one the main advantage of a fully Bayesian hierarchical model: once we have simulated the posterior, we can test all kinds of complicated hypothesis.

5.3a

- (i) For each school j , the probability that its coaching program is the best of eight: (**Important:** do not sort each posterior).

```

aux1      <- apply(s.theta,1,max)
best      <- apply(1*(s.theta==aux1), 2,mean)

```

Table 2: Probability that each coaching program is the best among the eight schools

School	Probability of having the best coaching program
A	0.275
B	0.11
C	0.055
D	0.09

School	Probability of having the best coaching program
E	0.06
F	0.075
G	0.23
H	0.105

- (ii) For each school j , the probability that its coaching program is better than other school k :

```
p
      <- sapply( 1:8,
                 function(y) sapply( 1:8,
                                     function(x)
                                       mean( 1 * ( s.theta[,x] > s.theta[,y] ) )
                                     )
                 )
```

Table 3: Probability that school j (row) has a better program than school k (column)

School j /School k									
	A	B	C	D	E	F	G	H	
A	0	0.64	0.7	0.655	0.725	0.69	0.49	0.625	
B	0.36	0	0.565	0.54	0.66	0.58	0.365	0.475	
C	0.3	0.435	0	0.465	0.555	0.48	0.3	0.42	
D	0.345	0.46	0.535	0	0.62	0.545	0.355	0.49	
E	0.275	0.34	0.445	0.38	0	0.425	0.275	0.385	
F	0.31	0.42	0.52	0.455	0.575	0	0.295	0.43	
G	0.51	0.635	0.7	0.645	0.725	0.705	0	0.625	
H	0.375	0.525	0.58	0.51	0.615	0.57	0.375	0	

5.3b

- (i) Now with $\tau = \infty$ compute for each school j , the probability that it has the best coaching program: With $\tau = \infty$ each school posterior effect is independent $\theta_j \sim N(y_j, \sigma_j^2)$. The probability of a school having the best coaching program is:

Wrong way to do it:

$$p(\theta_j > \max_{i \neq j} \{\theta_i\}) = \prod_{i \neq j} p(\theta_j > \theta_i) \\ = \prod_{i \neq j} \Phi\left(\frac{\theta_j - \theta_i}{\sigma_i}\right)$$

Right way to do it:

$$p(\theta_j > \max_{i \neq j} \{\theta_i\}) = \int \prod_{i \neq j} p(\theta_j > \theta_i) \phi(\theta_j | y_j, \sigma_j) d\theta_j \\ = \int \prod_{i \neq j} \Phi\left(\frac{\theta_j - \theta_i}{\sigma_i}\right) \phi(\theta_j | y_j, \sigma_j) d\theta_j$$

This integral has to be solved numerically:

```
set.seed(142857)
best <- sapply(1:8,
  function(y) mean( sapply( 1:1000 ,
    function(x)
      prod( pnorm( (
        rnorm( 1 , effect[y] , se.effect[y] ) - effect[-y] ) /
        se.effect[-y] ) ) )
    )
  )
# Ad-hoc normalization:
best <- best/sum(best)
```

Table 4: Probability that each coaching program is the best among the eight schools (with $\tau = \infty$)

School	Probability of having the best coaching program
A	0.559927
B	0.0329844
C	0.0265477
D	0.0364362
E	0.0034413
F	0.0136409
G	0.1614563
H	0.1655662

- (ii) Now with $\tau = \infty$ compute for each school j , the probability that its coaching program is the better than other school k :

$$p(\theta_i > \theta_j) = p\left(-\frac{y_j - y_i}{\sqrt{\sigma_i^2 + \sigma_j^2}} > \frac{(\theta_j - \theta_i) - (y_j - y_i)}{\sqrt{\sigma_i^2 + \sigma_j^2}}\right)$$

$$= \Phi\left(\frac{y_i - y_j}{\sqrt{\sigma_i^2 + \sigma_j^2}}\right)$$

The following table presents the different values for the expression above:

```
p      <- supply(1:8,function(x)
      supply(1:8,function(y)
        pnorm( q = 0, mean = (effect[x] - effect[y]) / sqrt(se.effect[x]^2 + se.effect[y]^2)
        sd = 1 )
      ) )
# Force all elementens in the diagonal to zero.
p      <- p - .5 * diag(8)
```

Table 5: Probability that j (row) has a better program than school k (column). With $\tau = \infty$

School j /School k	A	B	C	D	E	F	G	H
A	0	0.8663713	0.9212424	0.8705441	0.9513231	0.9266837	0.7104501	0.7526534
B	0.1336287	0	0.720053	0.5268155	0.748241	0.6811336	0.2397501	0.4229873
C	0.0787576	0.279947	0	0.3032674	0.4566223	0.4183914	0.1328547	0.2666945
D	0.1294559	0.4731845	0.6967326	0	0.713241	0.6501386	0.2296682	0.4063196
E	0.0486769	0.251759	0.5433777	0.286759	0	0.4440458	0.0789369	0.2591477
F	0.0733163	0.3188664	0.5816086	0.3498614	0.5559542	0	0.1264065	0.3010267
G	0.2895499	0.7602499	0.8671453	0.7703318	0.9210631	0.8735935	0	0.6146218
H	0.2473466	0.5770127	0.7333055	0.5936804	0.7408523	0.6989733	0.3853782	0

5.3c

The estimated differences between the closed form solutions (5.3b) and the bayesian analysis (5.3a) is that the latter presents less extreme probability estimates (shrinkage)

5.3d

If $\tau = 0$, then all effects are the same so the probabilities can be 0 or 1 for all schools (all are the largest effect and the smallest at the same time)

5.13 - Bicycles

#Load data

```
y <- c(16, 9, 10, 13, 19, 20, 18, 17, 35, 55)
n <- c(74, 99, 58, 70, 122, 77, 104, 129, 308, 119)
```

5.13a

$y_i \sim \text{Bin}(\theta_i, n_i)$ where n_i represents the *total* number of vehicles (bicycles + other vehicles). $\theta_i \sim \text{Beta}(\alpha, \beta)$ the prior distribution of biking rates for each street. We set a non-informative hyper-prior $p(\alpha, \beta) \propto (\alpha + \beta)^{-5/2}$. This implies that the **joint posterior** distribution has the following (same as in equation 5.6 in BDA3):

$$p(\theta, \alpha, \beta | y) \propto p(\alpha, \beta) p(\theta | \alpha, \beta) p(y | \theta, \alpha, \beta)$$

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

5.13b

Compute the marginal posterior of θ , conditional on α, β . For the beta-binomial case we have that given the hyper-parameters, each θ_j has a posterior distribution $\text{Beta}(\alpha + y_j, \beta + n_j - y_j)$. Assuming exchangeability:

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

Now we compute the posterior marginal of (α, β) . Given that we do have a closed form solution in step 2, we compute the ratio of (`\ref{bic.joint.post1}`) and (`\ref{bic.cond.post.theta1}`).

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

Centering our grid around the methods of moments estimates for (α_0, β_0) :

$$\hat{\mu} = 0.1961412 = \frac{\hat{\alpha}_0}{\hat{\alpha}_0 + \hat{\beta}_0}$$

$$\hat{\sigma}^2 = 0.0111207 = \frac{\hat{\alpha}_0 \hat{\beta}_0}{(\hat{\alpha}_0 + \hat{\beta}_0)^2 (\hat{\alpha}_0 + \hat{\beta}_0 + 1)}$$

Solving form $(\hat{\alpha}_0, \hat{\beta}_0)$:

```

#Here 'x' represents alpha and beta
dslnex      <- function(x) {
  z          <- numeric(2)
  z[1]       <- x[1]/(x[1]+x[2]) - mean(y/n)
  z[2]       <- x[1]*x[2]/(((x[1]+x[2])^2)*(x[1]+x[2]+1)) - sd(y/n)^2
  z
}

sol1        <- nleqslv(c(1,1), dslnex)
res1        <- paste("(",round(sol1$x[1],1), ",", round(sol1$x[2],1), ")",sep="")

```

We get: $(\hat{\alpha}_0, \hat{\beta}_0) = (2.6, 10.6)$.

We center the grid (approximately) around that initial estimate and expand the grid to cover up to a factor of 4 of each parameter. The result is plotted in the following figure:

```

bic.marg.post.phi <- function(alpha, beta) {
  post          <- 1
  #notice the censoring in n (the gamma(.) function in R cannot handle large values)
  for (i in 1:length(y)) {
    if (n[i] > 100) n[i] = 100
    post = post * (
      ( gamma(alpha + beta) ) /
      ( gamma(alpha) * gamma(beta) ) ) *
      ( gamma(alpha + y[i]) * gamma(beta + n[i] - y[i]) ) /
      ( gamma(alpha + beta + n[i]) ) )
  }
  # The hyper prior is defined below
  bic.hyper.prior(alpha,beta) * post
}

bic.hyper.prior <- function(alpha,beta)
{
  alpha*beta*(alpha + beta)^(-5/2)
}

v1          <- seq(log(sol1$x[1]/sol1$x[2])*1.5,
                  log(sol1$x[1]/sol1$x[2])/1.5,length.out =151)
v2          <- seq(log(sol1$x[1]+sol1$x[2])/1.5,
                  log(sol1$x[1]+sol1$x[2])*1.5,length.out =151)
beta        <- exp(v2)/(exp(v1)+1)
alpha       <- exp(v2+v1)/(exp(v1)+1)

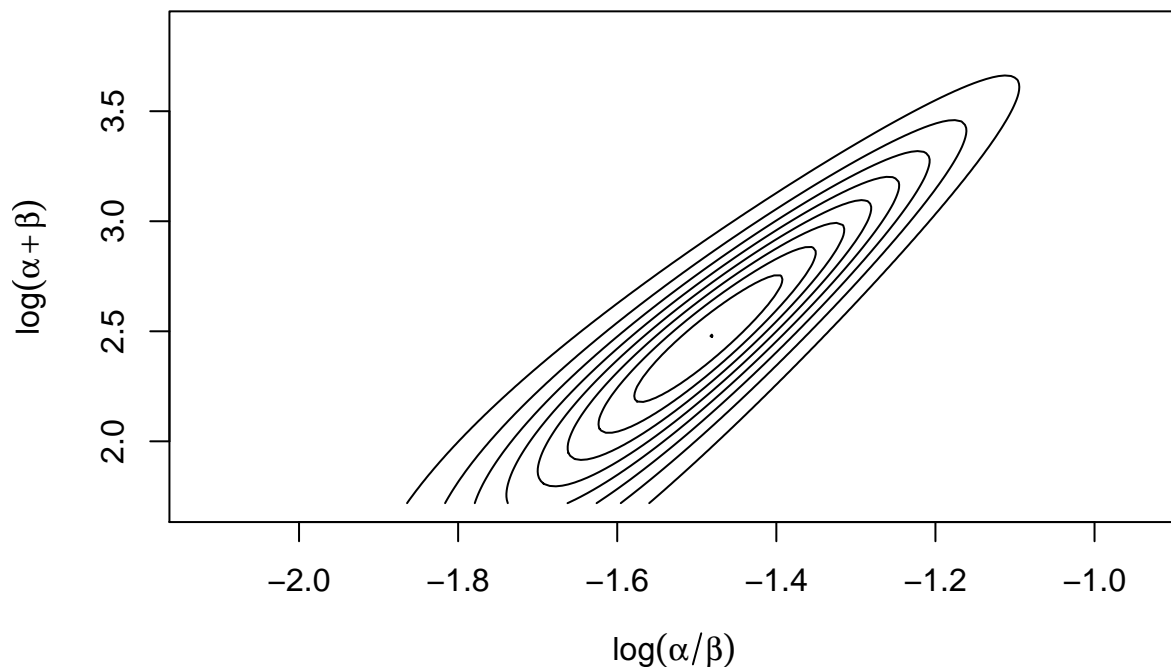
post.dens   <- outer(alpha,beta,function(x1,x2) log(bic.marg.post.phi(x1, x2)) )
post.dens   <- exp(post.dens - max(post.dens))
post.dens   <- post.dens/sum(post.dens)

contours    <- seq(min(post.dens), max(post.dens) , length=10)
contour(v1, v2, post.dens,
        levels=contours,
        xlab=expression( log(alpha/beta) ),
        ylab=expression( log(alpha+beta) ),
        xlim=c( min( v1 ), max( v1 ) ) ,

```

```
ylim=c( min( v2 ), max( v2 ) ),
drawlabels=FALSE,
main="Contour plot of joint posterior")
```

Contour plot of joint posterior



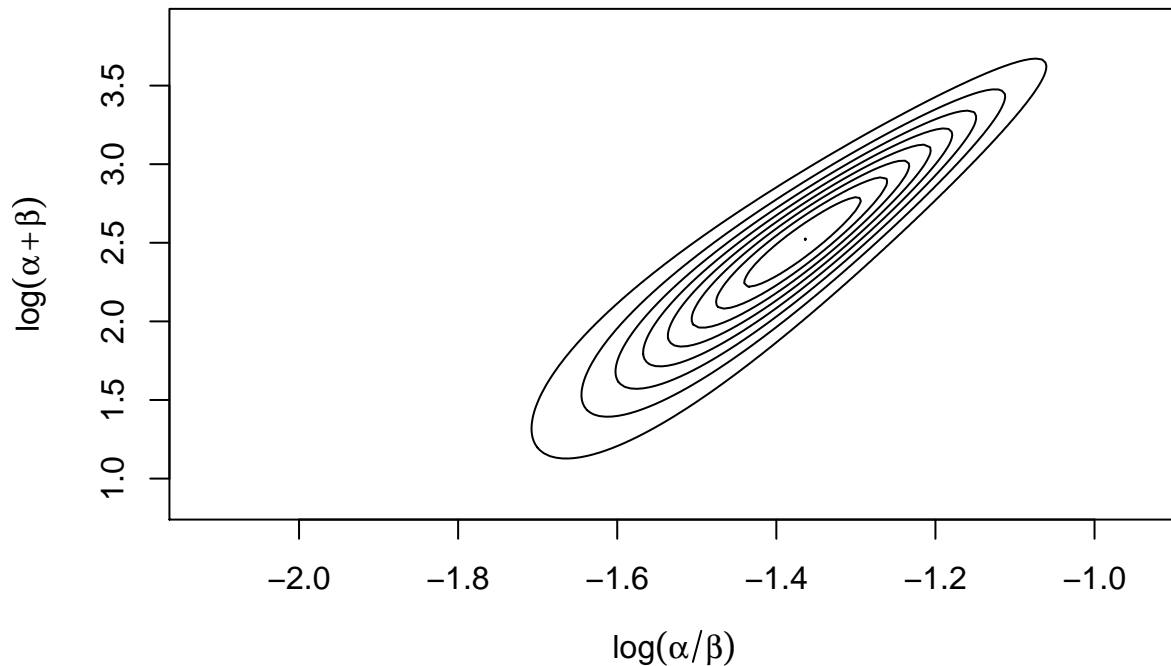
Adjust the grid and repeat:

```
v1      <- seq(log(sol1$x[1]/sol1$x[2])*1.5,
               log(sol1$x[1]/sol1$x[2])/1.5,length.out =151)
v2      <- seq(log(sol1$x[1]+sol1$x[2])/3,
               log(sol1$x[1]+sol1$x[2])*1.5,length.out =151)
beta    <- exp(v2)/(exp(v1)+1)
alpha   <- exp(v2+v1)/(exp(v1)+1)

post.dens <- outer(alpha,beta,function(x1,x2) log(bic.marg.post.phi(x1, x2)) )
post.dens <- exp(post.dens - max(post.dens))
post.dens <- post.dens/sum(post.dens)

contours <- seq(min(post.dens), max(post.dens) , length=10)
contour(v1, v2, post.dens,
        levels=contours,
        xlab=expression( log(alpha/beta) ),
        ylab=expression( log(alpha+beta) ),
        xlim=c( min( v1 ), max( v1 ) ),
        ylim=c( min( v2 ), max( v2 ) ),
        drawlabels=FALSE,
        main="Contour plot of joint posterior")
```

Contour plot of joint posterior



Draw samples (α^s, β^s) from $p(\alpha, \beta|y)$ (finally!). Here we repeat the procedure used in section 3.(v) of the book replication document.

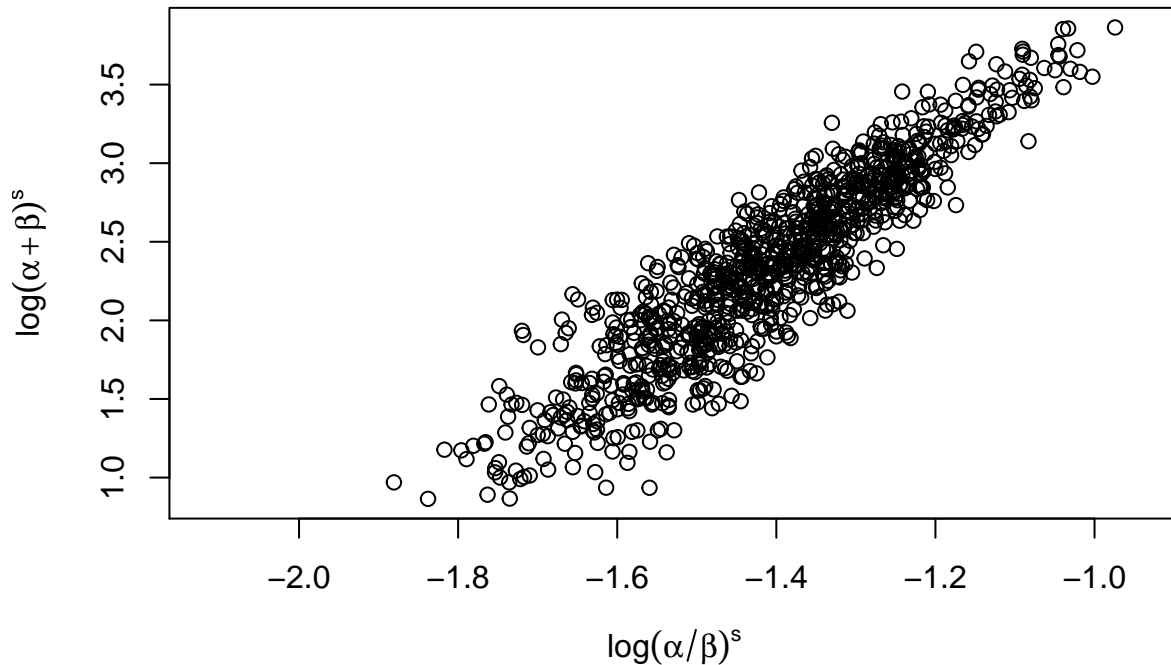
```
samps      <- 1000
#Integrate (sum) over all beta to get the marginal of alpha
v1.dens    <- apply(post.dens ,1, sum)
s.v1       <- sample(v1, samps, replace=TRUE, prob = v1.dens)

# Select the colum of the joint density corresponding to a specific value of v1 (p(v2/v1))
cond.v2    <- function(x)
{
  post.dens[which(v1 == s.v1[x]),]
}
# Sample a value of v2 according the the conditional probatibility above
s.v2       <- sapply(1:samps,function(x) sample(v2,1,replace=TRUE,prob=cond.v2(x)))

# Add a uniform random jitter centered at zero with with equal to the grid spacing.
# This will make the simulation draws more continuous. Plot the sampled values.
grid.v1    <- v1[2] - v1[1]
grid.v2    <- v2[2] - v2[1]
s.v2       <- s.v2 + runif(length(s.v2),-grid.v2/2,grid.v2/2)
s.v1       <- s.v1 + runif(length(s.v1),-grid.v1/2,grid.v1/2)
plot(s.v1, s.v2,
     xlab=expression( log(alpha/beta)^s ),
```

```
ylab=expression( log(alpha+beta)^s ),
xlim=c( min(v1) , max(v1) ) ,
ylim=c( min(v2) , max(v2) ),
main="Scatter Plot of Sample Draws of log(alpha/beta) and log(alpha+beta)"
```

Scatter Plot of Sample Draws of $\log(\alpha/\beta)$ and $\log(\alpha+\beta)$



By applying the inverse of the transformation we recover the marginal distribution of the original hyper-parameters.

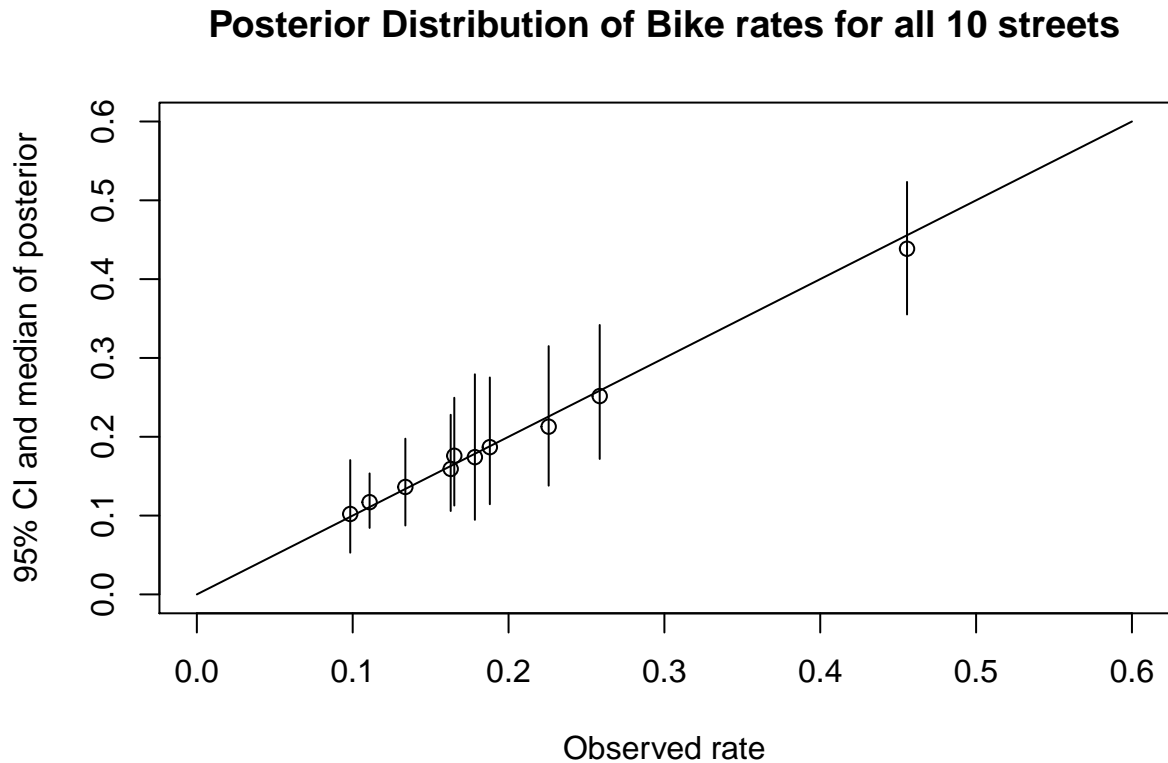
```
s.beta      <- exp( s.v2 ) / ( exp(s.v1)+1 )
s.alpha     <- exp( s.v2 + s.v1 ) / ( exp(s.v1)+1 )
```

5.13c

For each draw of ϕ^s , draw a sample of θ from $p(\theta|\phi^s, y)$

```
theta.dist  <- sapply(1:10,
                      function(x)
                        rbeta(1000, s.alpha+y[x], s.beta + n[x] - y[x])
                      )
theta.dist  <- apply(theta.dist,2,sort)
plot(0:600/1000, 0:600/1000,
     type="l",
     xlab="Observed rate",
     ylab="95% CI and median of posterior")
```

```
jitter.x      <- y/n + runif(length(y),-0.01,0.01)
points(jitter.x, theta.dist[500,])
segments(jitter.x,theta.dist[25,], jitter.x,theta.dist[975,] )
title(main="Posterior Distribution of Bike rates for all 10 streets")
```



The estimated proportions are almost the same as the raw proportions (no shrinkage).

5.13d

We generate 1000 draws from a $Beta(\alpha^s, \beta^s)$ where the parameters come from the draws obtained above:

```
s.theta      <- rbeta(1000, shape1 =s.alpha , shape2 = s.beta)
CI.num       <- round(s.theta[order(s.theta)][c(25,975)],2)
CI.str       <- paste("'", CI.num[1] , "," , CI.num[2] , "'")
```

The posterior interval for $\hat{\theta} = (0.01, 0.48)$

5.13e

If a new street is opening with 100 vehicles per day. The posterior interval predicts with 95% confidence that between 1 and 48. This CI is not so informative as it covers almost all the possible observed bike rates.

5.13f

The beta assumption might not have been so reasonable as the posterior estimates did not show much shrinkage.

5.14

5.14a

Set up a model in which the total number of vehicles observed at each location j follows a Poisson distribution with parameter θ_j , the ‘true’ rate of traffic per hour at the location. Assign a gamma population distribution for the parameters θ_j and a non-informative hyper-prior distribution. Write down the joint posterior distribution.

Now we have that $n_j \sim Poi(\lambda = \theta_j)$ and $\theta_j \sim Gamma(\alpha)$. And the joint posterior is:

$$\begin{aligned} p(\theta, \alpha, \beta | y) &\propto p(\alpha, \beta) \times p(\theta | \alpha, \beta) \times p(y | \theta, \alpha, \beta) \\ p(\theta, \alpha, \beta | y) &\propto 1 \times \prod_{j=1}^{10} Gamma(\theta_j | \alpha, \beta) \times \prod_{j=1}^{10} Poisson(y_j | \theta_j) \\ &= \prod_{j=1}^{10} \frac{\beta^\alpha}{\Gamma(\alpha)} \theta_j^{\alpha-1} \exp(-\beta \theta_j) \times \frac{\theta_j^{y_j} \exp(-\theta_j)}{y_j!} \\ &\propto \frac{\beta^{n\alpha}}{\Gamma(\alpha)^n} \exp(-\sum \theta_j (1 + \beta)) \prod_{j=1}^{10} \theta_j^{\alpha+y_j-1} \end{aligned}$$

5.14b

Then compute the marginal posterior of θ , conditional on α, β . For the gamma-poisson case we have that given the hyper-parameters, each θ_j has a posterior distribution $Gamma(\alpha + n_j, \beta + 1)$. Assuming exchangeability:

$$\begin{aligned} p(\theta | \alpha, \beta, y) &\propto \prod_{j=1}^{10} Gamma(\theta_j | \alpha + y_j, \beta + 1) \\ &\propto \prod_{j=1}^{10} \frac{\beta^{\alpha+y_j}}{\Gamma(\alpha+y_j)} \theta_j^{\alpha+y_j-1} \exp(-(\beta+1)\theta_j) \\ &\propto \prod_{j=1}^{10} \theta_j^{\alpha+y_j-1} \exp(-(\beta+1)\theta_j) \end{aligned}$$

Now we compute the posterior marginal of (α, β) . Given that we do have a closed form solution in step 2, we compute the ratio of (\ref{bic.joint.post2}) and (\ref{bic.cond.post.theta2}).

$$p(\alpha, \beta | y) \propto \frac{\beta^{n\alpha}}{\Gamma(\alpha)^n} \prod_{i=1}^n \frac{\Gamma(\alpha + y_i)}{(\beta + 1)^{\alpha+y_i}}$$

Centering our grid around the methods of moments estimates for (α_0, β_0) :

$$\hat{\mu} = 116 = \frac{\hat{\alpha}_0}{\hat{\beta}_0}$$

$$\hat{\sigma}^2 = 5141.7777778 = \frac{\hat{\alpha}_0}{\hat{\beta}_0^2}$$

Solving for $(\hat{\alpha}_0, \hat{\beta}_0)$:

```
#Here 'x' represents alpha and beta
dslnex      <- function(x) {
  z          <- numeric(2)
  z[1]       <- x[1]/(x[2]) - mean(n)
  z[2]       <- x[1]/(x[2]^2) - sd(n)^2
  z
}

sol1        <- nleqslv(c(1,1), dslnex)
res1        <- paste("(",round(sol1$x[1],1), ",", round(sol1$x[2],2), ")",sep="")
```

We get: $(\hat{\alpha}_0, \hat{\beta}_0) = (2.6, 0.02)$.

We center the grid (approximately) around that initial estimate and expand the grid to cover up to a factor of 4 of each parameter. The result is plotted in the following figure:

```
bic.marg.post.phi <- function(alpha, beta) {
  log.post        <- 0
  #notice the censoring in n
  for (i in 1:length(n))
  {
    if (n[i] > 100) n[i] <- 100
    log.post       <- log.post + log(gamma( alpha+n[i] )) - (alpha+n[i])*log((beta + 1))
  }
  # The hyper prior is defined below
  log(bic.hyper.prior2(alpha,beta)) +
  log.post + (length(n)*alpha)*log(beta) -
  length(n)*log(gamma(alpha))
}

bic.hyper.prior2 <- function(alpha,beta) {
  1
}

alpha          <- seq(sol1$x[1]/1.5,sol1$x[1]*1.5,length.out =151)
beta           <- seq(sol1$x[2]/1.5,sol1$x[2]*1.5,length.out =151)

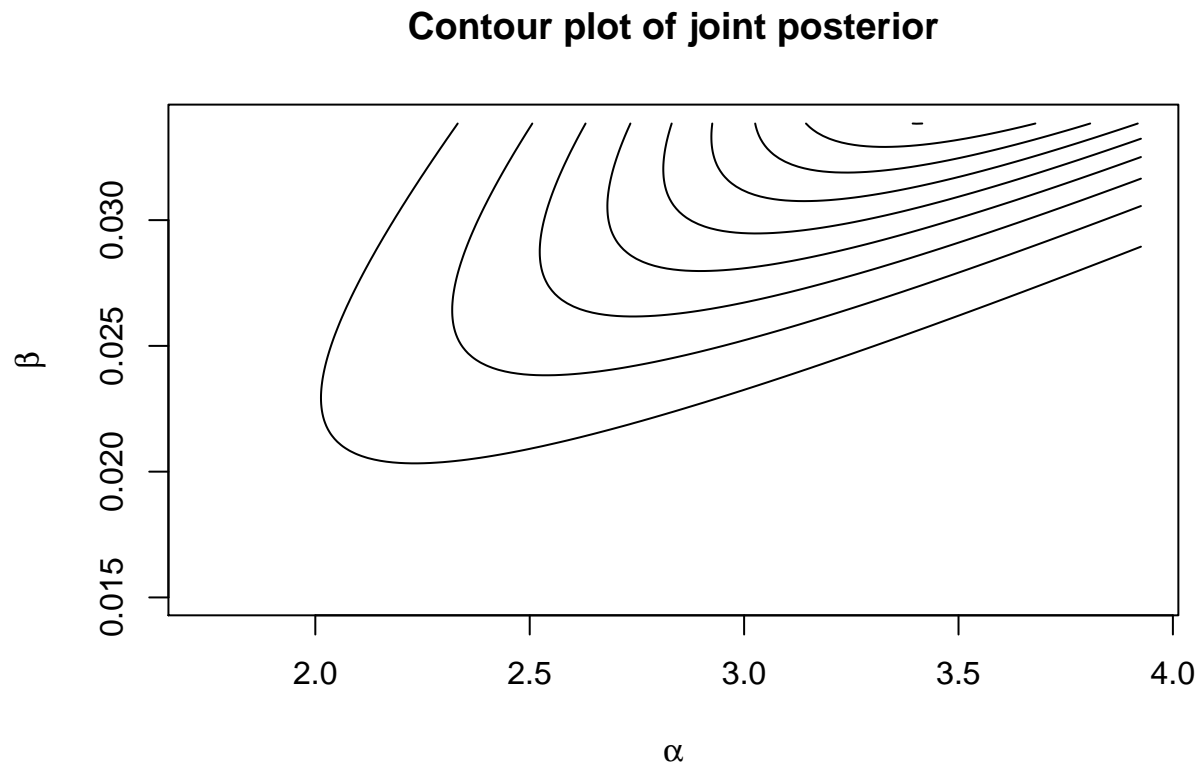
post.dens      <- outer(alpha,beta,function(x1,x2) bic.marg.post.phi(x1, x2) )
post.dens      <- exp(post.dens - max(post.dens))
post.dens      <- post.dens/sum(post.dens)

contours       <- seq(min(post.dens), max(post.dens), length=10)
contour(alpha, beta, post.dens,levels=contours,
```

```

xlab=expression(alpha),
ylab=expression(beta),
xlim=c(min(alpha),max(alpha)) ,
ylim=c(min(beta),max(beta)),
drawlabels=FALSE,
main="Contour plot of joint posterior")

```



Adjust the grid and repeat:

```

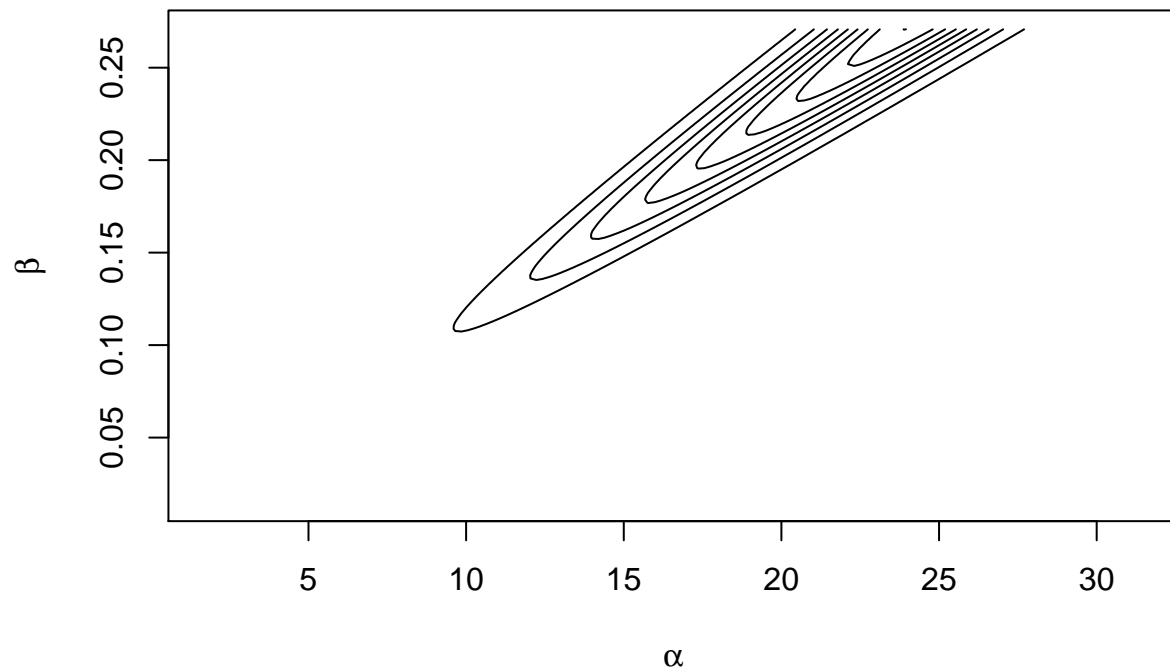
alpha      <- seq(sol1$x[1]/1.5,sol1$x[1]*12,length.out =151)
beta       <- seq(sol1$x[2]/1.5,sol1$x[2]*12,length.out =151)

post.dens  <- outer(alpha,beta,function(x1,x2) bic.marg.post.phi(x1, x2) )
post.dens  <- exp(post.dens - max(post.dens))
post.dens  <- post.dens/sum(post.dens)

contours    <- seq(min(post.dens), max(post.dens) , length=10)
contour(alpha, beta, post.dens,levels=contours,
        xlab=expression(alpha),
        ylab=expression(beta),
        xlim=c(min(alpha),max(alpha)) ,
        ylim=c(min(beta),max(beta)),
        drawlabels=FALSE,
        main="Contour plot of joint posterior")

```

Contour plot of joint posterior



Draw samples (α^s, β^s) from $p(\alpha, \beta|y)$.

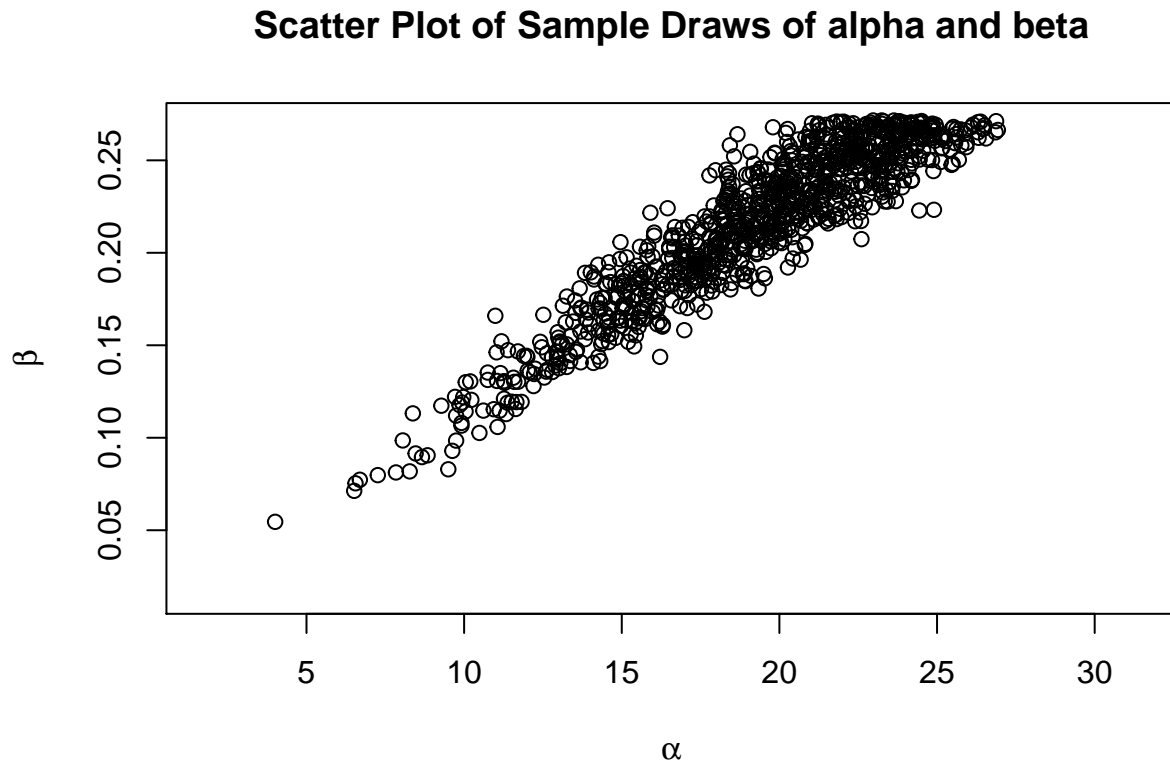
```
samps      <- 1000
alpha.dens  <- apply(post.dens ,1, sum)
s.alpha    <- sample(alpha,samps, replace=TRUE, prob = alpha.dens)

#Select the colum of the joint density corresponding to a specific
#value of v1 (p(beta/alpha))
cond.beta   <- function(x) {
  post.dens[which(alpha == s.alpha[x]),]
}

#Sample a value of v2 according the the conditional probatility above
s.beta      <- sapply(1:samps,function(x)
  sample(beta,1,replace=TRUE,prob=cond.beta(x)))

#Add a uniform random jitter centered at zero with with equal to the
#grid spacing. This will make the simulation draws more continuous.
#Plot the sampled values.
grid.alpha  <- alpha[2]-alpha[1]
grid.beta   <- beta[2]-beta[1]
s.beta      <- s.beta + runif(length(s.beta),-grid.beta/2,grid.beta/2)
s.alpha     <- s.alpha + runif(length(s.alpha),-grid.alpha/2,grid.alpha/2)
plot(s.alpha, s.beta,
     xlab=expression(alpha),
     ylab=expression(beta),
     xlim=c(min(alpha),max(alpha)) ,
```

```
ylim=c(min(beta),max(beta)),
main="Scatter Plot of Sample Draws of alpha and beta")
```



Note: regardless of how much I change the range of α and β I don't seem to cover the whole graph.

5.14c

Given the previous result we can say that the posterior is not integrable.

5.14d

I don't know how to alter it such that it becomes integrable.

5.15 [NEW] Meta-analysis

5.15a

Plot the posterior density of τ .

We use the functions written for Exercise 5.3

```
df <- read.csv("C:/Users/fhocesde/Documents/tutorials/Bayesian/BDA3/meta.csv")
y1 <- df$treated.deaths
```

```

y0 <- df$control.deaths
n1 <- df$treated.total
n0 <- df$control.total
effect <- log( y1/(n1- y1) ) - log( y0/(n0- y0) )
se.effect <- sqrt( 1/y1 + 1/(n1 - y1) + 1/y0 + 1/(n0 - y0) )

# Eqn 5.17 of BDA3
post.theta.j <- function(mu,tau,j) {
  ( effect[j] / ( se.effect[j]^2 ) + mu / ( tau^2 ) ) /
  ( 1 / ( se.effect[j]^2 ) + 1 / ( tau^2 ) )
}

post.v.theta.j <- function(tau,j) {
  1 / ( 1 / ( se.effect[j]^2 ) + 1 / ( tau^2 ) )
}

# Eqn 5.20 of BDA3
post.mu.hat <- function(tau) {
  sum( effect * 1 / ( se.effect^2 + tau^2 ) ) /
  sum( 1 / ( se.effect^2 + tau^2 ) )
}

post.v.mu <- function(tau) {
  ( sum( 1 / ( se.effect^2 + tau^2 ) ) )^-1
}

# Eqn 5.21 of BDA3
marginal.tau <- function(tau) {
  hyper.prior(tau) *
  ( post.v.mu(tau)^.5 ) *
  prod( ( ( se.effect^2 + tau^2 )^(-.5) ) *
    exp( - ( ( effect - post.mu.hat( tau ) )^2 )
      / ( 2 * ( se.effect^2 + tau^2 ) ) ) )
}

```

Define a hyper-prior and draw 5000 samples from each distribution (for all 22 studies).

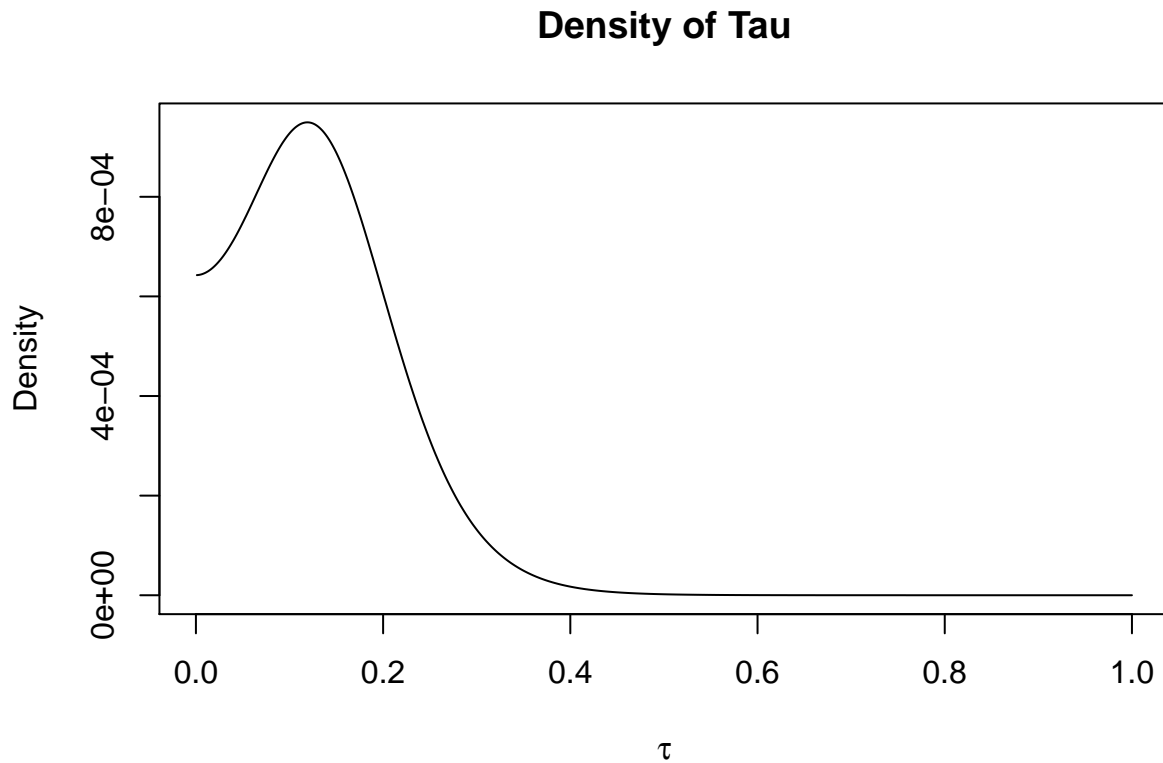
```

samps <- 5000

hyper.prior <- function(tau) 1
tau.grid <- seq(0.001,1, length=samps)
pdf.tau <- sapply(tau.grid,function(x) marginal.tau(x))
pdf.tau <- pdf.tau/sum(pdf.tau)

plot(tau.grid,pdf.tau, type="l",
     main="Density of Tau",
     xlab=expression(tau),
     ylab="Density")

```



5.15b

Produce figures analogous to 5.6 and 5.7 in BDA3 to show how the mean and variance of the posterior mean and standard deviations of θ depend on τ .

```
# Sampling
s.tau      <- sample(tau.grid,samps,prob=pdf.tau, replace=TRUE)
s.mu       <- sapply(s.tau,function(x) rnorm(1,post.mu.hat(x),(post.v.mu(x))^0.5))
s.theta    <- NULL
for (j in 1:length(df$study)) {
  s.theta[[j]] <- sapply(1:samps,
    function(x)
      rnorm(1,
        post.theta.j(s.mu[x],s.tau[x],j),
        (post.v.theta.j(s.tau[x],j))^0.5
      ) )
}
```

```
post.theta.j.no.mu <- function(tau,j) post.theta.j( post.mu.hat( tau ), tau, j )
post.se.theta.j.no.mu <- function(tau,j) sqrt(
  ( post.v.theta.j( tau, j ) ) * ( 1 + post.v.mu( tau ) * tau^(-2) ) )

plot( tau.grid, sapply(tau.grid, function(x) post.theta.j.no.mu(x,1) ),
  type="l", ylim=c(-.7,.3),
  xlab="", ylab="")
```

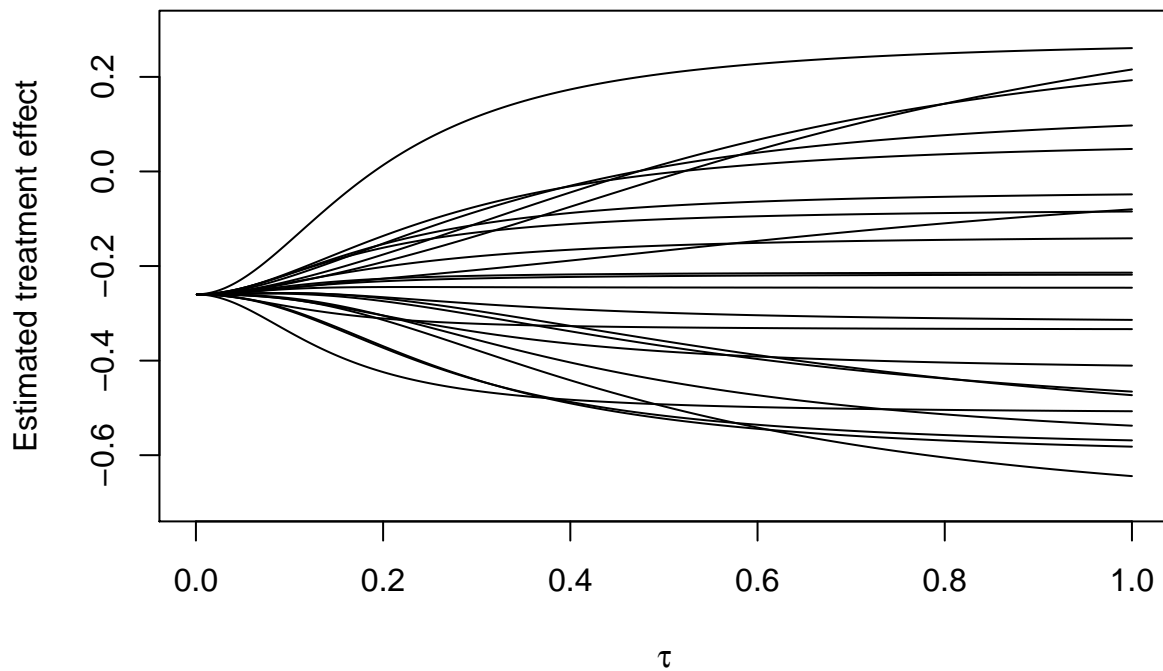
```

title(main="Analogous to Figure 5.6 from BDA3",
      xlab=expression(tau),
      ylab="Estimated treatment effect")

for (j in 2:22) {
  lines( tau.grid,
        sapply( tau.grid, function(x) post.theta.j.no.mu(x,j) ) )
}

```

Analogous to Figure 5.6 from BDA3



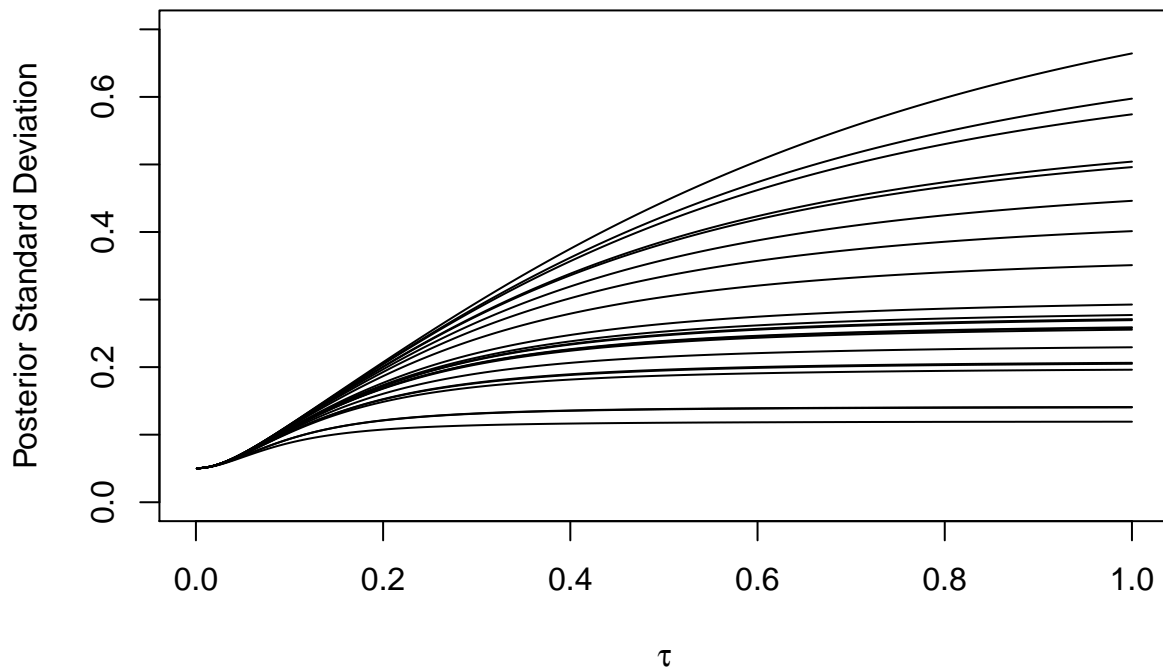
```

plot( tau.grid, sapply(tau.grid, function(x) post.se.theta.j.no.mu(x,1) ),
      type="l", ylim=c(0,.7),
      xlab="", ylab="")
title(main="Analogous to Figure 5.7 from BDA3",
      xlab=expression(tau),
      ylab="Posterior Standard Deviation")

for (j in 2:22) {
  lines( tau.grid,
        sapply( tau.grid, function(x) post.se.theta.j.no.mu(x,j) ) )
}

```

Analogous to Figure 5.7 from BDA3



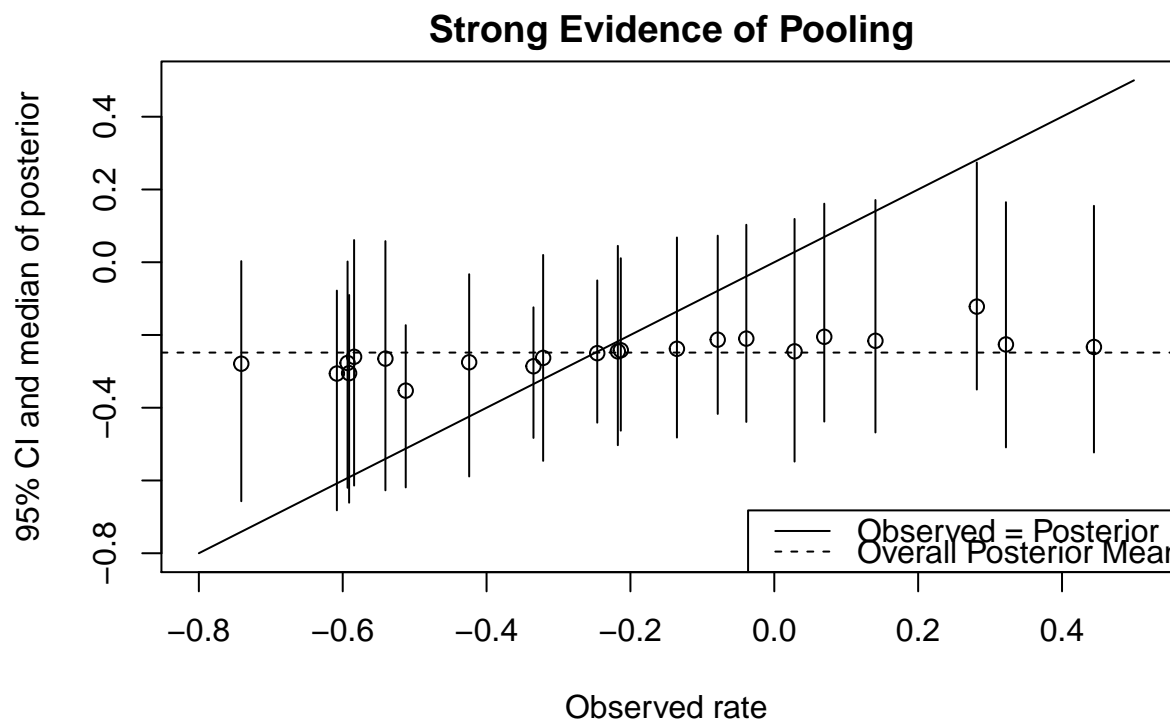
5.15c

Produce a scatterplot of the crude effects vs the posterior median. Show evidence of pooling

```
s.theta      <- matrix(unlist(s.theta), ncol = 22, byrow = FALSE)
s.theta.sort <- apply(s.theta, 2, sort)
theta.dist   <- t( apply( s.theta.sort, 2, function(x)
                        quantile( x, c( .025, .25, .5, .75, .975), type=1 ) ) )
theta.dist   <- round(theta.dist,3)
#Bug in line below
#knitr::kable(p, caption="Misclassification and Reliability for Different Thresholds")

x11()
plot(-800:500/1000,-800:500/1000, type = "l", lty=1,
     xlab="Observed rate",
     ylab="95% CI and median of posterior")
points(effect,theta.dist[,3])
segments(effect,theta.dist[,1], effect,theta.dist[,5] )
title(main="Posterior Distribution of Tumor Rates for all 22 Experiments
        \n Strong Evidence of Pooling")
abline(h = mean(s.theta), lty = 2)
legend("bottomright",legend=c("Observed = Posterior",
                              "Overall Posterior Mean"),lty = c(1,2))
```


Posterior Distribution of Tumor Rates for all 22 Experiments

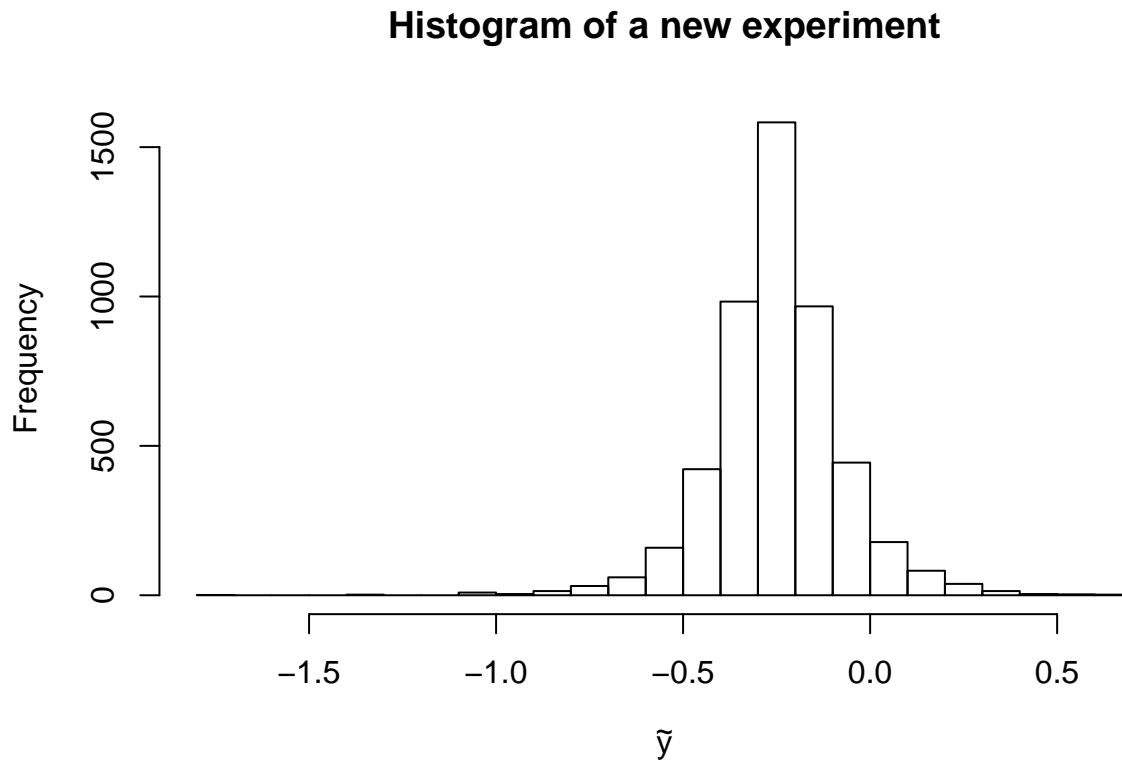


5.15d

Draw simulations from the posterior of a new treatment effect. Plot a histogram of the simulations.

Answer: to study a new treatment effect we need to draw samples from $N(\hat{\mu}, \hat{\tau})$, where :

```
new.exp <- rnorm(5000, mean = s.mu, sd = s.tau)
x11()
hist(new.exp, xlab = expression(tilde(y)) , breaks = 20,
     main="Histogram of a new experiment")
```



5.15d

Given the simulations just obtained, draw simulated outcomes from replications of a hypothetical new experiment with 100 persons in each of the treated and control groups. Plot a histogram of the crude estimated effect in the new experiment.

Not Answered: I don't know how to use the log odds to recover the treatment effect. If I knew then I would apply that function to each element just simulated, then I would compute the overall proportion of deaths under the control group and I would add the proportion under the treatment group. Finally I would use both proportions to get the total number of deaths in each group.

Chapter 10

10.1

10.1a

If $\theta \sim N(\mu, \sigma_\theta)$ then R draws $y^{(r)}$ will have a simulation standard error of $\hat{\sigma}_\theta / \sqrt{R}$. Hence in order to be within $0.1\sigma_\theta$ we need $R = 100$.

10.1b

For the simulation exercise we choose $\mu = 0, \sigma_\theta = 1$. We perform R draws and for each set of simulated numbers, we compute the 2.5% percentile $\{y^r\}_{p=0.025}$ and its difference with the theoretical percentile (-1.96),

we repeat this exercise 100 times and look at the average of the difference for different values of R .

```
y_reps      <- apply(
  sapply( 1:100,
    function(x) sapply(10^(1:5),
      function(x) abs(-1.96 -
        quantile(rnorm(x, mean=0, sd=1) , c(.025) ) ) )
    ), 1, mean
  )
```

Table 6: Standard Error of Simulations

R	$ \{y^r\}_{p=0.025} + 1.96 $
10	0.5857821
100	0.1771539
1000	0.0609399
10000	0.018992
100000	0.0069582

Note: both results don't match.

10.4a

I follow [this proof](#).

We want to prove that the conditional distribution of θ

We want to prove that $g(\theta|acceptance) = p(\theta)$. The pdf of a drawn from using the rejection sampling algorithm follows:

$$\begin{aligned}
 Pr(\theta \leq \theta^* | \theta \text{ is accepted}) &= Pr(\theta \leq \theta^* | U \leq \frac{p(\theta)}{Mg(\theta)}) \\
 &= \frac{Pr(\theta \leq \theta^* \text{ and } U \leq \frac{p(\theta)}{Mg(\theta)})}{Pr(U \leq \frac{p(\theta)}{Mg(\theta)})} \\
 &= \frac{\int_{-\infty}^{\theta^*} \int_0^{\frac{p(\theta)}{Mg(\theta)}} g(\theta) du d\theta}{\int_{-\infty}^{\infty} \int_0^{\frac{p(\theta)}{Mg(\theta)}} g(\theta) du d\theta} \\
 &= \frac{\int_{-\infty}^{\theta^*} \left[\frac{p(\theta)}{Mg(\theta)} - 0 \right] g(\theta) du d\theta}{\int_{-\infty}^{\infty} \left[\frac{p(\theta)}{Mg(\theta)} - 0 \right] g(\theta) du d\theta} \\
 &= \int_{-\infty}^{\theta^*} p(\theta) d\theta \quad \blacksquare
 \end{aligned}$$

Note: I was not able to map perfectly this proof with this [other one](#). that seems more complete. For the proof presented here, my main doubt regards the step between the second and third line.

11.2

[CHECK WITH SUSAN]

Replicate the computations for the bioassay example of section 3.7 using the Metropolis algorithm. Be sure to define your starting points and your jumping rule.

From 3.7 we have that the posterior of α and β follows:

$$p(\alpha, \beta | y, n, x) \propto \prod_{i=1}^k p(y_i | \alpha, \beta, n_i, x_i)$$

$$p(y_i | \alpha, \beta, n_i, x_i) \propto [\text{logit}^{-1}(\alpha + \beta x_i)]^{y_i} [1 - \text{logit}^{-1}(\alpha + \beta x_i)]^{n_i - y_i}$$

Our starting values will be the ML estimates.

```
# Data
x      <- c(-0.86, -0.30, -0.05,  0.73)
y      <- c(0, 1, 3, 5)
n      <- rep(5,4)

# ML estimation
mlfit <- glm( cbind(y, n-y) ~ x, family = binomial)
mlfit$coefficients = round(mlfit$coefficients*1000)/1000
res1 = paste(paste("(",mlfit$coefficients[1], sep=""),
             paste(mlfit$coefficients[2],")", sep=""), sep=",")
```

With the estimates $(\hat{\alpha}_0, \hat{\beta}_0) = (0.847, 7.749)$.

Now we will simulated values of α and β using the Metropolis algorithm. Our jumping function will be a bi-variate normal. We will start with a variance of $1 \times I$ and tweak in in order to get an acceptance ratio between 20% and 50%.

Note for Susan: So far I have not been able to get convergence bellow.

```
posterior <- function(arg1) {
  alpha <- arg1[1]
  beta  <- arg1[2]
  posterior <- 1
  for (i in 1:length(y)) {
    posterior <- posterior * (
      ( ( inv.logit( alpha + beta * x[i] ) ) ^ y[i] ) *
      ( ( 1 - inv.logit( alpha + beta * x[i] ) ) ^ ( n[i] - y[i] ) )
    )
  }
  posterior
}

#THERE HAS TO BE SOME ERROR IN THE CODE AS MY ALGORITHM KEEP DIVERGING
jumping <- function(theta, scale=.1) rmvnorm(1,
                                             mean = theta,
                                             sigma = scale * diag(2))

d.jumping <- function(theta, theta_1, scale=.1) dmnorm(theta,
                                                         theta_1,
```

```

scale * diag(2))

sims      <- 2000
theta     <- matrix(0,nrow = sims, ncol = 2)
accept    <- rep(0,sims)
r.all     <- rep(0,sims)
theta[1,] <- mlfit$coefficients

for (i in 2:sims)
{
  theta.star <- jumping(theta[i-1,], .1)
  #print(i)
  r <- min( exp(
    ( log( posterior(theta.star) ) ) -
    ( log( posterior(theta[i-1,]) ) )
  ), 1)

  r.all[i] <- r
  #print(r)
  if ( r < runif(1) )
  {
    theta[i,] <- theta.star
    accept[i] <- 1
  }
  else
  {
    theta[i,] <- theta[i-1,]
    accept[i] <- 0
  }
  sum(accept)
}

```

11.3

Given the following data:

```

df      <- data.frame(machine = rep(1:6, each=5),
                        measure = c(83, 92, 92, 46, 67,
                                   117, 109, 114, 104, 87,
                                   101, 93, 92, 86, 67,
                                   105, 119, 116, 102, 116,
                                   79, 97, 103, 79, 92,
                                   57, 92, 104, 77, 100))

J      <- length(unique(df$machine))
n      <- length(df$measure)

```

Implement a separate, a pooled and a hierarchical Gaussian model with common variance described in section 11.6. Run the simulations long enough for appropriate convergence. Using each of three models -separate, pooled, and hierarchical- report: (i) the posterior distribution of the mean of the quality measurements of the sixth machine, (ii) the predictive distribution for another quality measurement of the sixth machine, and (iii) the posterior distribution of the mean of the quality measurements of the seventh machine.

The Hierarchical Model

Data $y_{ij}, i = 1, \dots, 5, j = 1, \dots, 6$ are iid within each of J groups with $y_{ij} \sim N(\theta_j, \sigma^2)$. The prior distribution of θ_j is assumed to be normal with hyper-parameters μ, τ ($\theta_j \sim N(\mu, \tau^2)$), σ is assumed to be unknown and the hyper-prior is assumed to be uniform over $(\mu, \log(\sigma), \tau)$, which implies $p(\mu, \log(\sigma), \log(\tau)) \propto \tau$. The joint posterior density for all the parameters is:

$$p(\theta, \mu, \log(\sigma), \log(\tau) | y) \propto \tau \prod_{j=1}^J N(\theta_j | \mu, \tau^2) \prod_{j=1}^J \prod_{i=1}^{n_j} N(y_{ij} | \theta_j, \sigma^2)$$

Starting points

Select 10 θ_j randomly from the y_{ij} sample. And take μ to be the average starting values of θ_j .

```
set.seed(142857)
theta.0      <- sapply(1:6, function(x) sample(df$measure[df$machine==x],
                                              10, replace=TRUE))
mu.0         <- apply(theta.0, 1, mean)
```

1. Draw from conditional posterior of τ^2 using:

$$\hat{\tau}^2 = \frac{1}{J-1} \sum_{j=1}^J (\theta_j - \mu)^2$$

```
tau.hat.2    <- function(theta) {
  mu         <- mean(theta)
  tau.2      <- ( 1/(J-1) ) * sum((theta - mu)^2)
  return(tau.2)
}
```

And the fact that:

$$\tau^2 | \theta, \mu, \sigma, y \sim \text{Inv} - \chi^2(J-1, \hat{\tau}^2)$$

We can draw samples for τ^2

```
s.tau.post   <- function(theta) {
  tau.2       <- tau.hat.2(theta)
  tau.cond    <- (J - 1) * (tau.2)/rchisq(1, J-1)
  return(tau.cond)
}
```

2. Draw from conditional posterior of σ^2 Using:

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - \theta_j)^2$$

```
f.sigma.hat.2 <- function(theta) {
  sigma.hat.2 <- sapply(1:6, function(x) (df$measure[df$machine==x] - theta[x])^2)
  sigma.hat.2 <- (1/n) * sum(unlist(sigma.hat.2))
  return(sigma.hat.2)
}
```

And the fact that:

$$\sigma^2 | \theta, \mu, \tau, y \sim \text{Inv} - \chi^2(n, \hat{\sigma}^2)$$

We can draw samples for σ^2

```
s.sigma.post      <- function(theta) {
  sigma2.hat      <- f.sigma.hat.2(theta)
  sigma2.post     <- (n) * (sigma2.hat)/rchisq(1,n)
  return(sigma2.post)
}
```

3. Draw from conditional posterior of μ Using:

$$\hat{\mu} = \frac{1}{J} \sum_{j=1}^J \theta_j$$

```
mu.hat           <- function(theta) {
  mean(theta)
}
```

And the fact that:

$$\mu | \theta, \sigma, \tau, y \sim N(\hat{\mu}, \tau^2/J)$$

We can draw values for μ

```
s.mu             <- function(theta,tau2) {
  mu.hat         <- mu.hat(theta)
  rnorm(1,mu.hat,sqrt(tau2/J))
}
```

4. Finally, we can draw values for θ Using the fact that:

$$\theta_j | \mu, \sigma, \tau, y \sim N(\hat{\theta}_j, V_{\theta_j})$$

With:

$$\hat{\theta}_j = \frac{\frac{1}{\tau^2} \mu + \frac{n_j}{\sigma^2} \bar{y}_j}{\frac{1}{\tau^2} + \frac{n_j}{\sigma^2}}$$

$$V_{\theta_j} = \frac{1}{\frac{1}{\tau^2} + \frac{n_j}{\sigma^2}}$$

```
theta.hat.j      <- function(j,mu,sigma2,tau2) {
  y.bar.j        <- mean(df$measure[df$machine==j])
  n.j            <- length(df$measure[df$machine==j])
  ( (1/tau2) * mu + (n.j/sigma2) * y.bar.j ) / ( (1/tau2) + (n.j/sigma2) )
}

V.theta.hat.j    <- function(j,mu,sigma2,tau2) {
  n.j            <- length(df$measure[df$machine==j])
```

```

( 1 ) / ( (1/tau2) + (n.j/sigma2) )
}

s.theta      <- function(mu,sigma2,tau2) {
  theta      <- NULL
  for (j in 1:J) {
    t.hat     <- theta.hat.j(j,mu,sigma2,tau2)
    v.t.hat   <- V.theta.hat.j(j,mu,sigma2,tau2)
    theta[j]  <- rnorm(1,t.hat,sqrt(v.t.hat))
  }
  return(theta)
}

sims         <- 200
mcmc.gibbs   <- function(chain) {
  res1       <- as.list(NULL)
  param      <- 9
  s.param     <- matrix(NA, ncol = param, nrow = sims )
  colnames(s.param)<- c("theta1", "theta2", "theta3",
                       "theta4", "theta5", "theta6",
                       "mu", "sigma2", "tau2")
  s.param[1,1:6]<- theta.0[chain,]
  s.param[1,9]  <- s.tau.post(theta.0[chain,])
  s.param[1,8]  <- s.sigma.post(theta.0[chain,])
  s.param[1,7]  <- s.mu(theta.0[chain,],s.param[1,9])

  for (s in 2:sims) {
    s.param[s,1:6]<- s.theta(s.param[s-1,7],s.param[s-1,8],s.param[s-1,9])
    s.param[s,9]  <- s.tau.post(s.param[s,1:6])
    s.param[s,8]  <- s.sigma.post(s.param[s,1:6])
    s.param[s,7]  <- s.mu(s.param[s,1:6],s.param[s,9])
  }
  return(s.param)
}
#Warm-up
}
s.param      <- lapply(1:10, function(x) mcmc.gibbs(x))
s.param.1    <- s.param[[1]][101:200, ]

#Transform the variance in to sd.
s.param.1[,8:9] <- sqrt(s.param.1[,8:9] )

t(apply(s.param.1,2, function(x) quantile(x, c(.025,.25,.5,.75,.975))))

```

```

##          2.5%      25%      50%      75%      97.5%
## theta1 68.226089  75.421107  80.39244  85.48695  94.65629
## theta2 90.198936  96.618055  101.97768  107.76752  113.77738
## theta3 78.833357  85.921457  89.34768  93.42686  99.81877
## theta4 94.264990  102.451558  106.88234  111.32589  118.46376
## theta5 79.180597  87.179518  90.30892  94.51635  98.87628
## theta6 75.661817  84.683794  87.73719  91.37322  97.42413
## mu     83.438952  88.912486  92.94220  96.81082  105.40103
## sigma2 10.986146  12.872836  14.69001  16.11049  19.45944
## tau2   4.626044   8.358545  12.10177  16.47120  29.15127

```



```

r.hat      <- function(arg1) {
  n        <- dim(arg1)[1]
  m        <- dim(arg1)[2]
  B        <- (n/(m-1)) * sum( ( apply(arg1,2,mean) - mean(arg1) )^2 )
  W        <- (1/m) * (1/(n-1)) * sum( (arg1 - apply(arg1,2,mean))^2 )
  sqrt( (n-1)/(n) + B/(W*n) )
}

#r.hat(sapply(1:10,function(x) s.param[[x]][,4] ))

```

In addition, the separate and pooled models estimators of the mean of the sixth machine are both normal with means and variances defined by [Susan: I'm not so sure of the following definitions. If wrong, could you show me the correct answer?]:

$$\bar{y}_{..} = \frac{\sum_{i=1}^5 \sum_{j=1}^6 y_{ij}}{5 \times 6} \quad V_6^{pooled} = \frac{\sum_{i=1}^5 \sum_{j=1}^6 (y_{i6} - \bar{y}_{..})^2}{5 \times 6 - 1}$$

$$\bar{y}_{6.} = \frac{\sum_{i=1}^5 y_{6i}}{5} \quad V_6^{sep} = \frac{\sum_{i=1}^5 (y_{6i} - \bar{y}_{6.})^2}{5 - 1}$$

```

y.bar.dot.dot <- mean(df$measure)
var.pooled   <- sum( (df$measure[df$machine==6] -
                     mean(df$measure))^2 )/( length(df$measure) - 1)
s.theta.pooled.6<- rnorm(sims, y.bar.dot.dot, sqrt(var.pooled))
y.bar.6.dot   <- mean(df$measure[df$machine==6])
var.sep.6     <- sum( (df$measure[df$machine==6] -
                     mean(df$measure[df$machine==6]))^2 ) /
                 ( length(df$measure[df$machine==6]) - 1)
s.theta.sep.6 <- rnorm(sims, y.bar.6.dot, sqrt(var.sep.6))

```

Now we can answer the original questions:

- (i) The posterior distribution of the mean of the quality measurements of the sixth machine.

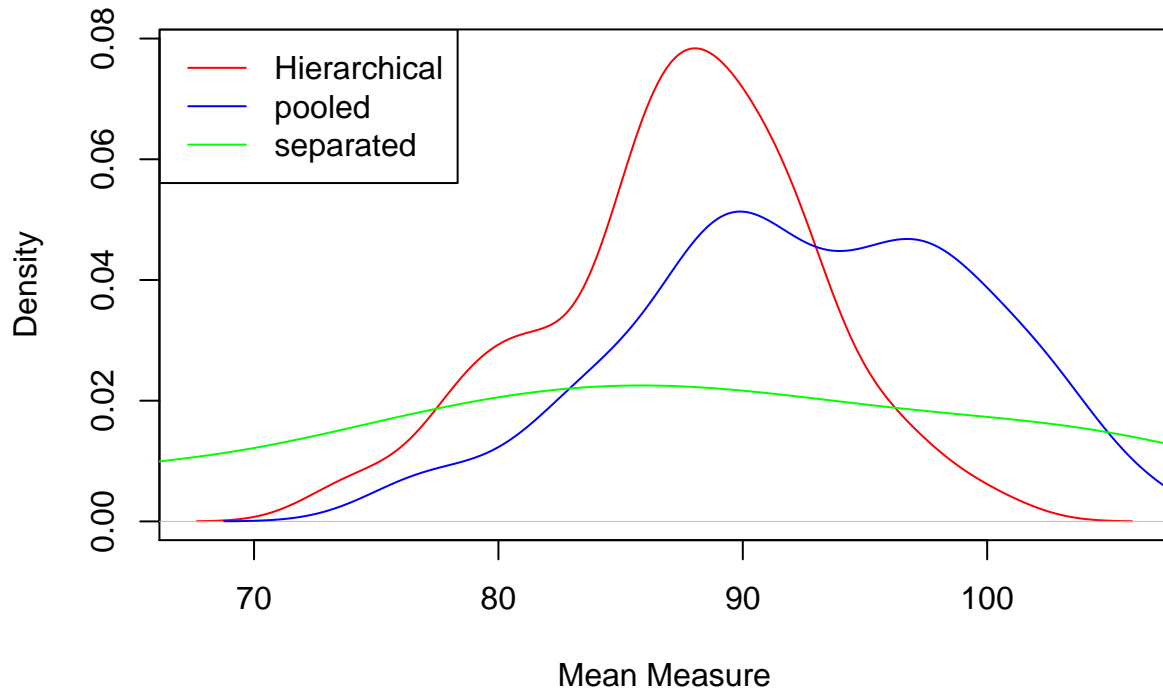
For the hierarchical, pooled and separate models: draw from $N(\theta_6, \sigma^2)$, $N(\bar{y}_{..}, V_6^{pooled})$, $N(\bar{y}_{6.}, V_6^{sep})$. The results are summarize in the following plot:

```

plot(density(s.param.1[, "theta6"]), col="red",
     xlab="Mean Measure",
     ylab="Density",
     main="Distribution of Mean 7th Measure under Three Models")
lines(density(s.theta.pooled.6), col="blue")
lines(density(s.theta.sep.6), col="green")
legend("topleft", col = c("red", "blue", "green"),
      legend=c("Hierarchical", "pooled", "separated"),
      lty = c(1,1,1))

```

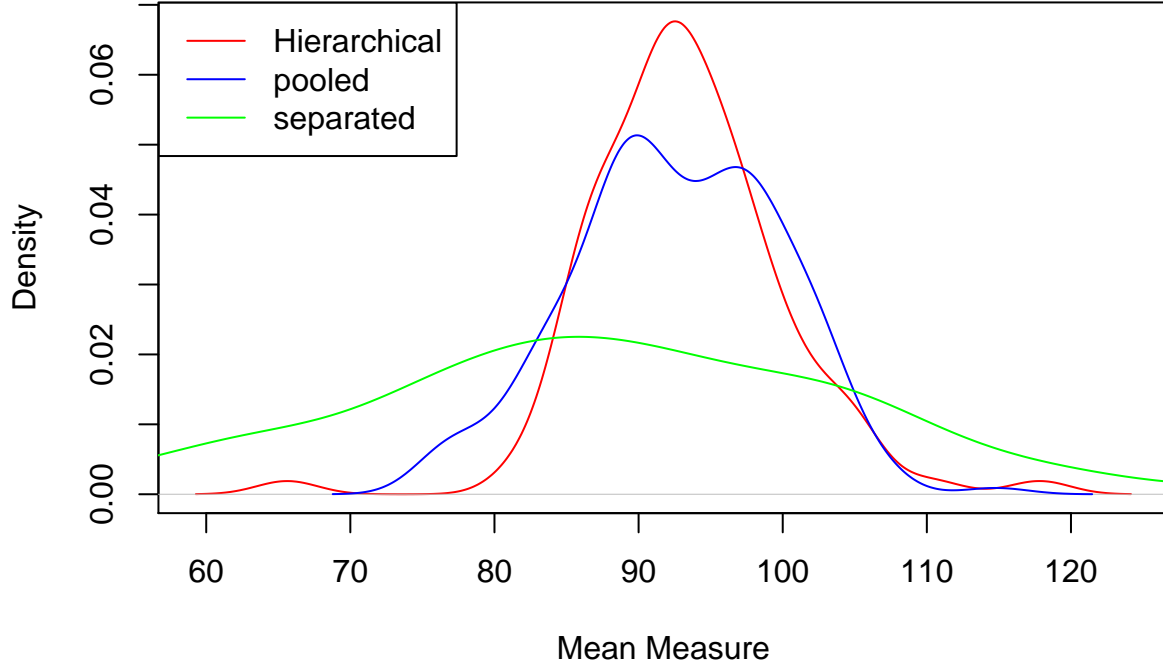
Distribution of Mean 7th Measure under Three Models



- (ii) The predictive distribution for another quality measurement of the sixth machine.
I was not able to think of a way to sample for this part, I would appreciate suggestions
- (iii) The posterior distribution of the mean of the quality measurements of the seventh machine.
For the hierarchical, pooled and separate models: draw from $N(\mu, \tau)$, $N(\bar{y}_{..}, V_6^{pooled})$, $N(\bar{y}_{6.}, V_6^{sep})$.
The results are summarize in the following plot:

```
plot(density(s.param.1[,"mu"]), col="red", xlab="Mean Measure",
     ylab="Density", main="Distribution of a 7th Measure under Three Models")
lines(density(s.theta.pooled.6), col="blue")
lines(density(s.theta.sep.6), col="green")
legend("topleft", col = c("red", "blue", "green"),
      legend=c("Hierarchical", "pooled", "separated"),
      lty = c(1,1,1))
```

Distribution of a 7th Measure under Three Models



11.6a

For any given scalar estimand ϕ , the simulation methods studied in chapter 11 generate *chains* of random numbers. It is common practice to delete (warm-up) the first half of each chain. The resulting half is usually split in to two new halves. Denote n the length of each resulting chain, and m the number of such chains. Show that:

$$\lim_{n \rightarrow \infty} mn \operatorname{var}(\bar{\psi}_{..}) = \left(1 + 2 \sum_{t=1}^{\infty} \rho_t\right) \operatorname{var}(\psi|y),$$

Where:

$$\bar{\psi}_{..} = \frac{1}{m} \sum_{j=1}^m \bar{\psi}_{.j}, \quad \text{and} \quad \bar{\psi}_{.j} = \frac{1}{n} \sum_{i=1}^n \bar{\psi}_{ij}$$

Then we have:

$$\begin{aligned} \operatorname{var}(\bar{\psi}_{..}) &= \frac{1}{(mn)^2} \left(\sum_i \sum_j \operatorname{var}(\psi_{ij}|y) + 2 \sum_i \sum_j \sum_{k \neq i} \sum_{l \neq j} \operatorname{cov}(\psi_{ij}, \psi_{kl}|y) \right) \\ &= \frac{1}{(mn)^2} mn \operatorname{var}(\psi|y) \left(2 \sum_i \sum_j \sum_{k \neq i} \sum_{l \neq j} \operatorname{corr}(\psi_{ij}, \psi_{kl}|y) \right) \end{aligned}$$

...

I was not able to show that the term inside the parenthesis was equal to $\sum_{t=1}^{\infty} \rho_t$.

Chapter 12

Install and run examples in STAN [DONE]

Chapter 14

14.1

```
# Analysis of Radon measures
y.1      <- c(5.0, 13.0, 7.2, 6.8, 12.8, 5.8, 9.5,
              6.0, 3.8, 14.3, 1.8, 6.9, 4.7, 9.5)
y.2      <- c(0.9, 12.9, 2.6, 3.5, 26.6, 1.5, 13.0,
              8.8, 19.5, 2.5, 9.0, 13.1, 3.6, 6.9)
y.3      <- c(14.3, 6.9, 7.6, 9.8, 2.6, 43.5, 4.9,
              3.5, 4.8, 5.6, 3.5, 3.9, 6.7)
basement.1 <- c(1,1,1,1,1,0,1,1,1,0,1,1,1)
basement.2 <- c(0,1,1,0,1,1,1,1,0,1,1,1,0)
basement.3 <- c(1,0,1,0,1,1,1,1,1,1,1,1)
counties  <- rep(1:3,c(length(y.1),length(y.2),length(y.3)))
y         <- c(y.1,y.2,y.3)
indicators <- as.matrix(table(1:length(counties), counties))
x         <- cbind(basement=c(basement.1,basement.2,basement.3),
                  indicators)

colnames(x) <- c("basement", "countie1", "countie2", "countie3")
lf.1       <- summary( lm(log(y)~x-1) )

nsim <- 10000
n <- nrow(x)
k <- ncol(x)

s.sigma    <- lf.1$sigma*sqrt((n-k)/rchisq(nsim,n-k))
s.beta     <- matrix(rep(lf.1$coefficients[,1], 2),
                    ncol=4, nrow=nsim, byrow = TRUE) +
  matrix(rnorm( nsim * k ),
        nrow = nsim, ncol = k ) %*%
  chol( vcov(lf.1) ) * s.sigma/lf.1$sigma
```

14a

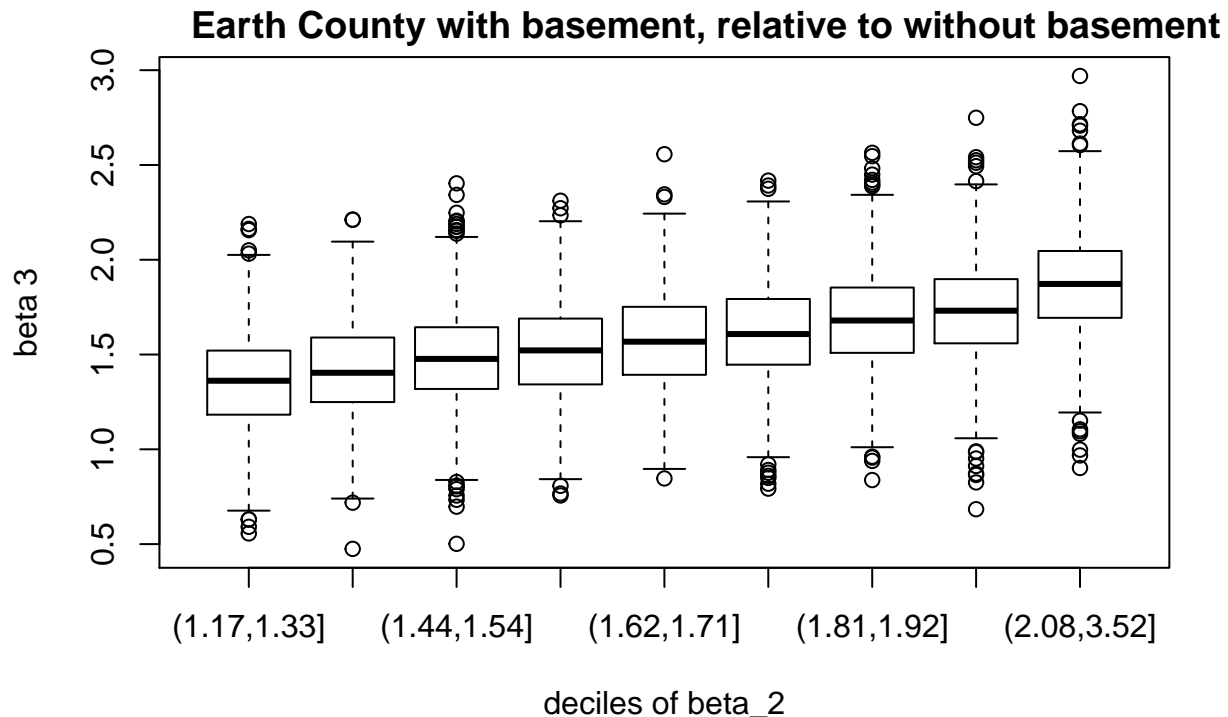
Fit a linear regression to the logarithms of the radon measurements (y), with indicator variables for the three counties and for whether a measurement was recorded on the first floor. Summarize your posterior inference in nontechnical terms.

Answer: after estimating the posterior of the parameters we can explore the potential of the bayesian approach. Given that we have an estimation of the full posterior of the parameters we could analyse the data in all sorts of ways. In the classical approach on the other hand, any time that we would like to explore the model beyond linear hypothesis, we would need to use the delta method and some derivations to obtain the distribution of each new hypothesis.

For example in the bayesian setup we could ask: does the mean for Blue Earth County with basement (β_3) changes in some systematic way with the variation of the mean without basement for the same county (β_2)? The answer is presented in the following figure:

```
s.beta.2.bins <- cut(s.beta[,2], quantile(s.beta[,2], probs=1:10/10))
boxplot(s.beta[,3]~s.beta.2.bins, xlab="deciles of beta_2", ylab="beta 3",
        main="Distribution of the mean for B.\n
        Earth County with basement, relative to without basement")
```

Distribution of the mean for B.



```
#output <- exp (cbind (beta[,2], beta[,1]+beta[,2], beta[,3],
#beta[,1] + beta[,3], beta[,4], beta[,1] + beta[,4], beta[,1], sigma))
#for (i in 1:ncol(output)) print (round(quantile(output[,i],c(.25,.5,.75)),1))
```

As we can see, this two effects co-variate positively.

14b

Suppose another house is sampled at random from Blue Earth County (`counties=1`). Sketch the posterior predictive distribution for its radon measurement and give a 95% predictive interval. Express the interval on the original (unlogged) scale.

A proposed answer is [here](#) but I did not write it down as I don't fully understand it yet (and run out of time).

15

Run a full HLM example and compare with a fixed effect regression.

A proposed answer is [here](#) but I did not write it down as I don't fully understand it yet (and run out of time).

The following code comes from this website and make reference to the original author as Dr. Surya Tokdar.

```

setwd("C:/Users/fhocesde/Documents/tutorials/Bayesian/BDA3")
## Get Data

prez88 <- read.table("prez48to88.txt", header = T)
prez92 <- read.table("prez92.txt", header = T)

## Get y, X

y <- prez88[,1]
X <- as.matrix(prez88[, -(1:4)])
n <- nrow(X)
p <- ncol(X)

## Least square calculations

beta.ls <- lm(y ~ 0 + X)$coeff
V <- crossprod(X)
resid.ls <- y - c(X %*% beta.ls)
s2 <- sum(resid.ls^2) / (n - p)

## Xnew from 1992 data

Xnew <- as.matrix(prez92[, -(1:4)])
nnew <- nrow(Xnew)

```