

STA 602 HW5

William Tirone

HW 4 part 3 d)

The MAEs actually look fairly similar in shape to the MSEs calculated and plotted in the previous homework. Very roughly, that is - in the sense that there is no single best estimator across ALL thetas, so we may have to compromise in certain ranges of theta. There's a lot more "bumpiness" here though than the MSEs.

I think δ_5 is still minimax just based on observing what the plots look like, but it is difficult to say since we haven't developed a formal definition of what the minimax estimator is yet.

```
thetas = seq(0,1,0.01)

S = 100000
n=5

d1.result = c()
d2.result = c()
d3.result = c()
d4.result = c()
d5.result = c()

for (i in thetas) {
  data = rbinom(S,n,i)

  d1.hat = (data/n)
  d1.MAE = mean(abs(d1.hat - i))
  d1.result = c(d1.MAE,d1.result)

  d2.hat = .5
  d2.MAE = mean(abs(.5 - i))
```

```

d2.result = c(d2.MAE,d2.result)

d3.hat = (data + 12) / (n + 24)
d3.MAE = mean(abs(d3.hat - i))
d3.result = c(d3.MAE,d3.result)

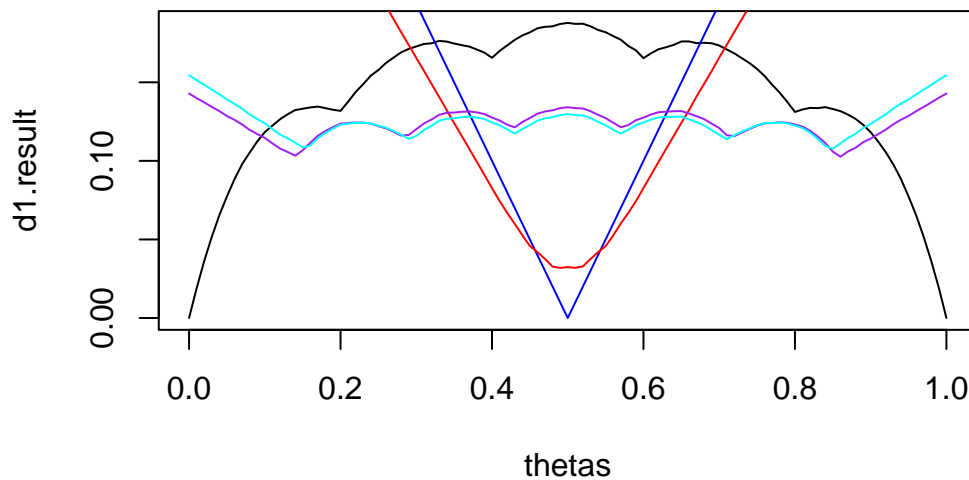
d4.hat = (data + 1) / (n + 2)
d4.MAE = mean(abs(d4.hat - i))
d4.result = c(d4.MAE,d4.result)

d5.hat = (data + sqrt(n)/2) / (n + sqrt(n))
d5.MAE = mean(abs(d5.hat - i))
d5.result = c(d5.MAE,d5.result)

}

plot(thetas,d1.result,type='l')
lines(thetas,d2.result,type='l',col='blue')
lines(thetas,d3.result,type='l',col='red')
lines(thetas,d4.result,type='l',col='purple')
lines(thetas,d5.result,type='l',col='cyan')

```



n=100

```

d1.result = c()
d2.result = c()

```

```

d3.result = c()
d4.result = c()
d5.result = c()

for (i in thetas) {
  data = rbinom(S,n,i)

  d1.hat = (data/n)
  d1.MAE = mean(abs(d1.hat - i))
  d1.result = c(d1.MAE,d1.result)

  d2.hat = .5
  d2.MAE = mean(abs(.5 - i))
  d2.result = c(d2.MAE,d2.result)

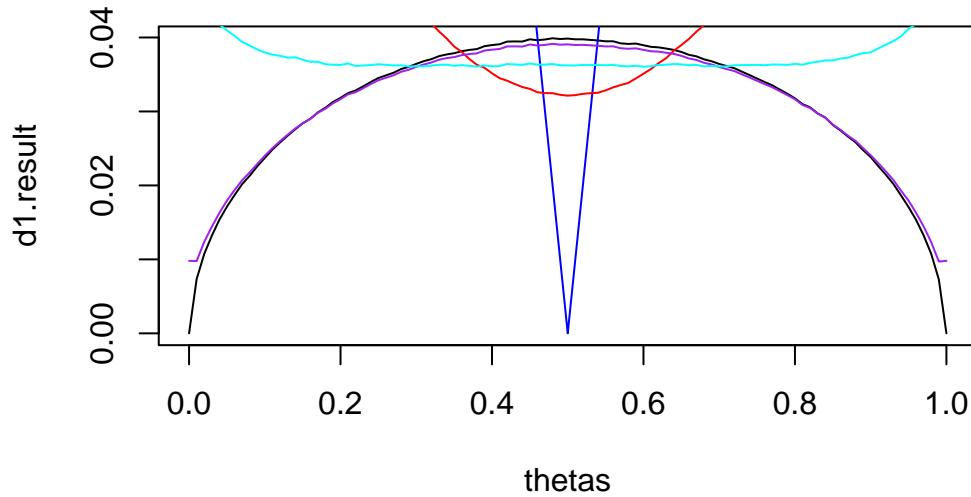
  d3.hat = (data + 12) / (n + 24)
  d3.MAE = mean(abs(d3.hat - i))
  d3.result = c(d3.MAE,d3.result)

  d4.hat = (data + 1) / (n + 2)
  d4.MAE = mean(abs(d4.hat - i))
  d4.result = c(d4.MAE,d4.result)

  d5.hat = (data + sqrt(n)/2) / (n + sqrt(n))
  d5.MAE = mean(abs(d5.hat - i))
  d5.result = c(d5.MAE,d5.result)
}

plot(thetas,d1.result,type='l')
lines(thetas,d2.result,type='l',col='blue')
lines(thetas,d3.result,type='l',col='red')
lines(thetas,d4.result,type='l',col='purple')
lines(thetas,d5.result,type='l',col='cyan')

```



4.1

Using beta - uniform conjugacy, we know the posterior will also be a beta distribution. So we have:

$$p(\theta_2|Y = 30) \sim \text{beta}(30 + 1, 20 + 1)$$

$$p(\theta_1|Y = 57) \sim \text{beta}(57 + 1, 43 + 1)$$

$$P(\theta_1 < \theta_2|data) = \frac{1}{S} \sum_{i=1}^S 1(\theta_1^{(S)} < \theta_2^{(S)})$$

```
theta1.sample = rbeta(5000,58,44)
theta2.sample = rbeta(5000,31,21)

mean(theta1.sample < theta2.sample)
```

```
[1] 0.6338
```

4.2

a)

The posteriors in 3.3 part a were given by:

$$\theta_A|y_A \sim \text{Gamma}(237, 20)$$

$$\theta_B|y_B \sim \text{Gamma}(125, 14)$$

Using the same methodology above with the indicator function:

```
thetaA.sample = rgamma(5000,237,20)
thetaB.sample = rgamma(5000,125,14)

mean(thetaB.sample < thetaA.sample)
```

```
[1] 0.9954
```

b)

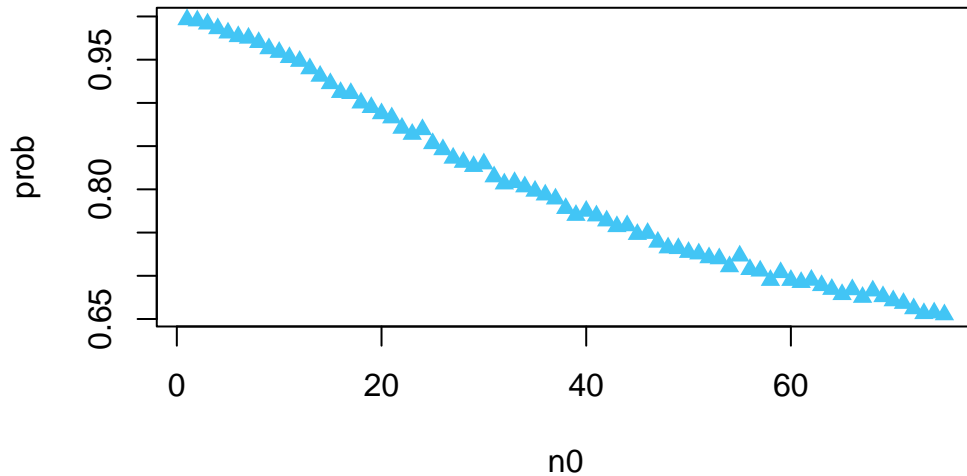
Our posterior for $\theta_B|y_B$ is $\text{Gamma}(\sum y + \alpha, \beta + n)$ so iterating through we will have $\text{Gamma}(117 + \alpha, \beta + 10)$. Observing the graph, it doesn't look like the event $\{\theta_A < \theta_B\}$ is incredibly sensitive, since it drops off fairly linearly over the iterations of n_0 . I would consider it sensitive if the probability dropped sharply at any n_0 , but it doesn't.

```
# original data from 3.3
yB = c(11, 11, 10, 9, 9, 8, 7, 10, 6, 8, 8, 9, 7)
s = sum(yB)
n = length(yB)

# calculating the updates
n0 = seq(1,75,1)
prob = c()

for (i in n0){
  thetaB.sample = rgamma(5000, s+(12*i),i+n)
  prob = c(prob,mean(thetaB.sample < thetaA.sample))
}

plot(n0,prob,col='#42c5f5',pch=17)
```



c)

From p. 47/48 of Hoff we know that a poisson sampling model and gamma posterior will give a negative binomial predictive distribution with parameters $(\alpha + \sum y_i, \beta + n)$. So we have:

$$\tilde{Y}_A \sim NBIN(237, 20)$$

$$\tilde{Y}_B \sim NBIN(125, 14)$$

repeating part a) and using the parameterization from p. 49 (using mu, the posterior mean):

```
predictA.sample = rnbinom(5000,size=237,mu=237/20)
predictB.sample = rnbinom(5000,size=125,mu=125/14)

mean(predictB.sample < predictA.sample)
```

```
[1] 0.6968
```

repeating part b), since the predictive distribution has the same parameters as the posterior we found above, I'm just updating mu, the posterior mean, with each loop. This is more sensitive to changes in the prior since it looks like the probability drops from 0.65 to about 0.5 fairly quickly based on the prior.

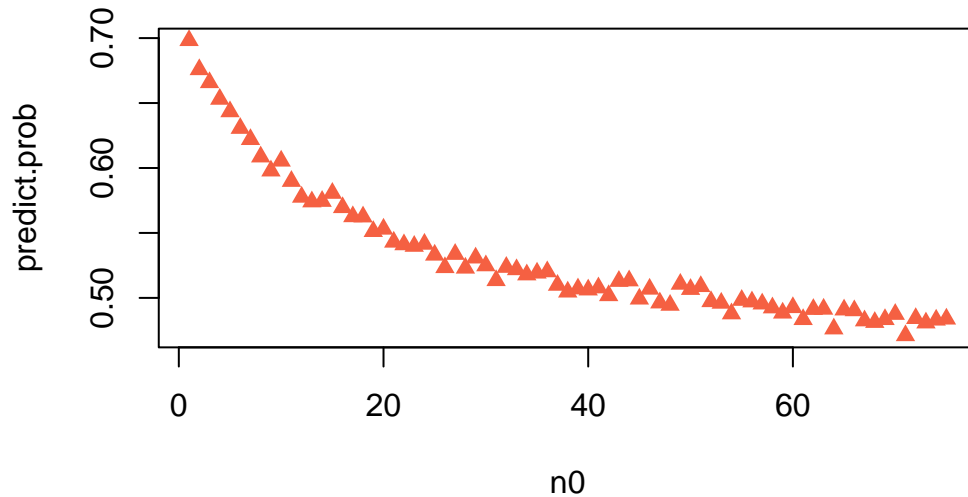
```
# calculating the updates
n0 = seq(1,75,1)
predict.prob = c()
```

```

for (i in n0){
  predictB.sample = rlnbinom(5000, s+(12*i),mu = (113 + (12*i)) / (i+n))
  predict.prob = c(predict.prob,mean(predictB.sample < predictA.sample))
}

plot(n0,predict.prob,col='#f56042',pch=17)

```



4.4

a)

Our posterior from 3.4 is plotted below with the 95% CI.

Here, I'm trying to calculate the updated weights based on what we did in question 3.4 and from referring to the homework solution here.

Honestly the 95% CI worked better than I would have thought, I just used a mixture of the quantiles. I'm guessing this is just an approximation but not positive. I tried manually integrating this without updating the weights (just using proportionality) and that wasn't working so this was my solution!

```

a1 = 2
B1 = 8
a2 = 8
B2 = 2
y = 15

```

```

n = 43

w0 = .75
w1 = .25

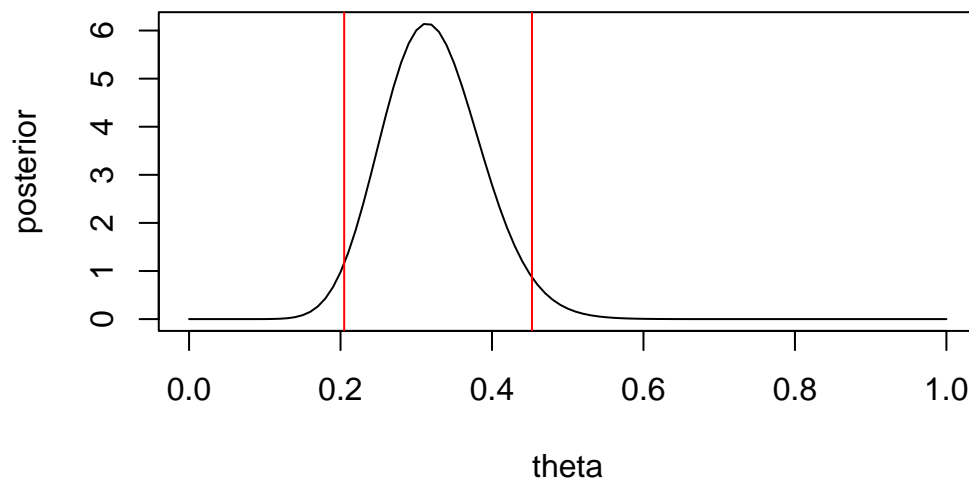
#updated weight
w2 = (w0 * beta(a1 + y, B1 + n - y)) / ((w0 * beta(a1 + y, B1 + n - y)) + (1-w0)*(beta(a2 +
w3 = 1-w2

theta = seq(0,1,0.01)
posterior = w2 * dbeta(theta, 17,36) + w3 * dbeta(theta, 23,30)

l = w2 * qbeta(0.025, 17,36) + w3 * qbeta(0.025, 23,30)
u = w2 * qbeta(0.975, 17,36) + w3 * qbeta(0.975, 23,30)

plot(theta,posterior,type='l')
abline(v=l,col='red')
abline(v=u,col='red')

```



```

cat("95% CI based on our mixture posterior : ",l,u)

```

95% CI based on our mixture posterior : 0.204828 0.452786

b)

Based on the question, I'm not sure if I need to state the distribution I've approximated, but I've simply plotted the data from the MC process below and hope that counts for what the question is asking.

For the 95% CI, I just found the quantiles of the generated data. I'm not positive this is the correct approach, but it looks very similar to CI calculated above in part a).

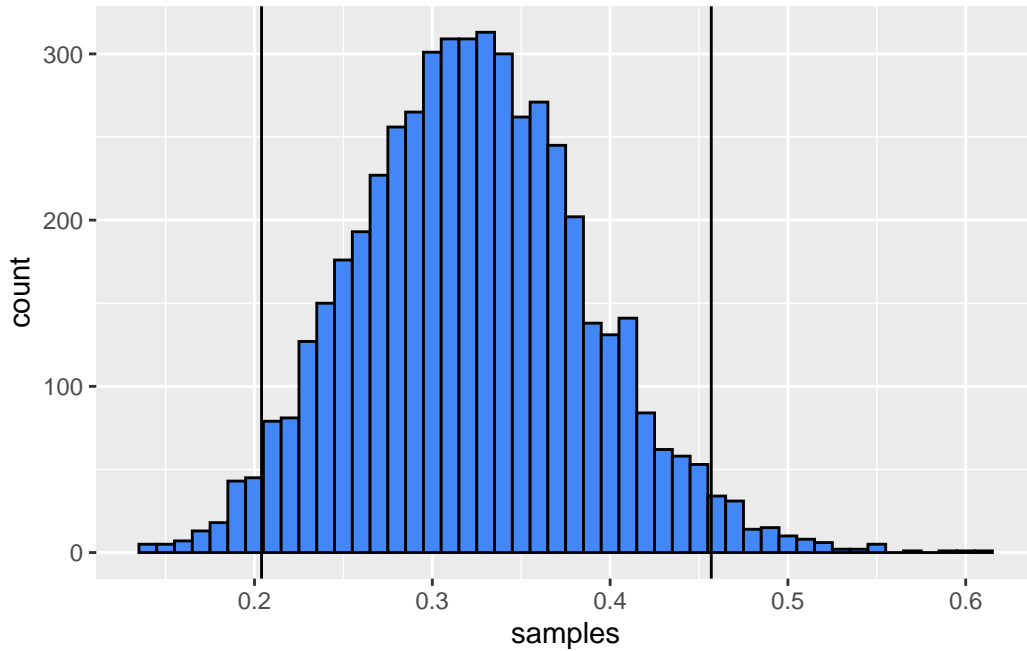
```
w=w2
x = rbinom(5000,1,w)
samples = c()

for (i in x){
  if (i==1){
    samples = c(rbeta(1,17,36),samples)
  } else if (i==0) {
    samples = c(rbeta(1,23,30),samples)
  }
}

samples.df = as.data.frame(samples)

q = quantile(samples.df$samples, probs = c(0.025,0.975))
l_mc = q[1]
u_mc = q[2]

ggplot(samples.df,aes(x=samples)) + geom_histogram(color='black',fill='#4287f5',binwidth =
```



```
cat('95% CI Using MC Method',q)
```

```
95% CI Using MC Method 0.2039566 0.4569012
```

4.5

a)

Let $Y = iid = Y_1, \dots, Y_n$ and $X = iid = X_1, \dots, X_n$

$$\begin{aligned}
 p(\theta|(Y, X)) &\propto P(Y, X|\theta)p(\theta) \\
 &\propto \prod_{i=1}^n \frac{e^{-\theta X_i} (\theta X_i)^{Y_i}}{Y_i!} \cdot d\text{gamma}(a, b) \\
 &\propto \frac{e^{-\theta \sum X_i} (\theta)^{\sum Y_i}}{\prod Y_i!} \cdot \frac{b^a}{\Gamma(a)} \theta^{a-1} e^{-b\theta} \\
 &\propto c(X, Y, a, b) \theta^{\sum(Y_i+a)-1} e^{-\theta(\sum X_i+b)}
 \end{aligned}$$

which is the kernel of a gamma, so our posterior is $\text{gamma}(\sum Y_i + a, \sum X_i + b)$ and we know there is poisson-gamma conjugacy as well.

b)

Per question, we will use the following priors:

$$\begin{aligned}\theta_1(\text{no reactor}) &\sim \text{gamma}(a_1, b_1) \\ \theta_2(\text{nearby reactors}) &\sim \text{gamma}(a_2, b_2)\end{aligned}$$

Based on the sums of the data calculated below, we will have the following posteriors for any a & b values on the prior, where the parameters are both in units of 10,000s:

$$\begin{aligned}p(\theta_1|X, Y) &= \text{gamma}(a_1 + 2285, b_1 + 1037) \\ p(\theta_2|X, Y) &= \text{gamma}(a_2 + 256, b_2 + 95)\end{aligned}$$

```
# y = pop in thousands, x = deaths per thousands
cancer_noreact = read.table('http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/cancer_noreact.data')
cancer_react = read.table('http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/cancer_react.data')

apply(cancer_noreact, MARGIN=2, FUN=sum)
```

```
      x      y
1037 2285
```

```
apply(cancer_react, MARGIN=2, FUN=sum)
```

```
      x      y
95 256
```

c)

For each of the below we need:

$E(\theta_1|data)$
 $E(\theta_2|data)$
95%CI for θ_1 & θ_2
 $p(\theta_2 > \theta_1|data)$
plot posterior densities

```

# writing these out to compute a variety of posteriors later
t1.sumY = 2285
t1.sumX = 1037
t2.sumY = 256
t2.sumX = 95

post_CI = function(a1,b1,a2,b2,sumY1,sumX1,sumY2,sumX2){

  alpha1 = a1 + sumY1
  beta1 = b1 + sumX1

  alpha2 = a2 + sumY2
  beta2 = b2 + sumX2

  CI_1 = qgamma(c(0.025,0.975),alpha1,beta1)
  CI_2 = qgamma(c(0.025,0.975),alpha2,beta2)

  predict.t1.sample = rgamma(5000,alpha1,beta1)
  predict.t2.sample = rgamma(5000,alpha2,beta2)
  prob = mean(predict.t1.sample < predict.t2.sample)

  cat("95% CI for a = ",a1," and b = ",b1," : ",CI_1)
  cat('\n')
  cat("95% CI for a = ",a2," and b = ",b2," : ",CI_2)
  cat('\n')
  cat("p(theta 2 > theta 1 | data) = ", prob)

  thetas = seq(0,3,0.001)
  values1 = dgamma(thetas,alpha1,beta1)
  values2 = dgamma(thetas,alpha2,beta2)
  plot(thetas,values1,type='l',col='purple')
  lines(thetas,values2,type='l',col='blue')
}

```

i)

$$p(\theta_1|data) = \text{gamma}(220 + 2285, 100 + 1037)$$

$$p(\theta_2|data) = \text{gamma}(220 + 256, 100 + 95)$$

$$E(\theta_1|data) = \frac{\alpha}{\beta} = 2.2032$$

$$E(\theta_2|data) = 2.4412$$

Below, blue plot is posterior for θ_1 , purple is θ_2

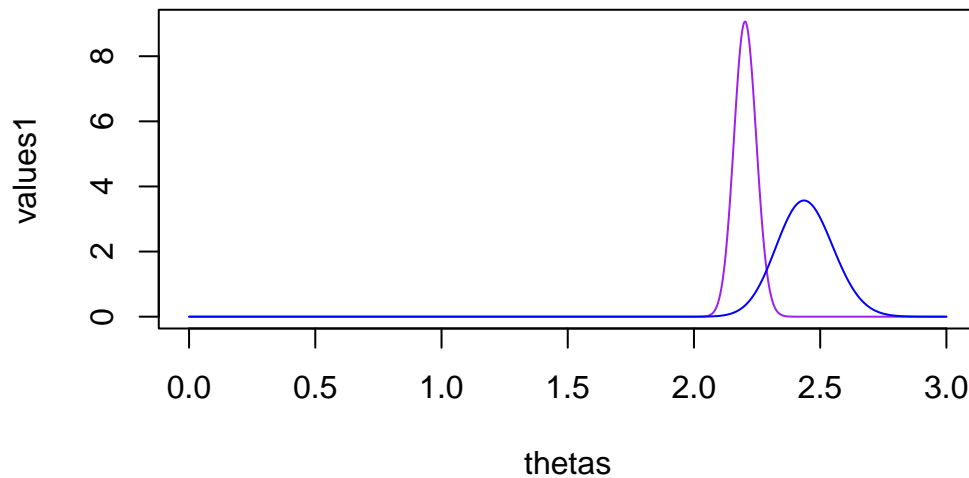
The opinion that “cancer rates are similar in both types of counties” will sort of serve as our null hypothesis. The rates of cancer are about 0.3 cancer deaths per 10,000 in the counties near nuclear reactors. The posterior is also “flatter” for theta2 than it is for theta1 (not near reactors) which indicates more uncertainty about the posterior. This is also just because we have far more measurements for the counties not near reactors.

```
# the best functions have 8 arguments
post_CI(a1=220,
        b1=100,
        a2=220,
        b2=100,
        sumY1=t1.sumY,
        sumX1=t1.sumX,
        sumY2=t2.sumY,
        sumX2=t2.sumX)
```

```
95% CI for a = 220 and b = 100 : 2.117726 2.290273
```

```
95% CI for a = 220 and b = 100 : 2.226633 2.665131
```

```
p(theta 2 > theta 1 | data) = 0.9772
```



ii)

$$p(\theta_1|data) = \text{gamma}(220 + 2285, 100 + 1037)$$

$$p(\theta_2|data) = \text{gamma}(2.2 + 256, 1 + 95)$$

$$E(\theta_1|data) = \frac{\alpha}{\beta} = 2.2032$$

$$E(\theta_2|data) = 2.6896$$

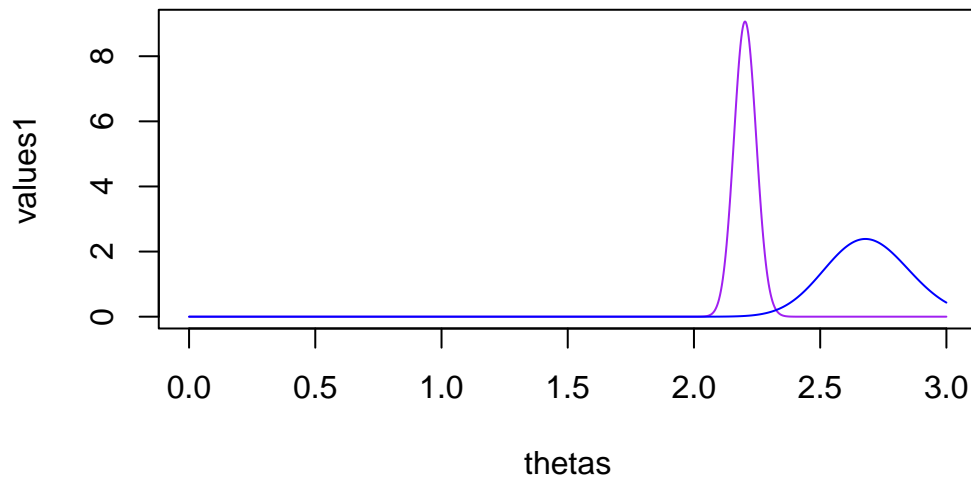
I'm not sure if this counts as a non-informative prior, but just saying “both counties see similar measurements as previous years” does not affect the posterior much. However, this posterior for theta2 is even flatter than the opinion from (i) which indicates even more uncertainty. Again the sampled data dominates the posterior.

```
post_CI(a1=220,
        b1=100,
        a2=2.2,
        b2=1,
        sumY1=t1.sumY,
        sumX1=t1.sumX,
        sumY2=t2.sumY,
        sumX2=t2.sumX)
```

95% CI for a = 220 and b = 100 : 2.117726 2.290273

95% CI for a = 2.2 and b = 1 : 2.371497 3.027397

p(theta 2 > theta 1 | data) = 0.9984



```
(2.2 + 256) / (1+95)
```

```
[1] 2.689583
```

iii)

$$p(\theta_1|data) = \text{gamma}(2.2 + 2285, 1 + 1037)$$

$$p(\theta_2|data) = \text{gamma}(2.2 + 256, 1 + 95)$$

$$E(\theta_1|data) = 2.2035$$

$$E(\theta_2|data) = 2.6896$$

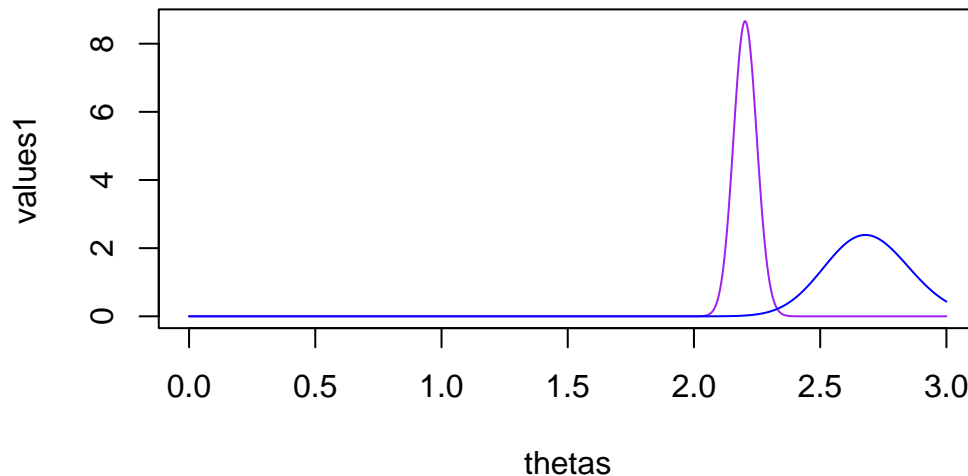
The plots here (and confidence intervals, etc) are virtually the same because the sample dominates the posteriors. Our priors are adding almost nothing to the data.

```
post_CI(a1=2.2,
        b1=1,
        a2=2.2,
        b2=1,
        sumY1=t1.sumY,
        sumX1=t1.sumX,
        sumY2=t2.sumY,
        sumX2=t2.sumX)
```

```

95% CI for a = 2.2 and b = 1 : 2.114081 2.29468
95% CI for a = 2.2 and b = 1 : 2.371497 3.027397
p(theta 2 > theta 1 | data) = 0.9988

```



d)

Since we have the data in 10,000s, we're ignoring population size and just considering everything as a proportion. There's probably something interesting here in the sense that nuclear reactors are probably not close to major population centers for obvious reasons! Because of that, we should probably be looking at this in raw population numbers. This would change the scale of the plots clearly, so the comparison might be a little more difficult.

e)

How: If we're relying on expert opinion, I think it would be difficult to prove you truly had independent priors if they're coming from the same expert. Mathematically, of course, we could just verify that $p(\theta_1, \theta_2) = p(\theta_1)p(\theta_2)$ based on our source of prior.

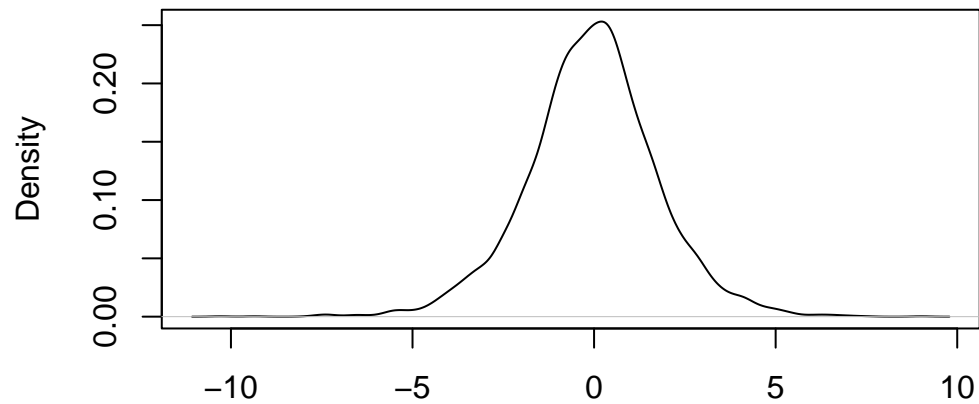
Why: if we want our posteriors to be independent, we want the priors to be independent. That way, we can try to make inference by comparing the two posteriors.

4.6

Following with the example from PH p. 58, the density of γ below looks more informative than a uniform prior. This is peaked over 0 which indicates that we are more confident about $\gamma = 0$ than we are about other values, which is in contrast to a uniform belief.


```
a=1  
b=1  
theta.prior.mc = rbeta(5000,1,1)  
gamma.prior.mc = log(theta.prior.mc / (1-theta.prior.mc))  
  
plot(density(gamma.prior.mc))
```

density.default(x = gamma.prior.mc)



N = 5000 Bandwidth = 0.2584