

Assignment 7.b for **STATGR6103**

submitted to Professor Andrew Gelman

Advait Rajagopal

24 October 2016

1 Question 1

Write a few sentences on ideas for your final project. This will be the basis of our discussion in class where we form pairs and make plans. Download some data and make some plots in preparation for this in-class discussion. Or, if your plan is more theoretical or computational, simulate some data and make some plots.

As discussed earlier, our idea is to identify the nature of the relationship between infant mortality rate (IMR) and state health expenditure in India. We suspect and moreover have a strong prior knowledge that there will be variation across states in India owing to deep seeded development issues and rampant inequality across states. So we built a dataset that has the following variables, for 30 states¹ in India, from 2006 - 2014;

1. IMR
2. State health expenditure
3. State GDP
4. Female to male ratio
5. Literacy rates

We create a state “id” variable for each of the 30 states and a time “id” variable for variation across time. While IMR and state health expenditure are the variables of interest there are several other factors that impact infant health and death and these include literacy rate and female to male ratios because these are typically low in the economically and socially worst off states.

For instance you would expect to see a higher IMR in a state with low literacy rate as female infanticide is very common in underdeveloped states. I make some plots summarizing these relationships. Figure 1 and 2 show the variation in health expenditure and IMR per state across the stipulated time period.

¹We call them states but most of them are states with state governments and a few of them are Union territories administered by the Federal government

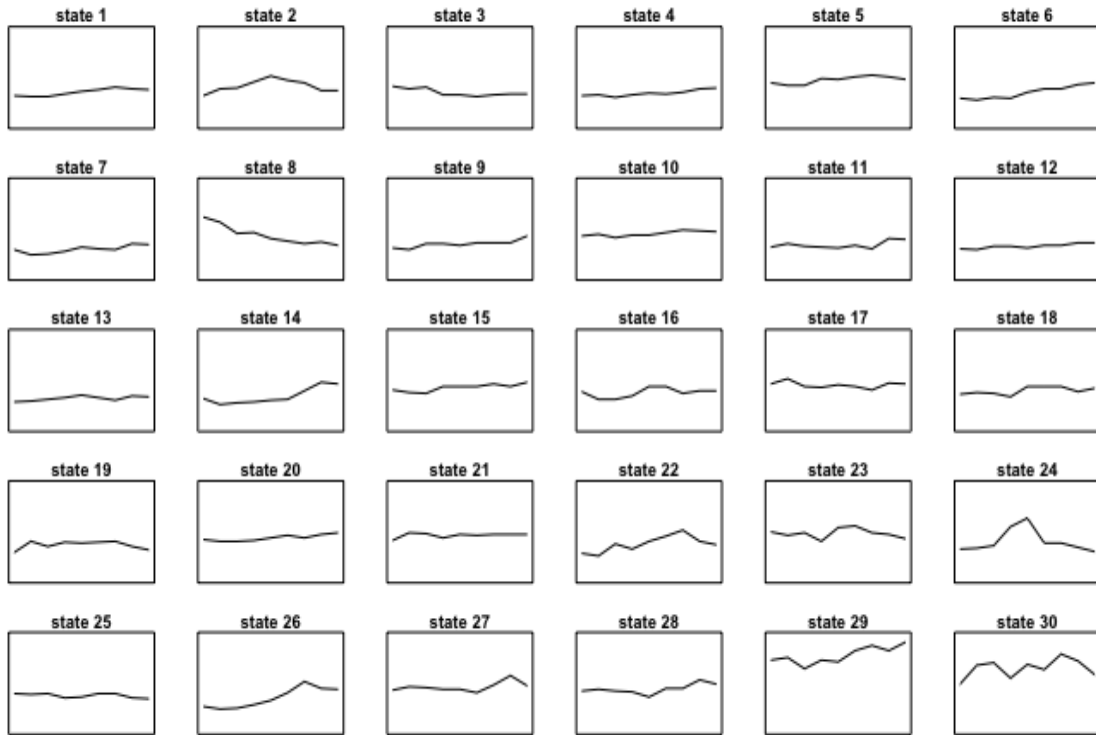


Figure 1: Variation in health expenditure per state over the time period

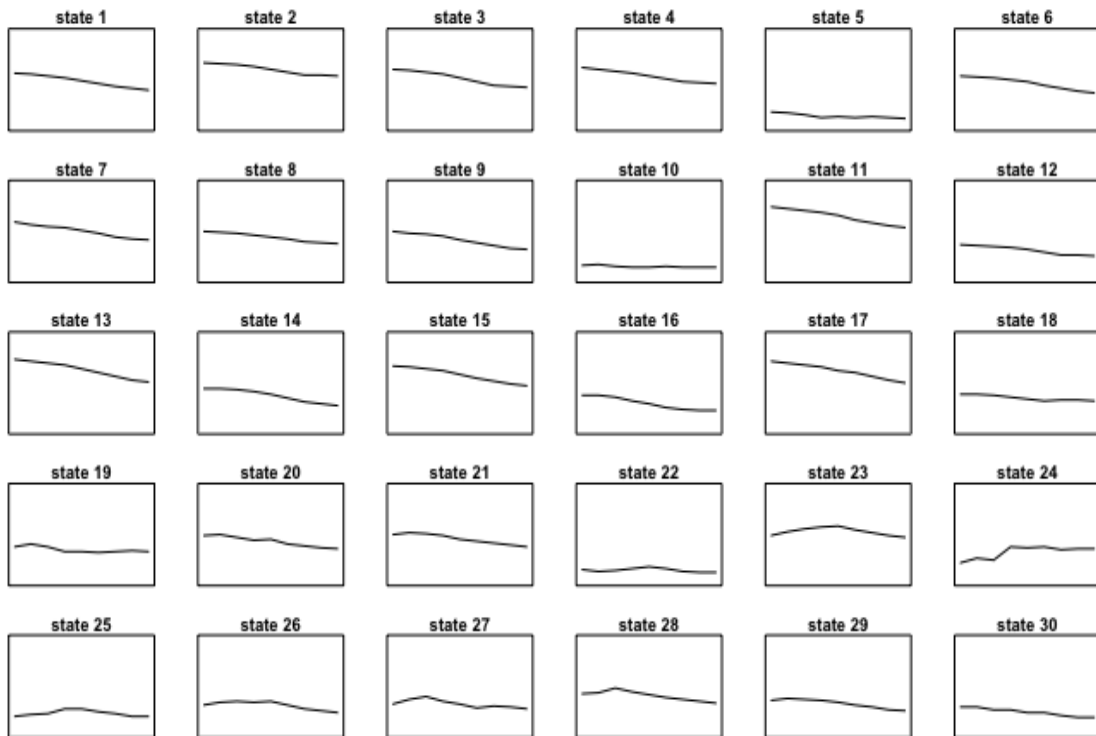


Figure 2: Variation in IMR per state over the time period

Figure 1 shows us that many states have undertaken a higher health expenditure over time while many have remained relatively stagnant. Also Figure 2 reveals that as time has passed most states have seen declining IMR but the absolute numbers seem very different as some states have very sharp declining trends in IMR while some have very slow declining trends.

Figure 3 shows IMR, state health expenditure and literacy rates side by side.

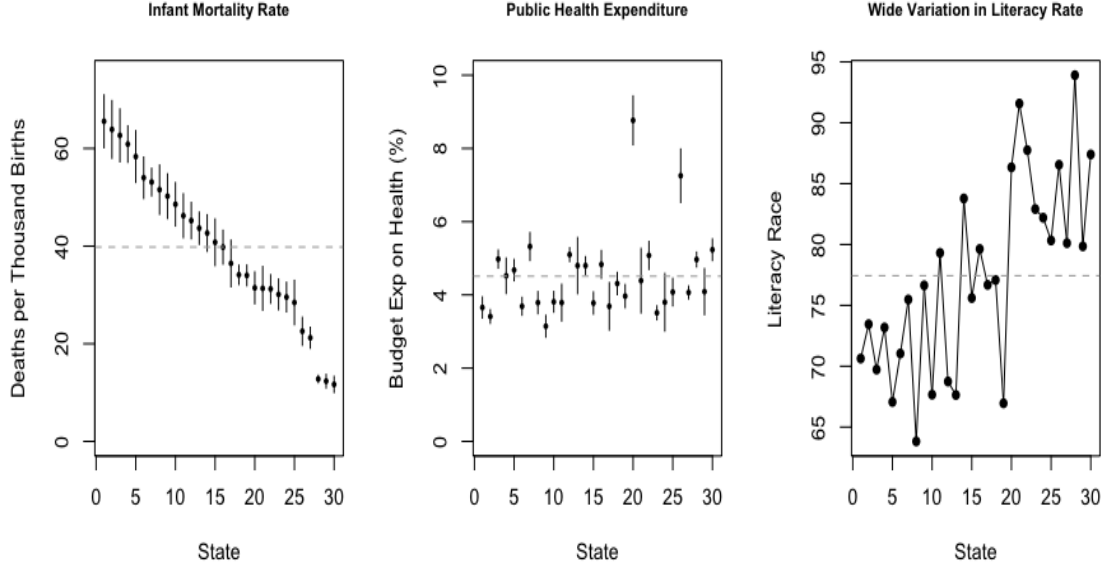


Figure 3: (a)IMR in descending order, (b)Health expenditure over time, (c) Literacy rates vary substantially.

When we view the IMR in descending order and the health expenditure over states we see that there is a very different trend across states and this is a perfect case for Bayesian multilevel modeling. Moreover we see that other factors are important to take into consideration like literacy rate. As expected, on average a higher literacy rate means a lower IMR which makes intuitive sense. We intend to delve deeper into this analysis in the upcoming weeks.

2 Question 2

Read this article: <https://arxiv.org/pdf/1610.05559v1.pdf> and explore this method as follows: Simulate a fake-data scenario of a linear regression with n data points and D predictors, of which $D - p$ are close to zero and p are far from zero. Then fit the horseshoe model and compare your inferences to the true parameter values.

Do this for a range of values of n , D , and p .

I simulate some fake data and fit the horseshoe prior model to my simulated data. At the outset I write a function² that will generate any model with as many parameters close to or far away from 0 as desired. The function take arguments ‘n’ for number of data points, ‘D-p’ predictors close to zero and ‘p’ predictors close to 0. It creates a matrix of data points called ‘X’ which is now the fake data matrix of dimensions $n \times D$. y is a column vector of length ‘n’ and β is coefficient vector for the regression analysis. I also add some noise to the data just so estimates are not too precise. First I provide an exposition of the model itself as taken from (Piironen and Vehtari 2016).

$$y_i \sim N(X\beta, \sigma^2) \quad (1)$$

$$\beta_j | \lambda_j, \tau \sim N(0, \lambda_j^2 \tau^2) \quad (2)$$

$$(3)$$

Now λ and τ are called the *local* and *global* shrinkage parameters respectively. The parameter τ pulls all the weights toward zero while the fat tails of the distribution of lambda will allow some of the parameters to escape this shrinkage. This is ideal for the estimation of sparse parameters. With a very large τ all variables will have very little shrinkage toward 0 and with $\tau \rightarrow 0$ this shrinks all the weights β_j to 0.

I provide hyperparameter λ the following distribution;

$$\lambda_j \sim C^+(0, 1) \quad (4)$$

For τ I provide it three different prior distributions in each case and then fit the regression model with each of the different priors on τ .

$$\tau \sim C^+(0, 1) \quad (5)$$

$$\tau \sim C^+(0, \sigma^2) \quad (6)$$

$$\tau \sim C^+(0, \tau_0^2) \quad (7)$$

Where $\tau_0 = \frac{p_0}{D-p_0} \frac{\sigma}{\sqrt{n}}$. Equation 5 arises from Gelman(2006) and equation 6 is recommended by Polson and Scott(2011). However in equation 7, Piironen and Vehtari (2016) make this important contribution because they like Polson and Scott allow τ to vary with the scale parameter σ and in particular if we have a prior guess for number of relevant parameters p_0 we choose the prior so that most of the mass is located near the value τ_0 .

I run the model for 3 different specifications of n , D and p but each case with each different type of prior on τ . The specifications of each model are given below in Table 1.

Table 1: Three Specifications

Specification	D	p	n
I	6	3	50
II	10	4	50
III	20	15	500

The function I made as discussed earlier forces some β ’s to be very close to 0 and some β ’s to be positive numbers roughly between 1 and 16. So these are the true specifications of the model and the

²see code attached

number of β 's close to and far away from 0 are decided by the simulator however for this assignment only the above specifications have been considered. Table 2 provides a summary of the posterior estimates for β from the 3 specifications. The mean values largely belong to the 95% posterior interval of the estimate of each of the β_j values, however only the mean is reported in Table 2.

Table 2: Posterior estimates of β_j with prior $\tau \sim C^+(0, \tau_0^2)$

β_j	Spec I (Mean)	Spec II (Mean)	Spec III (Mean)
1	-0.18	0.11	-0.02
2	0.05	-0.07	0.01
3	-0.08	-0.09	0.02
4	6.20	-0.06	0.00
5	8.60	0.02	0.01
6	12.62	0.04	9.72
7		22.66	5.74
8		10.86	14.14
9		10.57	11.11
10		16.44	10.78
11			15.14
12			7.56
13			10.34
14			16.14
15			8.40
16			11.42
17			15.37
18			17.15
19			0.06
20			3.04

We observe that the model largely does well and the parameters close to 0 are generally estimated accurately and those far from 0 are estimated convincingly as well. Rhat converges to 1.

Figure 4, 5 and 6 show the posterior distribution of beta along with the true value for each of the model specifications along with the true value across different priors of τ .

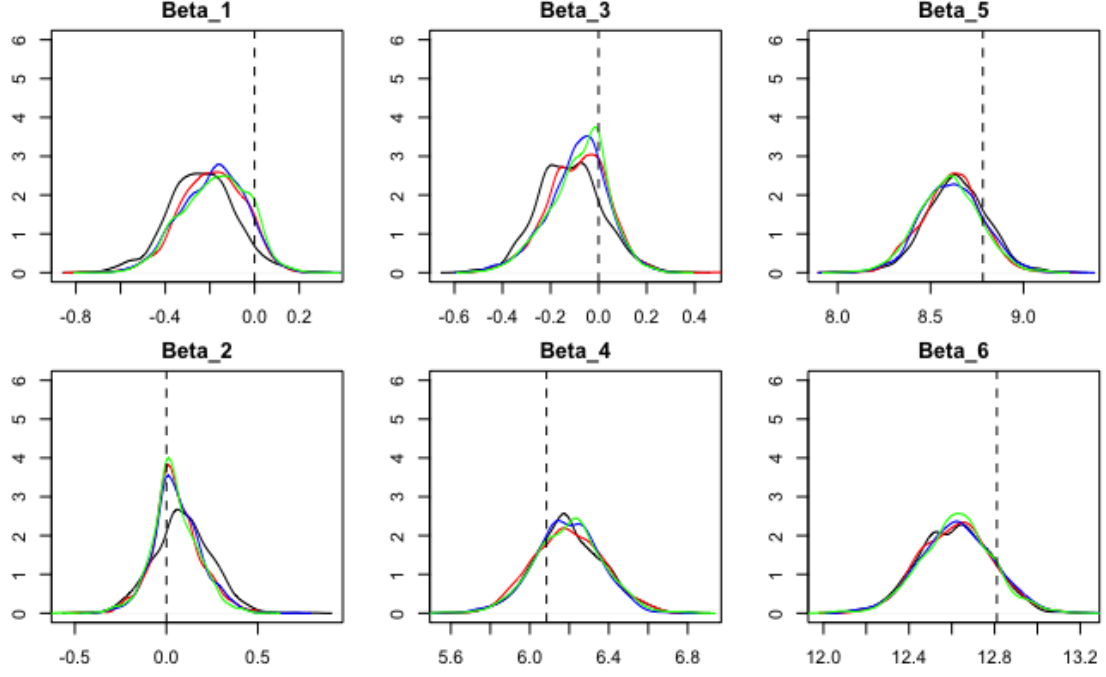


Figure 4: The dotted line shows the true value. The black curve shows the MLE estimate. In red is the estimate with $\tau \sim C^+(0, 1)$, in blue is the estimate with $\tau \sim C^+(0, \sigma)$, in green is the estimate with $\tau \sim C^+(0, \tau_0)$

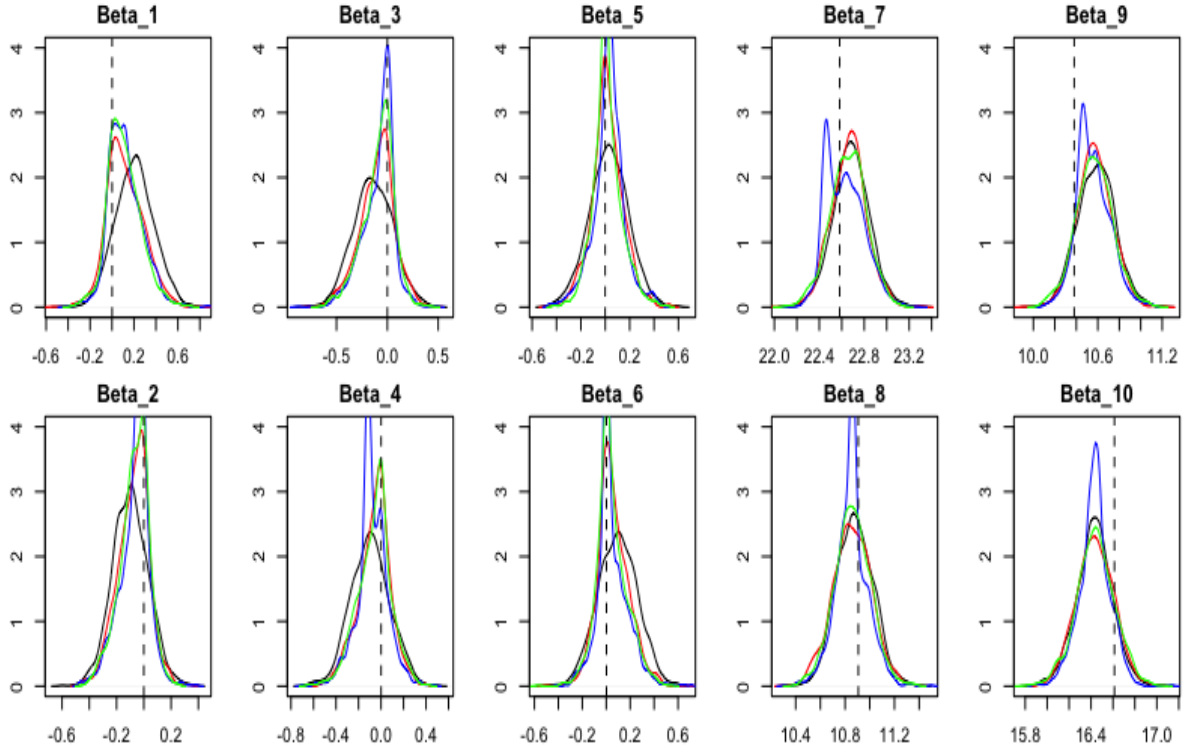


Figure 5: The dotted line shows the true value. The black curve shows the MLE estimate. In red is the estimate with $\tau \sim C^+(0, 1)$, in blue is the estimate with $\tau \sim C^+(0, \sigma)$, in green is the estimate with $\tau \sim C^+(0, \tau_0)$

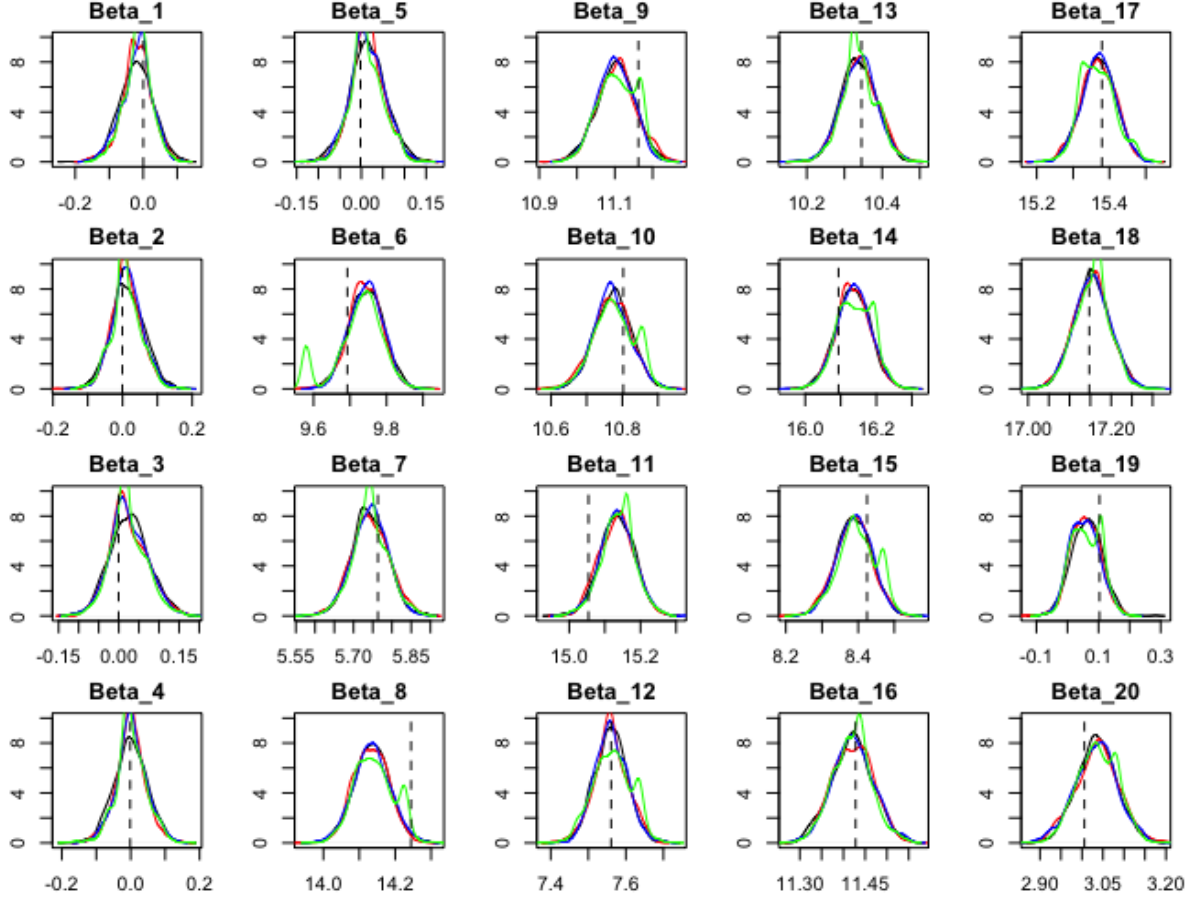


Figure 6: The dotted line shows the true value. The black curve shows the MLE estimate. In red is the estimate with $\tau \sim C^+(0, 1)$, in blue is the estimate with $\tau \sim C^+(0, \sigma)$, in green is the estimate with $\tau \sim C^+(0, \tau_0)$

We observe that the estimates with priors on τ consistently perform better than the MLE estimation of the parameters and in particular the green curve generally does the best job out of the estimates of β with priors on τ .

3 Code

Listing 1: R Code

```
rm(list = ls())
setwd("/Users/Advait/Desktop/New_School/Fall16/BDA/Class13")
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())

##Question 1 - India
library(plyr)

#Read in Data
ind <- read.csv("paneldata_csv.csv", header = TRUE)
str(ind)

#Plot Health Exp by state over time
inf.state.time <- ddply(ind, .(State.no., Time), summarise,
                        inf = mean(IMR),
                        exp = mean(state_exp))

par(mfrow = c(5, 6), mar = c(1, 1, 1, 1))
for(i in 1:30){
  plot(inf.state.time$Time[inf.state.time$State.no. == i],
       inf.state.time$exp[inf.state.time$State.no == i],
       type = "l",
       ylim = c(0, 11),
       yaxt = "n", xaxt = "n",
       main = paste("state", i), cex.main = 0.9)
}

#Plot IMR by state over time
par(mfrow = c(5, 6), mar = c(1, 1, 1, 1))
for(i in 1:30){
  plot(inf.state.time$Time[inf.state.time$State.no. == i],
       inf.state.time$inf[inf.state.time$State.no == i],
       type = "l",
       ylim = c(0, 100),
       yaxt = "n", xaxt = "n",
       main = paste("state", i), cex.main = 0.9)
}
```


Listing 2: R code contd.

```
#Plot IMR and health exp by state
inf.state <- ddply(ind, .(State.no.), summarise,
  inf = mean(IMR),
  exp = mean(state_exp),
  se.imr = sqrt(sd(IMR)^2/length(State.no.)),
  se.exp = sqrt(sd(state_exp)^2/length(State.no.)),
  lit = mean(lit_rate_I),
  se.lit = sqrt(sd(lit_rate_I)^2/length(State.no.)))

inf.state <- inf.state[order(inf.state$inf),]
inf.state$State.no. <- c(30:1)

par(mfrow = c(1, 3), mar = c(4, 3.9, 3.8, 2), cex = 0.9)
plot(inf.state$State.no., inf.state$inf,
  pch = 16, cex = 0.6,
  ylab = "Deaths_per_Thousand_Births",
  xlab = "State",
  main = "Infant_Mortality_Rate",
  ylim = c(0, 75),
  cex.main = 0.8)
arrows(inf.state$State.no., inf.state$inf + 2*inf.state$se.imr,
  inf.state$State.no., inf.state$inf - 2*inf.state$se.imr,
  length = 0)
abline(h = mean(ind$IMR), lty = 2, col = "darkgrey")

plot(inf.state$State.no., inf.state$exp,
  pch = 16, cex = 0.6,
  ylab = "Budget_Exp_on_Health_(%)",
  xlab = "State",
  main = "Public_Health_Expenditure",
  ylim = c(0, 10),
  cex.main = 0.8)
arrows(inf.state$State.no., inf.state$exp + 2*inf.state$se.exp,
  inf.state$State.no., inf.state$exp - 2*inf.state$se.exp,
  length = 0)
abline(h = mean(ind$state_exp), lty = 2, col = "darkgrey")
```

Listing 3: R code contd.

```
#Plot Literacy Rate by State
#par(mfrow = c(1, 1), mar = c(4, 4, 3, 2))
plot(ind.state$State.no., ind.state$lit,
     pch = 16, cex = 0.6,
     ylab = "Literacy_Rate",
     xlab = "State",
     main = "Wide_Variation_in_Literacy_Rate", type = "l",
     cex.main = 0.8)
points(ind.state$State.no., ind.state$lit, pch = 16)
arrows(ind.state$State.no., ind.state$lit + 2*ind.state$se.lit,
       ind.state$State.no., ind.state$lit - 2*ind.state$se.lit,
       length = 0)
abline(h = mean(ind$lit_rate_I), lty = 2, col = "darkgrey")

##Question 2##
#Generating fake data function#
random_gen <- function(lengthb0, lengthb1, numberx){
  b0 <- NULL
  b1 <- NULL
  y <- NULL
  noise <- NULL
  coeff <- NULL
  snoopx <- matrix(NA, nrow = numberx, ncol = (lengthb0 + lengthb1))
  for(i in 1:numberx){
    for(j in 1:(lengthb0+lengthb1)){
      snoopx[i,j] <- rnorm(1)
    }
  }
  step2 <- for(i in 1:lengthb0){
    b0[i] <- rnorm(1,0,0.001 )
  }
  step3 <- for (i in 1:lengthb1){
    b1[i] <- rnorm(1,8,5)
  }
  step4 <- for(i in 1:numberx){
    noise[i] <- rnorm(1)
  }
  beta <- matrix(c(b0,b1),nrow = (lengthb0 + lengthb1) , ncol = 1 )
  y <- (snoopx) %*% beta + noise
  coeff <- beta
  return(beta)
}
```

Listing 4: R code contd.

```
###TRY 1
# n = 50
# numberx => D = 6
# far from 0 => p = 3
# close to 0 => D-p = 3
random_gen(3,3,50)
nc <- 6
nr <- 50
X <- snoopx
y <- as.vector(y)
coeff <- as.vector(coeff)
stanc("7b.stan")

## MLE Estimate
fit1_mle <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit1_mle)
ext1_mle <- extract(fit1_mle)

## Horseshoe prior
fit1_h <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
              iter = 1000, chains = 3)
print(fit1_h)
ext1_h <- extract(fit1_h)

## Horseshoe prior - scale tau
fit1_h2 <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit1_h2)
ext1_h2 <- extract(fit1_h2)

## Horseshoe tau0
fit1_h3 <- stan("7b2.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit1_h3)
ext1_h3 <- extract(fit1_h3)

##Plot1
par(mfcol = c(2,3), mar = c(2,2,2,2))
```

Listing 5: R Code contd.

```
##Plots try 1##  
plot(density(ext1_mle$beta[,1]), main = "Beta_1", ylim = c(0,6))  
lines(density(ext1_h$beta[,1]), col = "red")  
lines(density(ext1_h2$beta[,1]), col = "blue")  
lines(density(ext1_h3$beta[,1]), col = "green")  
abline(v = coeff[1], lty = 2 )  
  
plot(density(ext1_mle$beta[,2]), main = "Beta_2", ylim = c(0,6))  
lines(density(ext1_h$beta[,2]), col = "red")  
lines(density(ext1_h2$beta[,2]), col = "blue")  
lines(density(ext1_h3$beta[,2]), col = "green")  
abline(v = coeff[2], lty = 2 )  
  
plot(density(ext1_mle$beta[,3]), main = "Beta_3", ylim = c(0,6))  
lines(density(ext1_h$beta[,3]), col = "red")  
lines(density(ext1_h2$beta[,3]), col = "blue")  
lines(density(ext1_h3$beta[,3]), col = "green")  
abline(v = coeff[3], lty = 2 )  
  
plot(density(ext1_mle$beta[,4]), main = "Beta_4", ylim = c(0,6))  
lines(density(ext1_h$beta[,4]), col = "red")  
lines(density(ext1_h2$beta[,4]), col = "blue")  
lines(density(ext1_h3$beta[,4]), col = "green")  
abline(v = coeff[4], lty = 2 )  
  
plot(density(ext1_mle$beta[,5]), main = "Beta_5", ylim = c(0,6))  
lines(density(ext1_h$beta[,5]), col = "red")  
lines(density(ext1_h2$beta[,5]), col = "blue")  
lines(density(ext1_h3$beta[,5]), col = "green")  
abline(v = coeff[5], lty = 2 )  
  
plot(density(ext1_mle$beta[,6]), main = "Beta_6", ylim = c(0,6))  
lines(density(ext1_h$beta[,6]), col = "red")  
lines(density(ext1_h2$beta[,6]), col = "blue")  
lines(density(ext1_h3$beta[,6]), col = "green")  
abline(v = coeff[6], lty = 2 )
```

Listing 6: R code contd.

```
###TRY 2
# n = 50
# numberx => D = 10
# far from 0 => p = 4
# close to 0 => D-p = 6
random_gen(6,4,50)
nc <- 10
nr <- 50
X <- snoopx
y <- as.vector(y)
coeff <- as.vector(coeff)
stanc("7b.stan")

## MLE Estimate
fit2_mle <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit2_mle)
ext2_mle <- extract(fit2_mle)

## Horseshoe prior
fit2_h <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
              iter = 1000, chains = 3)
print(fit2_h)
ext2_h <- extract(fit2_h)

## Horseshoe prior - scale tau
fit2_h2 <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit2_h2)
ext2_h2 <- extract(fit2_h2)

## Horseshoe tau0
fit2_h3 <- stan("7b2.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit2_h3)
ext2_h3 <- extract(fit2_h3)
##Plot2
par(mfcol = c(2,5), mar = c(2,2,2,2))
#
```

Listing 7: R code contd.

```
plot(density(ext2_mle$beta[,1]), main = "Beta_1", ylim = c(0,4))
lines(density(ext2_h$beta[,1]), col = "red")
lines(density(ext2_h2$beta[,1]), col = "blue")
lines(density(ext2_h3$beta[,1]), col = "green")
abline(v = coeff[1], lty = 2 )

plot(density(ext2_mle$beta[,2]), main = "Beta_2", ylim = c(0,4))
lines(density(ext2_h$beta[,2]), col = "red")
lines(density(ext2_h2$beta[,2]), col = "blue")
lines(density(ext2_h3$beta[,2]), col = "green")
abline(v = coeff[2], lty = 2 )

plot(density(ext2_mle$beta[,3]), main = "Beta_3", ylim = c(0,4))
lines(density(ext2_h$beta[,3]), col = "red")
lines(density(ext2_h2$beta[,3]), col = "blue")
lines(density(ext2_h3$beta[,3]), col = "green")
abline(v = coeff[3], lty = 2 )

plot(density(ext2_mle$beta[,4]), main = "Beta_4", ylim = c(0,4))
lines(density(ext2_h$beta[,4]), col = "red")
lines(density(ext2_h2$beta[,4]), col = "blue")
lines(density(ext2_h3$beta[,4]), col = "green")
abline(v = coeff[4], lty = 2 )

plot(density(ext2_mle$beta[,5]), main = "Beta_5", ylim = c(0,4))
lines(density(ext2_h$beta[,5]), col = "red")
lines(density(ext2_h2$beta[,5]), col = "blue")
lines(density(ext2_h3$beta[,5]), col = "green")
abline(v = coeff[5], lty = 2 )

plot(density(ext2_mle$beta[,6]), main = "Beta_6", ylim = c(0,4))
lines(density(ext2_h$beta[,6]), col = "red")
lines(density(ext2_h2$beta[,6]), col = "blue")
lines(density(ext2_h3$beta[,6]), col = "green")
abline(v = coeff[6], lty = 2 )

plot(density(ext2_mle$beta[,7]), main = "Beta_7", ylim = c(0,4))
lines(density(ext2_h$beta[,7]), col = "red")
lines(density(ext2_h2$beta[,7]), col = "blue")
lines(density(ext2_h3$beta[,7]), col = "green")
abline(v = coeff[7], lty = 2 )
```

Listing 8: R code contd.

```
plot(density(ext2_mle$beta[,8]), main = "Beta_8", ylim = c(0,4))
lines(density(ext2_h$beta[,8]), col = "red")
lines(density(ext2_h2$beta[,8]), col = "blue")
lines(density(ext2_h3$beta[,8]), col = "green")
abline(v = coeff[8], lty = 2 )

plot(density(ext2_mle$beta[,9]), main = "Beta_9", ylim = c(0,4))
lines(density(ext2_h$beta[,9]), col = "red")
lines(density(ext2_h2$beta[,9]), col = "blue")
lines(density(ext2_h3$beta[,9]), col = "green")
abline(v = coeff[9], lty = 2 )

plot(density(ext2_mle$beta[,10]), main = "Beta_10", ylim = c(0,4))
lines(density(ext2_h$beta[,10]), col = "red")
lines(density(ext2_h2$beta[,10]), col = "blue")
lines(density(ext2_h3$beta[,10]), col = "green")
abline(v = coeff[10], lty = 2 )

###TRY 6
# n = 500
# numberx => D = 20
# far from 0 => p = 15
# close to 0 => D-p = 5

random_gen(5,15,500)
nc <- 20
nr <- 500
X <- snoopx
y <- as.vector(y)
coeff <- as.vector(coeff)
stanc("7b.stan")
## MLE Estimate
fit6_mle <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit6_mle)
ext6_mle <- extract(fit6_mle)
## Horseshoe prior
fit6_h <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
               iter = 1000, chains = 3)
print(fit6_h)
ext6_h <- extract(fit6_h)
```

Listing 9: R code contd.

```
## Horseshoe prior - scale tau
fit6_h2 <- stan("7b.stan", data = list("X", "y", "nr", "nc"),
              iter = 1000, chains = 3)
print(fit6_h2)
ext6_h2 <- extract(fit6_h2)
# Horseshoe tau0
fit6_h3 <- stan("7b2.stan", data = list("X", "y", "nr", "nc"),
              iter = 1000, chains = 3)
print(fit6_h3)
ext6_h3 <- extract(fit6_h3)
##
par(mfcol = c(4,5), mar = c(2,2,2,2))

plot(density(ext6_mle$beta[,1]), main = "Beta_1", ylim = c(0,10))
lines(density(ext6_h$beta[,1]), col = "red")
lines(density(ext6_h2$beta[,1]), col = "blue")
lines(density(ext6_h3$beta[,1]), col = "green")
abline(v = coeff[1], lty = 2 )

plot(density(ext6_mle$beta[,2]), main = "Beta_2", ylim = c(0,10))
lines(density(ext6_h$beta[,2]), col = "red")
lines(density(ext6_h2$beta[,2]), col = "blue")
lines(density(ext6_h3$beta[,2]), col = "green")
abline(v = coeff[2], lty = 2 )

plot(density(ext6_mle$beta[,3]), main = "Beta_3", ylim = c(0,10))
lines(density(ext6_h$beta[,3]), col = "red")
lines(density(ext6_h2$beta[,3]), col = "blue")
lines(density(ext6_h3$beta[,3]), col = "green")
abline(v = coeff[3], lty = 2 )

plot(density(ext6_mle$beta[,4]), main = "Beta_4", ylim = c(0,10))
lines(density(ext6_h$beta[,4]), col = "red")
lines(density(ext6_h2$beta[,4]), col = "blue")
lines(density(ext6_h3$beta[,4]), col = "green")
abline(v = coeff[4], lty = 2 )

plot(density(ext6_mle$beta[,5]), main = "Beta_5", ylim = c(0,10))
lines(density(ext6_h$beta[,5]), col = "red")
lines(density(ext6_h2$beta[,5]), col = "blue")
lines(density(ext6_h3$beta[,5]), col = "green")
abline(v = coeff[5], lty = 2 )
```


Listing 10: R code contd.

```
plot(density(ext6_mle$beta[,6]), main = "Beta_6", ylim = c(0,10))
lines(density(ext6_h$beta[,6]), col = "red")
lines(density(ext6_h2$beta[,6]), col = "blue")
lines(density(ext6_h3$beta[,6]), col = "green")
abline(v = coeff[6], lty = 2 )

plot(density(ext6_mle$beta[,7]), main = "Beta_7", ylim = c(0,10))
lines(density(ext6_h$beta[,7]), col = "red")
lines(density(ext6_h2$beta[,7]), col = "blue")
lines(density(ext6_h3$beta[,7]), col = "green")
abline(v = coeff[7], lty = 2 )

plot(density(ext6_mle$beta[,8]), main = "Beta_8", ylim = c(0,10))
lines(density(ext6_h$beta[,8]), col = "red")
lines(density(ext6_h2$beta[,8]), col = "blue")
lines(density(ext6_h3$beta[,8]), col = "green")
abline(v = coeff[8], lty = 2 )

plot(density(ext6_mle$beta[,9]), main = "Beta_9", ylim = c(0,10))
lines(density(ext6_h$beta[,9]), col = "red")
lines(density(ext6_h2$beta[,9]), col = "blue")
lines(density(ext6_h3$beta[,9]), col = "green")
abline(v = coeff[9], lty = 2 )

plot(density(ext6_mle$beta[,10]), main = "Beta_10", ylim = c(0,10))
lines(density(ext6_h$beta[,10]), col = "red")
lines(density(ext6_h2$beta[,10]), col = "blue")
lines(density(ext6_h3$beta[,10]), col = "green")
abline(v = coeff[10], lty = 2 )

plot(density(ext6_mle$beta[,11]), main = "Beta_11", ylim = c(0,10))
lines(density(ext6_h$beta[,11]), col = "red")
lines(density(ext6_h2$beta[,11]), col = "blue")
lines(density(ext6_h3$beta[,11]), col = "green")
abline(v = coeff[11], lty = 2 )

plot(density(ext6_mle$beta[,12]), main = "Beta_12", ylim = c(0,10))
lines(density(ext6_h$beta[,12]), col = "red")
lines(density(ext6_h2$beta[,12]), col = "blue")
lines(density(ext6_h3$beta[,12]), col = "green")
abline(v = coeff[12], lty = 2 )
```

Listing 11: R code contd.

```
plot(density(ext6_mle$beta[,13]), main = "Beta_13", ylim = c(0,10))
lines(density(ext6_h$beta[,13]), col = "red")
lines(density(ext6_h2$beta[,13]), col = "blue")
lines(density(ext6_h3$beta[,13]), col = "green")
abline(v = coeff[13], lty = 2 )

plot(density(ext6_mle$beta[,14]), main = "Beta_14", ylim = c(0,10))
lines(density(ext6_h$beta[,14]), col = "red")
lines(density(ext6_h2$beta[,14]), col = "blue")
lines(density(ext6_h3$beta[,14]), col = "green")
abline(v = coeff[14], lty = 2 )

plot(density(ext6_mle$beta[,15]), main = "Beta_15", ylim = c(0,10))
lines(density(ext6_h$beta[,15]), col = "red")
lines(density(ext6_h2$beta[,15]), col = "blue")
lines(density(ext6_h3$beta[,15]), col = "green")
abline(v = coeff[15], lty = 2 )

plot(density(ext6_mle$beta[,16]), main = "Beta_16", ylim = c(0,10))
lines(density(ext6_h$beta[,16]), col = "red")
lines(density(ext6_h2$beta[,16]), col = "blue")
lines(density(ext6_h3$beta[,16]), col = "green")
abline(v = coeff[16], lty = 2 )

plot(density(ext6_mle$beta[,17]), main = "Beta_17", ylim = c(0,10))
lines(density(ext6_h$beta[,17]), col = "red")
lines(density(ext6_h2$beta[,17]), col = "blue")
lines(density(ext6_h3$beta[,17]), col = "green")
abline(v = coeff[17], lty = 2 )

plot(density(ext6_mle$beta[,18]), main = "Beta_18", ylim = c(0,10))
lines(density(ext6_h$beta[,18]), col = "red")
lines(density(ext6_h2$beta[,18]), col = "blue")
lines(density(ext6_h3$beta[,18]), col = "green")
abline(v = coeff[18], lty = 2 )

plot(density(ext6_mle$beta[,19]), main = "Beta_19", ylim = c(0,10))
lines(density(ext6_h$beta[,19]), col = "red")
lines(density(ext6_h2$beta[,19]), col = "blue")
lines(density(ext6_h3$beta[,19]), col = "green")
abline(v = coeff[19], lty = 2 )
```

Listing 12: R and Stan code

```
plot(density(ext6_mle$beta[,20]), main = "Beta_20", ylim = c(0,10))
lines(density(ext6_h$beta[,20]), col = "red")
lines(density(ext6_h2$beta[,20]), col = "blue")
lines(density(ext6_h3$beta[,20]), col = "green")
abline(v = coeff[20], lty = 2 )

##STAN Code##
## Model with MLE, and without tau_0##
data{
  int nr;
  int nc;
  matrix[nr,nc] X;
  vector[nr] y;
}
parameters{
  vector[nc] beta;
  vector<lower = 0>[nc] lambda;
  real <lower = 0> tau;
  real<lower = 0> sigma;
}
model{
  y ~ normal(X*beta, sigma);
  beta ~ normal(0,lambda*tau);
  tau ~ cauchy(0,1);
  // tau ~ cauchy(0,sigma);
  lambda ~ cauchy(0,1);
}
## Model with tau_0##
data{
  int nr;
  int nc;
  matrix[nr,nc] X;
  vector[nr] y;
}
parameters{
  vector[nc] beta;
  vector<lower = 0>[nc] lambda;
  real <lower = 0> tau;
  real<lower = 0> sigma;
}
```

Listing 13: Stan code

```
transformed parameters{
  real<lower = 0> tau01;
  // real<lower = 0> tau02;
  // real<lower = 0> tau06;
  tau01 = sigma/sqrt(50);
  // tau02 = 0.67*(sigma/sqrt(50));
  // tau06 = 3*(sigma/sqrt(500));
}
model{
  y ~ normal(X*beta, sigma);
  beta ~ normal(0,lambda*tau);
  tau ~ cauchy(0,tau01);
  // tau ~ cauchy(0,tau02);
  // tau ~ cauchy(0,tau06);
  lambda ~ cauchy(0,1);
}
```