Assignment 8.a for **STATGR6103**

submitted to Professor Andrew Gelman

Advait Rajagopal

26 October 2016

1 Question 1

With your partner, write up a plan for your final project. This plan should include some plots.

1.1 The data

As discussed in Assignment 7.b we build a data set that has the following variables collected from different public sources of data. The infant mortality rate (IMR) comes from vital statistics data in the Indian census. The health expenditure data is retrieved from annual budget publications of data which are made available to the public by the Reserve Bank of India. The other variables are relatively easy to find through the Indian census numbers.
So we built a dataset that has the following variables, for 30 states[1] 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.

1.2 Why this project and what we're trying to learn?

We chose this project due to the clear hierarchical structure in the data and the overwhelming need to use a Bayesian multilevel approach to capture differences in the impact of health expenditure on IMR across states and across time, while controlling for several other important indicators of social development and welfare. For instance, the number of females per 1000 males is a ratio that has often been worrisome in many states. This is due to chronic infanticide and a systemic dislike for

[1] We 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

girl children in many Indian states. Moreover this ratio varies substantially from the more rural agricultural states with lower GDP's to industrialized states with a robust manufacturing and service sector and higher GDP's. There are several angles through which we could regard the problem and our aim is to capture the multilevel variation in the data. The learning outcome of this project/model would be to really deal with hierarchical structures in an applied problem. Countries and policy makers are often looking for some universal policy that will help the country as a whole and capturing state level and time level variation will allow us to argue that there is no real top down enforcement alone which will be successful but there needs to be state specific policy. Moreover we want to analyze what the variance in IMR looks like as we keep increasing state health expenditure. At the outset we have some reason to believe that there is a floor below which IMR cannot drop regardless of how high health expenditure is increased.

1.3 Possible Models and Plots

If $y_{st} = IMR_{st}$ such that IMR_{st} represents the infant mortality rate in state 's' at time 't', and X_{st} signifies the proportion of health expenditure in total GDP, in state 's' and time 't', the exposition of the model is as follows;

$$y_{st} \sim \mathbf{N}(X_{st}\beta + \gamma_{st} + \delta_t, \sigma^2)$$

$$\gamma_{st} \sim \mathbf{N}(0, \tau_\gamma^2)$$

$$\delta_t \sim \mathbf{N}(0, \tau_\delta^2)$$

In the above description of the model β, σ^2, τ_γ^2, τ_δ^2 have non informative hyperpriors meaning they follow a uniform distribution on the real space. This model is somewhat similar to example 15.2 in BDA 3. This immediately shows that β will have a noninformative prior, and will be a pooled estimate. We would further like to allow beta to vary by state and perform a partially pooled hierarchical model to estimate the coefficient β meaning we impose the following prior on β;

$$\beta_j \sim \mathbf{N}(\mu, \tau_\beta^2)$$

Figure 1 shows IMR, state health expenditure and literacy rates side by side.

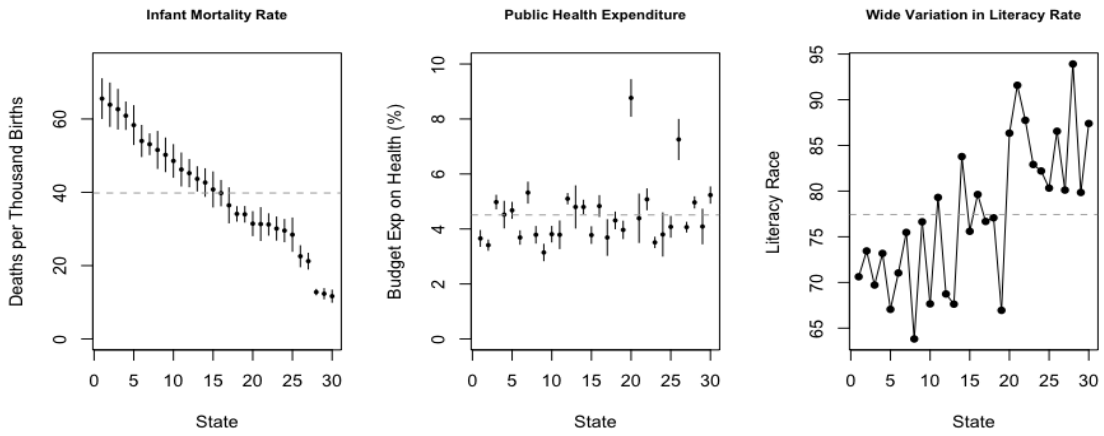

Figure 1: (a)IMR in descending order, (b)Health expenditure over time, (c) Literacy rates vary substantially.

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.
It is extremely important we consider a non-linear relationship between IMR and state health expenditure more importantly we want our variance to decrease for higher levels of expenditure. This makes the likelihood;

$$y_{st} \sim \Gamma(\alpha_s, \beta_s)$$

where;

$$\alpha_s = \alpha_0 + \alpha_1 * x_{st}$$
$$\beta_s = \beta_0 + \beta_1 * x_{st}$$

A quick exploratory plot is shown below in Figure 2, but there is work to be done on this model as variance is clearly not decreasing and the interval is too wide! We intend to pick up our analysis here.

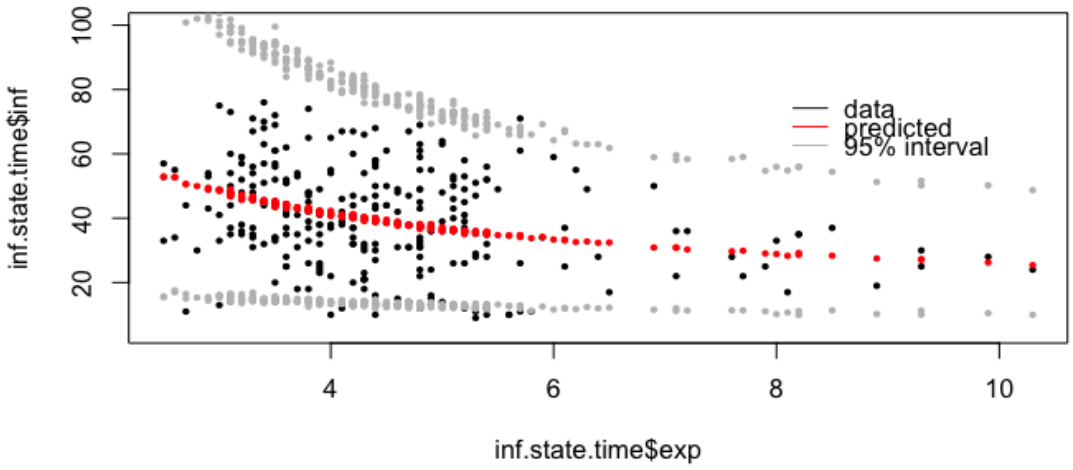

Figure 2: A gamma distribution experiment

2 Question 2

Repeat Exercise 2 from previous assignment with logistic instead of linear regression
I repeat Question 2 from Assignment 7.b using a logistic regression instead of a linear regression. The horseshoe prior I use this time is the prior that does not scale τ with variance in the data as I use a logistic regression model here. The likelihood function is given by the equation;

$$p(y_i|\theta_i, \beta_i, \lambda, \tau, n, X) \propto [\theta_i]^{y_i}[1-\theta_i]^{n-y_i} \quad (1)$$

where θ is the probability of a success, X is the matrix of data, β is the vector of coefficients, λ and τ are hyperparameters, and n is the number of data points such that;

$$\theta_i = \frac{e^{(X*\beta_i)}}{1 + e^{(X*\beta_i)}} \tag{2}$$

and the prior distributions are given by;

$$\beta_j|\lambda_j, \tau \sim N(0, \lambda_j^2\tau^2) \tag{3}$$

$$\lambda_j \sim C^+(0, 1) \tag{4}$$

$$\tau \sim C^+(0, 1) \tag{5}$$

I run 4 models with different specifications. The specifications of n, D and p are listed in Table 1.

Table 1: Three Specifications

Specification	D	p	n
I	6	3	50
II	10	4	50
III	20	15	500
IV	20	5	500

The estimates of the coefficients are shown in Figure 3 for all four models,

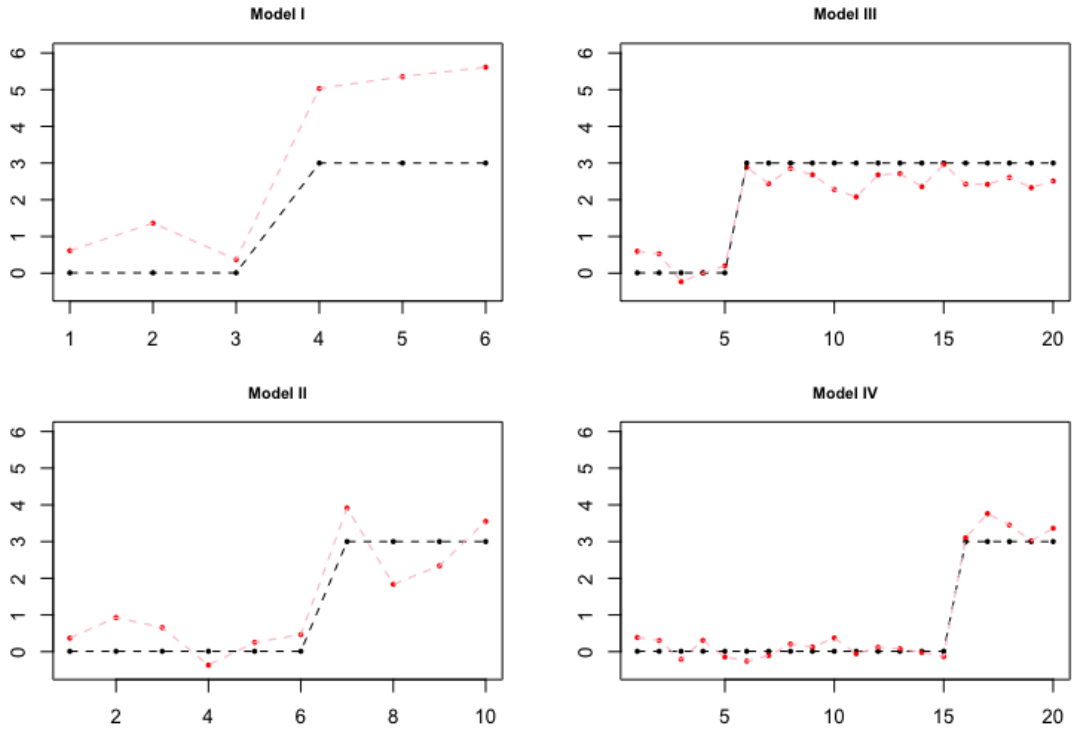

Figure 3: The black dots show the original values of β. The red dots show the estimates of the same β using the logistic regression model. The dotted lines are to aid visual representation of how far above or below the true value the estimate lies.

I observe that my model performs well consistently and estimates β very close to the true value. In Figures 4,5,6 and 7 I show the posterior density of the coefficient of regression along with the true value in red.

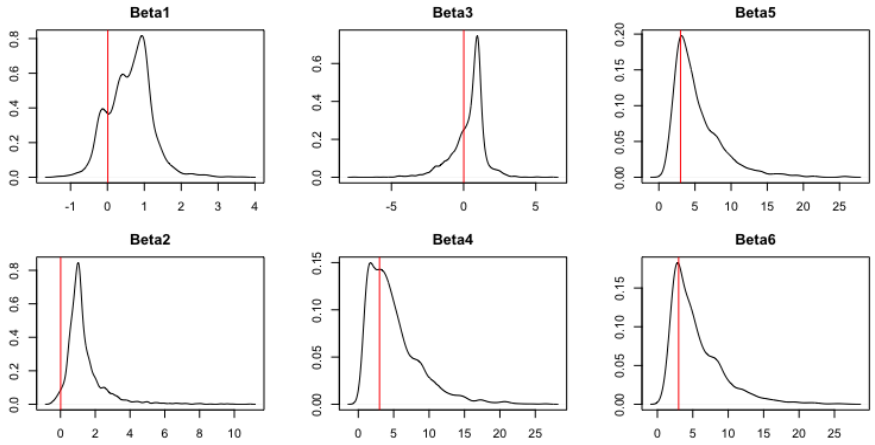

Figure 4: Model I

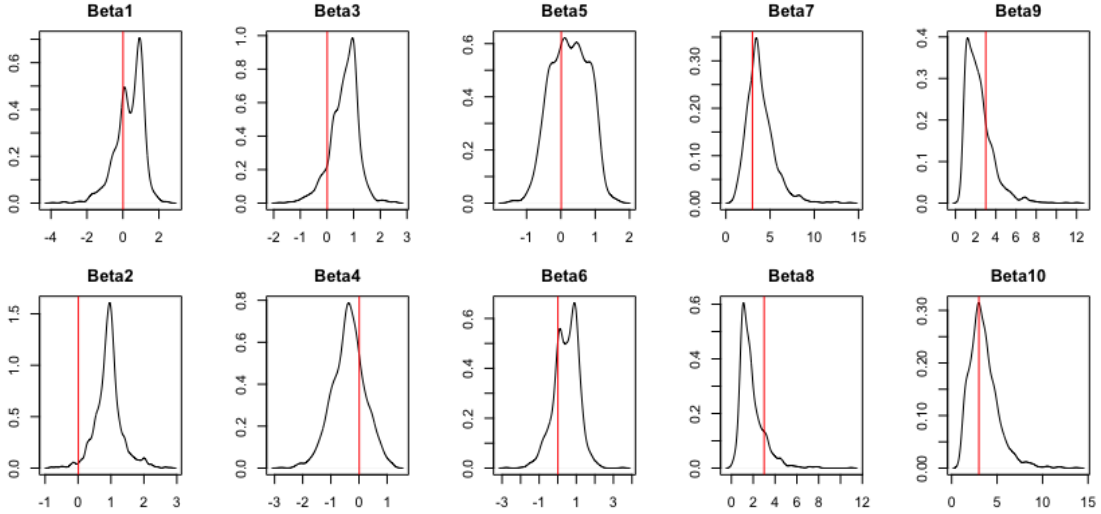

Figure 5: Model II

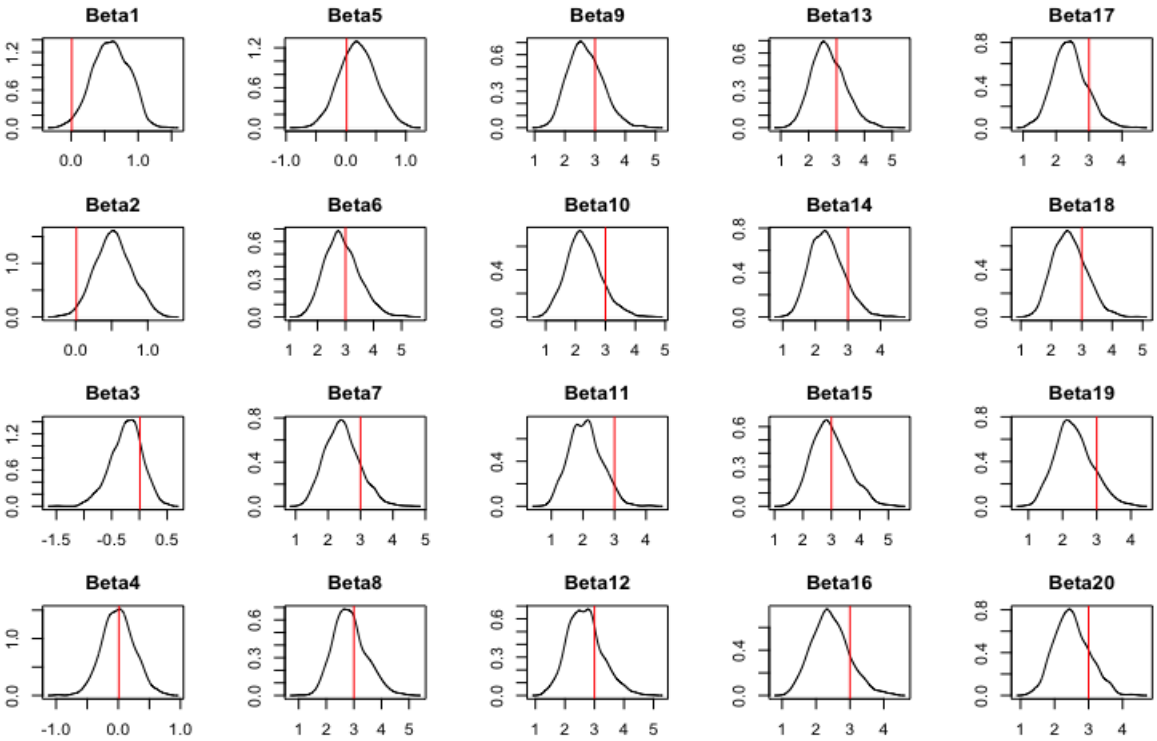

Figure 6: Model III

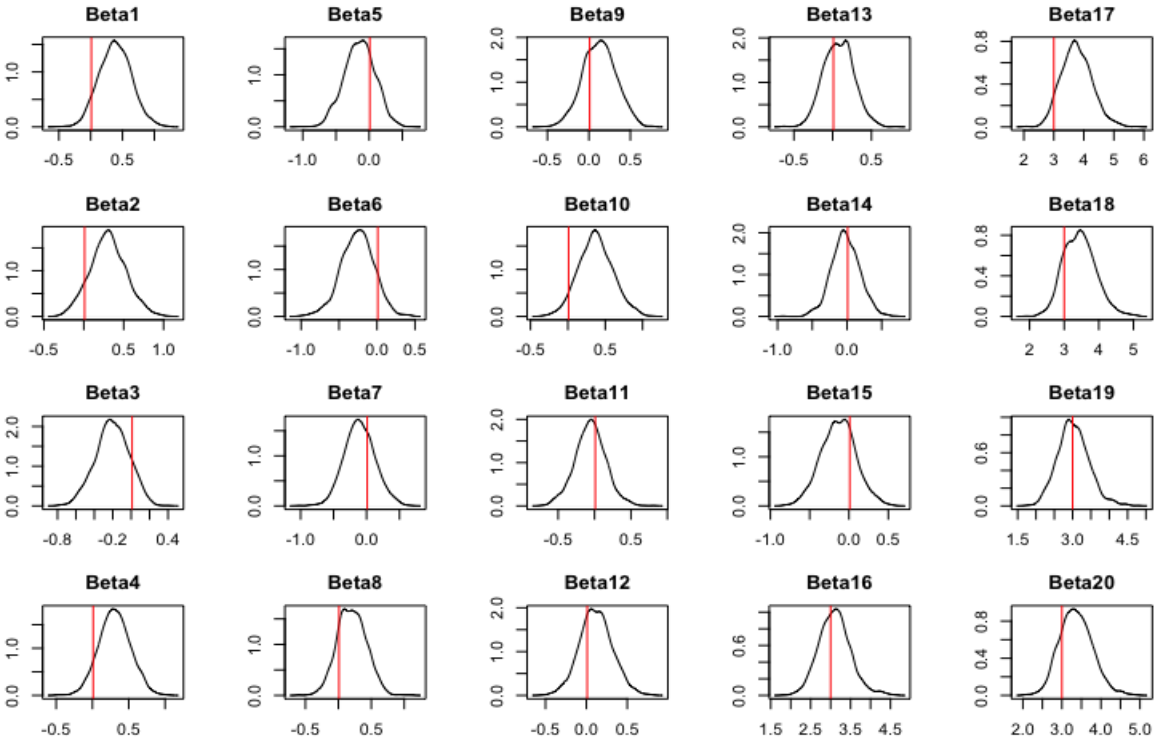

Figure 7: Model IV

3 Code

3.1 R Code

Listing 1: R Code

```
##Question 1
rm(list = ls())
setwd("/Users/Advait/Desktop/New School/Fall16/BDA/Class14")
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))
str(inf.state.time)
###TRY1
par(mfcol = c(1,1))
x <- inf.state.time$exp
y <- inf.state.time$inf
length(x)
stanc("8atrial.stan")
projfit <- stan("8atrial.stan", data = list("x", "y"),
                iter = 1000, chains = 3)
ext_proj <- extract(projfit)
y_alex <- colMeans(ext_proj$y_alex)
bar1 <- NULL
bar2 <- NULL
for (i in 1:270){
  bar1[i] <- quantile(ext_proj$y_alex[,i],probs = 0.025 )
  bar2[i] <- quantile(ext_proj$y_alex[,i],probs = 0.975 ) }

plot(inf.state.time$exp,inf.state.time$inf,
     pch = 16,
     cex = .6, ylim = c(5,100))
```

Listing 2: R code contd.

```
  points(inf.state.time$exp, y_alex, col = "red", pch = 16,
       cex = 0.6)
points(inf.state.time$exp, bar1, col = "gray", pch = 16,
       cex = 0.6)
points(inf.state.time$exp, bar2, col = "gray", pch = 16,
       cex = 0.6)
legend(8,80, legend = c("data","predicted","95\% interval"),
       col = c("black", "red","gray"),lwd = 1, bty = "n")
##############
###Question2
##############
rm(list = ls())
setwd("/Users/Advait/Desktop/New School/Fall16/BDA/Class14")
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
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)
  }
```

Listing 3: R code contd.

```
  beta <- matrix(c(rep(0.01,lengthb0), rep(3,lengthb1)),
                 nrow = (lengthb0 + lengthb1) ,
                 ncol = 1)
  trial <<- snoopx \%*\% beta
  prob <<- exp(snoopx \%*\% beta)/ (1+ exp(snoopx \%*\% beta))
  coeff <<- beta
  y <<- rbinom(numberx, 1, prob)
  ###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)
coeff1 <- as.vector(coeff)
fit1 <- stan("8a.stan", data = list("X", "y", "nc", "nr"),
             iter = 2000, chains = 3)
print(fit1)
ext1 <- extract(fit1)
est1 <- colMeans(ext1$beta)

###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)
coeff2 <- as.vector(coeff)
stanc("8a.stan")
fit2 <- stan("8a.stan", data = list("X", "y", "nc", "nr"),
             iter = 1000, chains = 3)
print(fit2)
ext2 <- extract(fit2)
est2 <- colMeans(ext2$beta)
}
```

Listing 4: R Code contd.

```
###TRY 3
# 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)
coeff3 <- as.vector(coeff)
fit3 <- stan("8a.stan", data = list("X", "y", "nc", "nr"),
             iter = 1000, chains = 3)
print(fit3)
ext3 <- extract(fit3)
est3 <- colMeans(ext3$beta)

###TRY 4
# n = 500
# numberx => D = 20
# far from 0 => p = 5
# close to 0 => D-p = 15
random_gen(15,5,500)
nc <- 20
nr <- 500
X <- snoopx
y <- as.vector(y)
coeff4 <- as.vector(coeff)
fit4 <- stan("8a.stan", data = list("X", "y", "nc", "nr"),
             iter = 1000, chains = 3)
print(fit4)
ext4 <- extract(fit4)
est4 <- colMeans(ext4$beta)
```

Listing 5: R code contd.

```
###PLOTS
par(mfcol = c(2,2),mar = c(2.5,2.5,2.5,2.5))
plot(c(1:6),coeff1,
     xlab = "Number␣of␣predictors",
     ylab = "Original/Predicted␣values", pch = 16,
     cex = .6,
     ylim = c(-0.5,6),
     main = "Model␣I",
     cex.main = 0.8)
points(c(1:6), est1, pch = 16, cex = .6, col = "red")
lines(c(1:6), coeff1, lty = 2)
lines(c(1:6), est1, lty = 2, col = "pink")
#
plot(c(1:10),coeff2,
     xlab = "Number␣of␣predictors",
     ylab = "Original/Predicted␣values", pch = 16,
     cex = .6,
     ylim = c(-0.5,6),
     main = "Model␣II",
     cex.main = 0.8)
points(c(1:10), est2, pch = 16, cex = .6, col = "red")
lines(c(1:10), coeff2, lty = 2)
lines(c(1:10), est2, lty = 2, col = "pink")
#
plot(c(1:20),coeff3,
     xlab = "Number␣of␣predictors",
     ylab = "Original/Predicted␣values", pch = 16,
     cex = .6,
     ylim = c(-0.5,6),
     main = "Model␣III",
     cex.main = 0.8)
points(c(1:20), est3, pch = 16, cex = .6, col = "red")
lines(c(1:20), coeff3, lty = 2)
lines(c(1:20), est3, lty = 2, col = "pink")
#
plot(c(1:20),coeff4,
     xlab = "Number␣of␣predictors",
     ylab = "Original/Predicted␣values", pch = 16,
     cex = .6,
     ylim = c(-0.5,6),
     main = "Model␣IV",
     cex.main = 0.8)
```

Listing 6: R code contd.

```
points(c(1:20), est4, pch = 16, cex = .6, col = "red")
lines(c(1:20), coeff4, lty = 2)
lines(c(1:20), est4, lty = 2, col = "pink")

coeff1
par(mfcol = c(2,3))
plot(density(ext1$beta[,1]), main = "Beta1")
abline(v = 0.01, col = "red")
plot(density(ext1$beta[,2]), main = "Beta2")
abline(v = 0.01, col = "red")
plot(density(ext1$beta[,3]), main = "Beta3")
abline(v = 0.01, col = "red")
plot(density(ext1$beta[,4]), main = "Beta4")
abline(v = 3, col = "red")
plot(density(ext1$beta[,5]), main = "Beta5")
abline(v = 3, col = "red")
plot(density(ext1$beta[,6]), main = "Beta6")
abline(v = 3, col = "red")
##
coeff2
par(mfcol = c(2,5))
plot(density(ext2$beta[,1]), main = "Beta1")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,2]), main = "Beta2")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,3]), main = "Beta3")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,4]), main = "Beta4")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,5]), main = "Beta5")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,6]), main = "Beta6")
abline(v = 0.01, col = "red")
plot(density(ext2$beta[,7]), main = "Beta7")
abline(v = 3, col = "red")
plot(density(ext2$beta[,8]), main = "Beta8")
abline(v = 3, col = "red")
plot(density(ext2$beta[,9]), main = "Beta9")
abline(v = 3, col = "red")
plot(density(ext2$beta[,10]), main = "Beta10")
abline(v = 3, col = "red")
```

Listing 7: R code contd.

```
coeff3
par(mfcol = c(4,5))
plot(density(ext3$beta[,1]), main = "Beta1")
abline(v = 0.01, col = "red")
plot(density(ext3$beta[,2]), main = "Beta2")
abline(v = 0.01, col = "red")
plot(density(ext3$beta[,3]), main = "Beta3")
abline(v = 0.01, col = "red")
plot(density(ext3$beta[,4]), main = "Beta4")
abline(v = 0.01, col = "red")
plot(density(ext3$beta[,5]), main = "Beta5")
abline(v = 0.01, col = "red")
plot(density(ext3$beta[,6]), main = "Beta6")
abline(v = 3, col = "red")
plot(density(ext3$beta[,7]), main = "Beta7")
abline(v = 3, col = "red")
plot(density(ext3$beta[,8]), main = "Beta8")
abline(v = 3, col = "red")
plot(density(ext3$beta[,9]), main = "Beta9")
abline(v = 3, col = "red")
plot(density(ext3$beta[,10]), main = "Beta10")
abline(v = 3, col = "red")
plot(density(ext3$beta[,11]), main = "Beta11")
abline(v = 3, col = "red")
plot(density(ext3$beta[,12]), main = "Beta12")
abline(v = 3, col = "red")
plot(density(ext3$beta[,13]), main = "Beta13")
abline(v = 3, col = "red")
plot(density(ext3$beta[,14]), main = "Beta14")
abline(v = 3, col = "red")
plot(density(ext3$beta[,15]), main = "Beta15")
abline(v = 3, col = "red")
plot(density(ext3$beta[,16]), main = "Beta16")
abline(v = 3, col = "red")
plot(density(ext3$beta[,17]), main = "Beta17")
abline(v = 3, col = "red")
plot(density(ext3$beta[,18]), main = "Beta18")
abline(v = 3, col = "red")
plot(density(ext3$beta[,19]), main = "Beta19")
abline(v = 3, col = "red")
plot(density(ext3$beta[,20]), main = "Beta20")
abline(v = 3, col = "red")
```

Listing 8: R code contd.

```
coeff4
par(mfcol = c(4,5))
plot(density(ext4$beta[,1]), main = "Beta1")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,2]), main = "Beta2")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,3]), main = "Beta3")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,4]), main = "Beta4")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,5]), main = "Beta5")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,6]), main = "Beta6")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,7]), main = "Beta7")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,8]), main = "Beta8")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,9]), main = "Beta9")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,10]), main = "Beta10")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,11]), main = "Beta11")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,12]), main = "Beta12")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,13]), main = "Beta13")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,14]), main = "Beta14")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,15]), main = "Beta15")
abline(v = 0.01, col = "red")
plot(density(ext4$beta[,16]), main = "Beta16")
abline(v = 3, col = "red")
plot(density(ext4$beta[,17]), main = "Beta17")
abline(v = 3, col = "red")
plot(density(ext4$beta[,18]), main = "Beta18")
abline(v = 3, col = "red")
plot(density(ext4$beta[,19]), main = "Beta19")
abline(v = 3, col = "red")
plot(density(ext4$beta[,20]), main = "Beta20")
abline(v = 3, col = "red")
```

3.2 Stan Code

Listing 9: Stan code Gamma Model

```
data{
  real x[270];
  real y[270];
}
parameters{
  real a0;
  real a1;
  real b0;
  real b1;
}
transformed parameters{
  real alpha[270];
  real beta[270];
  for (i in 1:270){
  alpha[i] = a0 + a1*x[i];
  beta[i] = b1 + b1*x[i];
}
}
model{
  for (i in 1:270){
    y[i] ~ gamma(alpha[i],beta[i]);
  }
}
generated quantities{
  real y_alex[270];
  for (i in 1:270)
  y_alex[i] = gamma_rng(alpha[i],beta[i]);
}
```

Listing 10: Stan code Logistic regression

```
data{
  int nr;
  int nc;
  matrix[nr,nc] X;
  int y[nr];
}
parameters{
  vector[nc] beta;
  vector<lower = 0>[nc] lambda;
  real <lower = 0> tau;
}
transformed parameters{
  matrix[nr,nc]theta;
  vector[nr] theta2;
  for (j in 1:nc){
  for(i in 1:nr){
    theta[i,j] = X[i,j]*beta[j];
}
}
  for(i in 1:nr){
    theta2[i] = sum(theta[i,]);
  }
}
model{
  for(i in 1:nr){
    y[i] ~ bernoulli_logit(theta2[i]);
  }
  for (i in 1:nc){
  beta[i] ~ normal(1,lambda[i]*tau);
  }
  tau ~ cauchy(0,1);
  lambda ~ cauchy(0,1);
}
```