Project: Introduction to Bayesian Inference (3579)
Project 4
2023-2024

1st year Master of Statistics
Hasselt University

Group members:
Ronald Makanga (2365466)
Juan Vanegas Jadan (2366405)
Gianni Guarraci (2366293)
Charles Muiruri (2365043)

Submission Date: 01 June, 2024

Lecturers:
Prof. Dr. Christel FAES

Contents

Modeling using Open Bugs

Question 1:

```
## Abstracting beta0 ... 8000 valid values
## Abstracting beta1 ... 8000 valid values
## Abstracting beta2 ... 8000 valid values
## Abstracting deviance ... 8000 valid values
## Abstracting beta0 ... 8000 valid values
## Abstracting beta1 ... 8000 valid values
## Abstracting beta2 ... 8000 valid values
## Abstracting deviance ... 8000 valid values
## Abstracting beta0 ... 8000 valid values
## Abstracting beta1 ... 8000 valid values
## Abstracting beta2 ... 8000 valid values
## Abstracting deviance ... 8000 valid values
```

The bugs model is as shown below

```
model
{
    for (i in 1:J) {
        Y[i] ~ dbin(p[i], N[i])
        logit(p[i]) <- beta0 + beta1 * btn133_400_pc[i] + beta2 *
            great_400pc[i]
    }
    beta0 ~ dnorm(0.00000E+00, 0.001)
    beta1 ~ dnorm(0.00000E+00, 0.001)
    beta2 ~ dnorm(0.00000E+00, 0.001)
}
```

```
##
## Iterations = 2001:10000
## Thinning interval = 1
## Number of chains = 3
## Sample size per chain = 8000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##              Mean      SD  Naive SE Time-series SE
## beta0      1.9208 0.05503 0.0003552      0.0004108
## beta1      0.3675 0.07584 0.0004895      0.0005638
## beta2      0.9763 0.08994 0.0005806      0.0006556
## deviance 254.0282 2.43933 0.0157458      0.0178846
##
## 2. Quantiles for each variable:
##
##              2.5%      25%      50%      75%    97.5%
## beta0      1.8140   1.8840   1.9200   1.9580   2.0300
## beta1      0.2189   0.3163   0.3671   0.4189   0.5157
## beta2      0.7994   0.9154   0.9766   1.0370   1.1530
## deviance 251.3000 252.2000 253.4000 255.1000 260.5000
```

```
## The mean probability of vaccination coverage for each poverty group is as follows:
```

```
## <133% FPL:  0.8722314
## 133% to <400% FPL:  0.9079025
## >400% FPL:  0.9477026
```

Question 2

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 124
##
## Initializing model
```

MCMC method and check converge of the MCMC chains Convergence tests

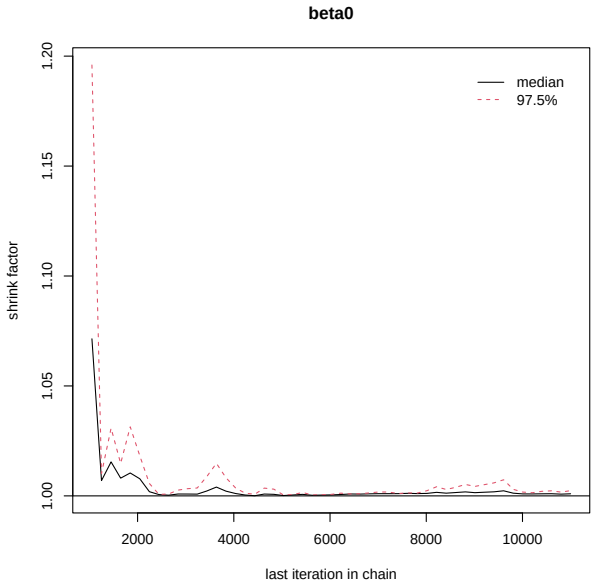
In our analysis, we used Gibbs Sampling to estimate the posterior distributions for our three unknown parameters (beta0, beta1 and beta2). To test for convergence for our MCMC chains, we used both Gelman-Rubin convergence diagnostic and trace plots.

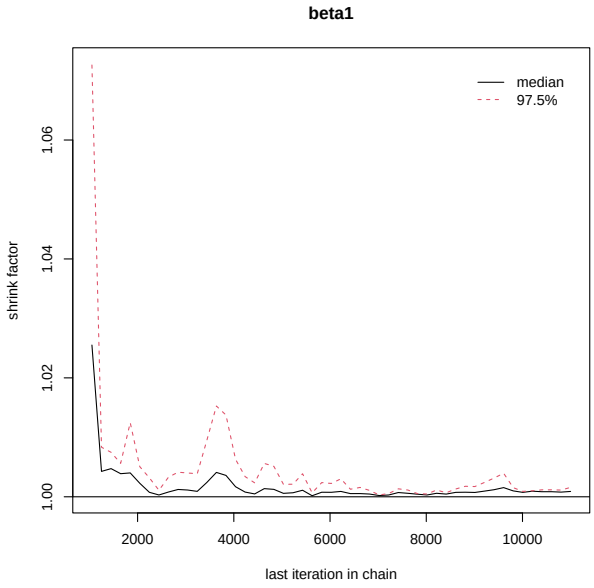
The Gelman-Rubin convergence diagnostic method allows us to compare within and between chain variances for each variable. Best results are obtained for parameters whose marginal posterior densities are approximately normal. To run the Gelman-Rubin convergence test, we set three MCMC chains and for each chain we set distinct starting values for our unknown parameters as shown in the code. We ran 10,000 MCMC trials with a burn-in value of 2,000 and a thining value of 1 for each chain.

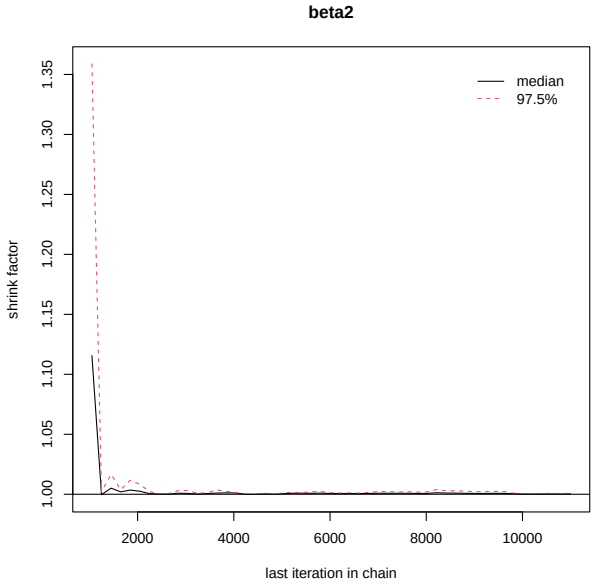
Then we produced gelman-rubin statistics and plot using the gelman.diag and gelman.plot functions in R. The gelman.diag function gives us the scale reduction factors for each parameter (beta0,beta1,and beta2). A factor of 1 means that the between variance and within chain variance are equal, larger values mean that there is still a notable difference between chains. on the other hand, gelman.plot shows if the shrink factor has really converged, or whether it still fluctuating. Both are results from both our functions.

Gelman-Rubin Convergence Diagnostic

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
## beta0          1          1
## beta1          1          1
## beta2          1          1
##
## Multivariate psrf
##
## 1
```

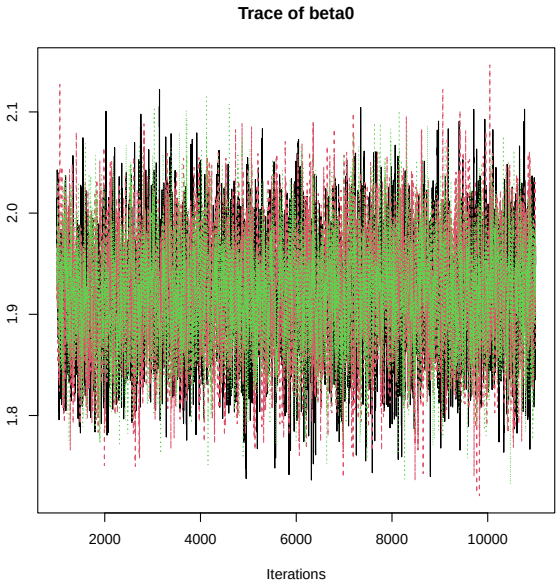


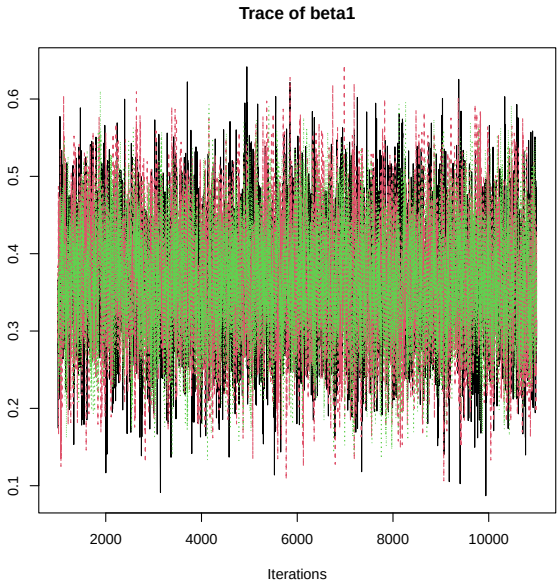


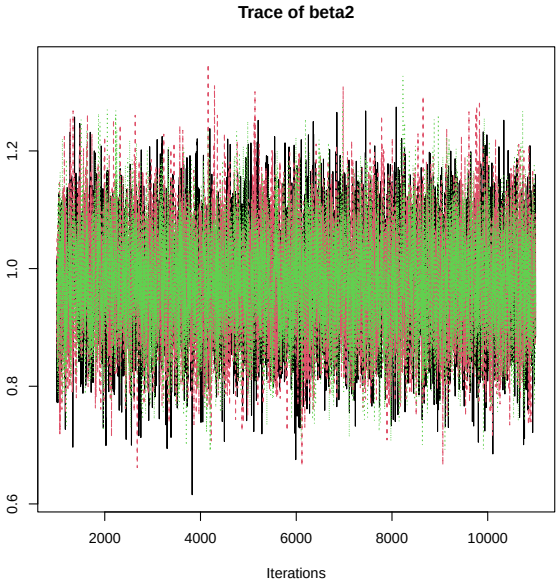


Trace Plots

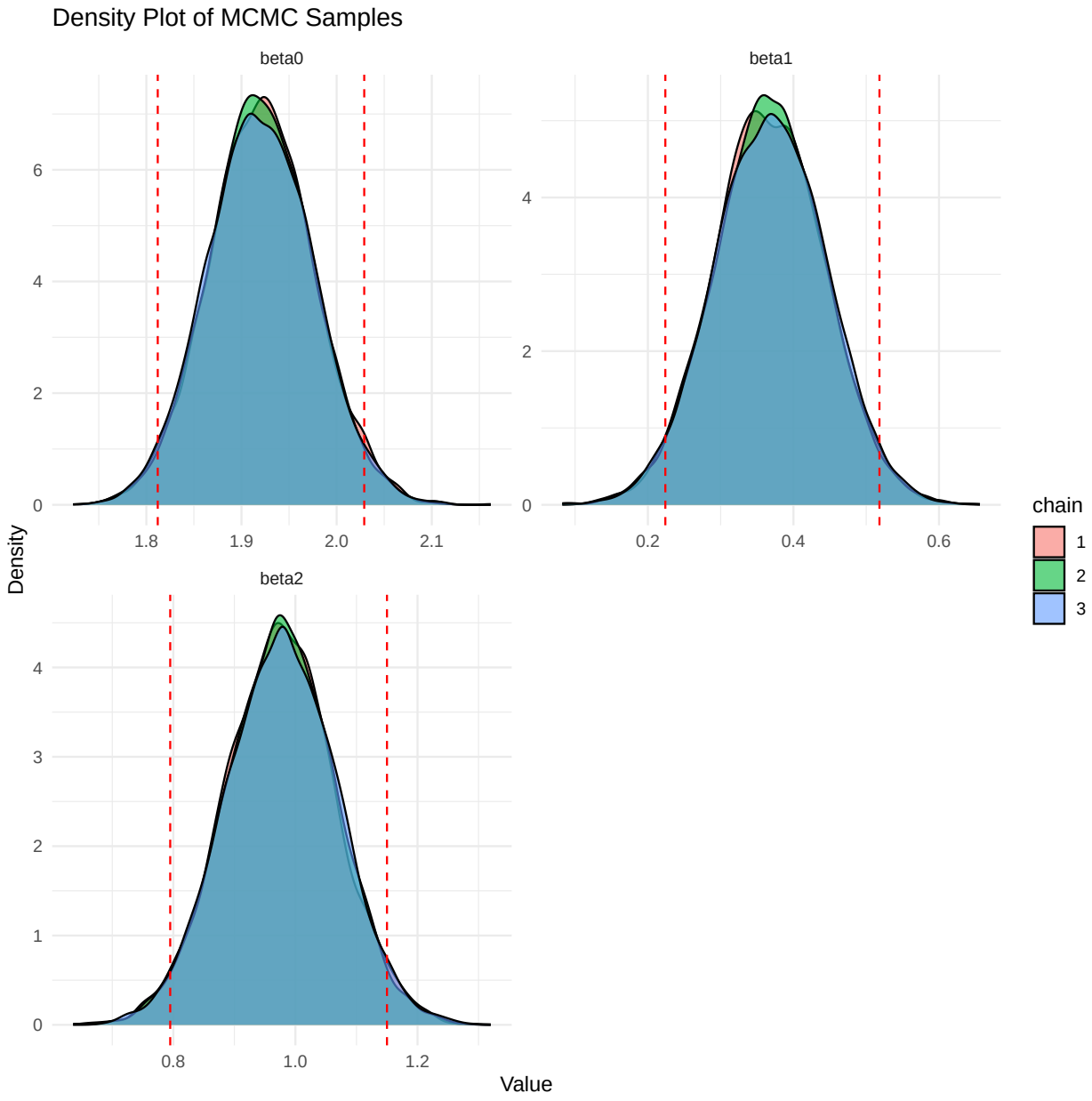
In addition, we also explored convergence of MCMC chains by looking at our trace plots for each of our unknown parameter. The trace plot shows the parameter value at the time t against the iteration number.


Trace of beta0
2.1
2.0
1.9
1.8
2000
4000
6000
8000
10000
Iterations


Trace of beta1
0.6
0.5
0.4
0.3
0.2
0.1
2000
4000
6000
8000
10000
Iterations


Trace of beta2
1.2
1.0
0.8
0.6
2000
4000
6000
8000
10000
Iterations

Question 3



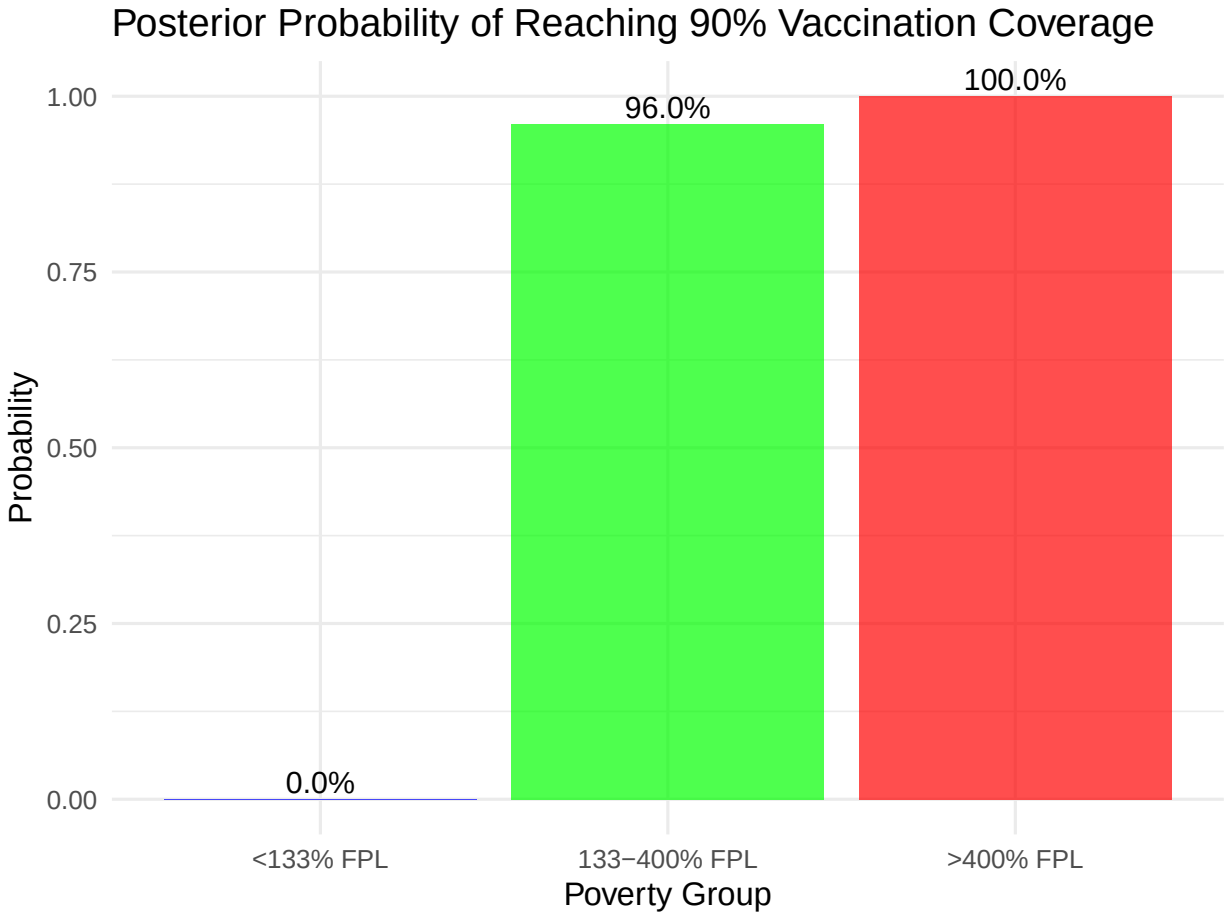
Summary of posterior distribution

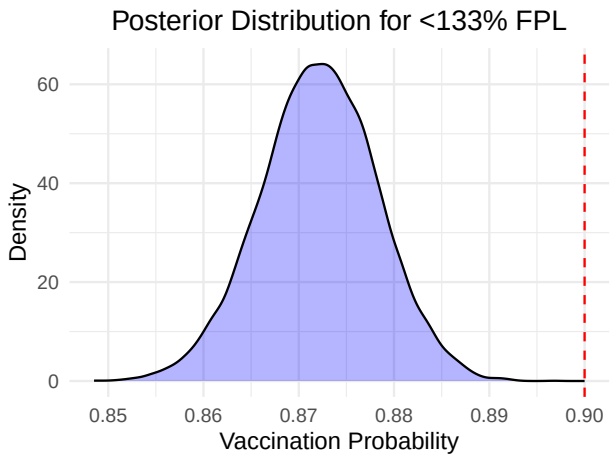
- The Posterior density with relation to β_0 has a mean of **0.8722314** which is the probability of children vaccinated from households whose income is less than 133%, with a 95% credible HPD interval of **(0.8596034, 0.8838084)** in which the proportion lies with 95% probability.
- The Posterior density with relation to β_1 has a mean of **0.9079025** which is the probability of children vaccinated from households whose income between 133% and 400% , with a 95% credible HPD interval of **(0.8845253, 0.9273852)** in which the proportion lies with 95% probability.
- The Posterior density with relation to β_2 has a mean of **0.9477026** which is the probability of children vaccinated from households whose income is more than 400%, with a 95% credible HPD interval of**(0.9313043, 0.9600363)** in which the proportion lies with 95% probability.

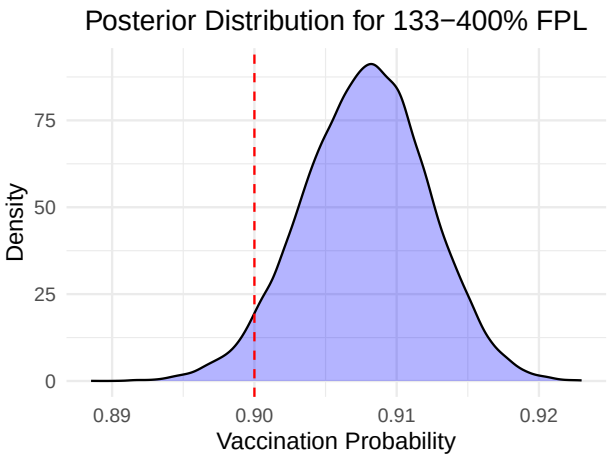
Question 4

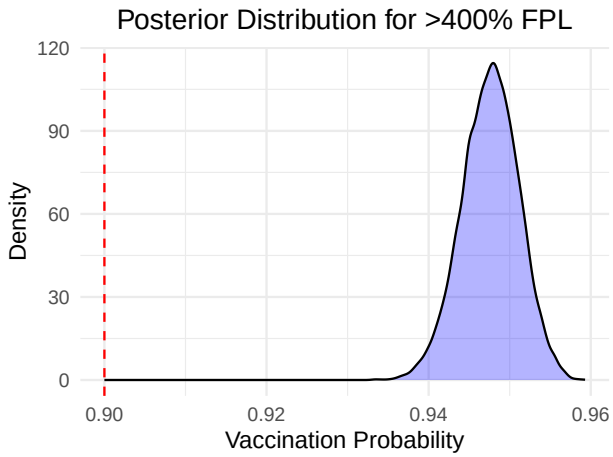
```
## Posterior probability that vaccination coverage target (90%) is reached:
## <133% FPL:  0
## 133-400% FPL:  0.9601667
## >400% FPL:  1
```

1:3, c(0, 0.960166666666667, 1)





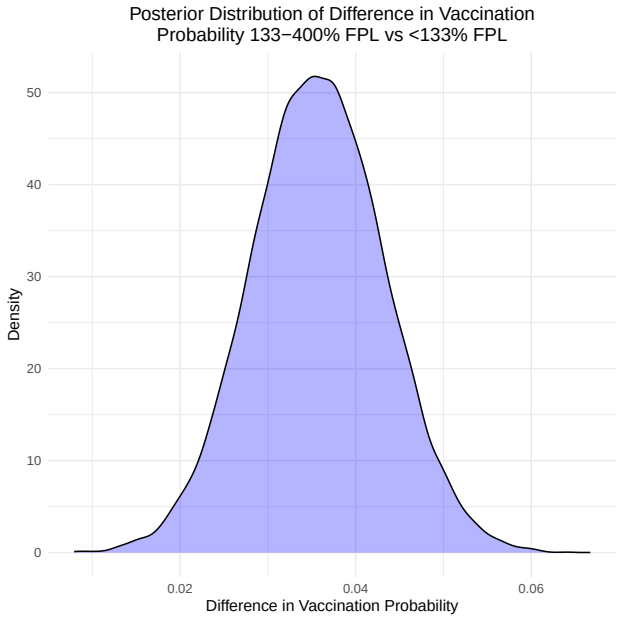


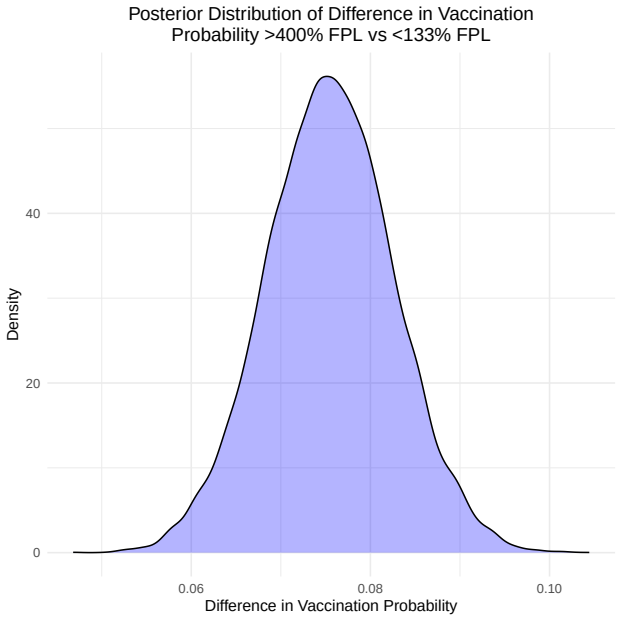


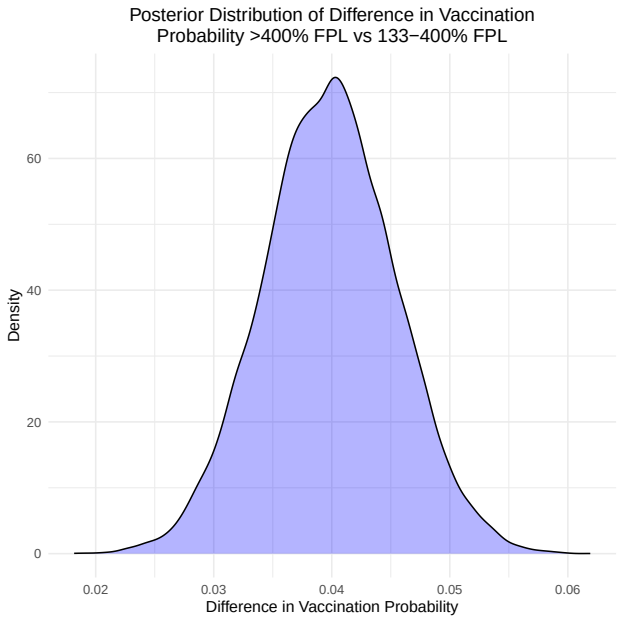
Posterior Probabilities of Reaching 90% Vaccination Coverage:

- <133% FPL: Only 'r round(prob_133less_reached * 100, 2)'% of the time does the vaccination coverage meet or exceed 90%, indicating lower vaccination rates in this poverty group.
- 133-400% FPL: With a r' round(prob_133_400_reached * 100, 2)'% probability, this group has a better chance of meeting the 90% target, showing improvement compared to the <133% FPL group.
- 400% FPL: This group has a r' round(prob_400more_reached * 100, 2)'% probability of reaching or exceeding 90% coverage, indicating high vaccination rates.

Question 5







Impact of Poverty on Vaccination Coverage:

- 133-400% FPL vs <133% FPL: The mean difference in vaccination coverage is 'r round(diff_133_400_vs_133less_summ 4)' with a 95% credible interval of ['r round(diff_133_400_vs_133less_summary["2.5%"], 4)', 'r round(diff_133_400_vs_133less_summary["97.5%"], 4)'] suggests that vaccination coverage is higher by about 'r round(diff_133_400_vs_133less * 100, 2)'% in the 133-400% FPL group compared to the <133% FPL group.
- 400% FPL vs <133% FPL: A mean difference of in vaccination coverage is 'r round(diff_400more_vs_133less_summary["Mean"], 4)' with a 95% credible interval of ['r round(diff_400more_vs_133less_summary["2.5%"], 4)', 'r round(diff_400more_vs_133less_summary["97.5%"], 4)'] indicates that the >400% FPL group has a higher vaccination coverage by about 'r round(diff_400more_vs_133less * 100, 2)'% compared to the <133% FPL group.

- 400% FPL vs 133-400% FPL: The mean difference of in vaccination coverage is 'r round(diff_400more_vs_133_400_summary["Mean"], 4)' with a 95% credible interval of ['r round(diff_400more_vs_133_400_summary["2.5%"], 4)', 'r round(diff_400more_vs_133_400_summary["97.5% 4)'] shows that the highest income group (>400% FPL) has about 'r round(diff_400more_vs_133_400 * 100, 2)'% higher coverage compared to the middle-income group (133-400% FPL).

Insights

Income Gradient in Vaccination Coverage:

- There is a clear gradient in vaccination coverage by poverty level, with higher-income groups achieving better vaccination rates. This suggests that income is a significant factor in vaccination coverage.

Target Achievement:

- The highest income group (>400% FPL) consistently meets or exceeds the 90% vaccination coverage target, indicating successful vaccination efforts in this group. Lower-income groups, especially the <133% FPL group, struggle to meet the target. highlighting the need for targeted interventions.

Question 6

Investigate whether the vaccination coverages are distinct at the different locations by adding a location-specific intercept.

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 11
##    Total graph size: 208
##
## Initializing model
```

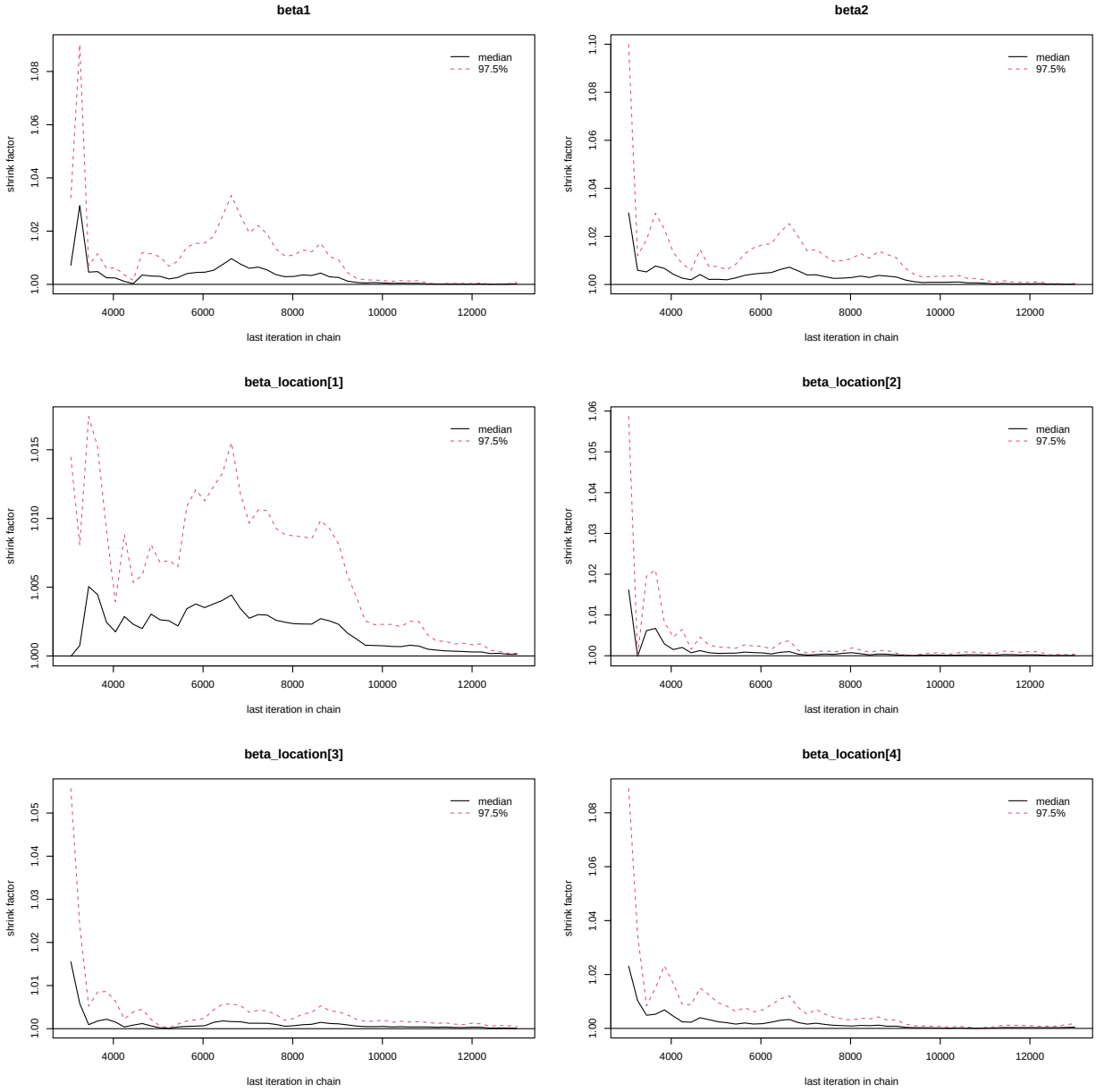
The bugs model is as shown below

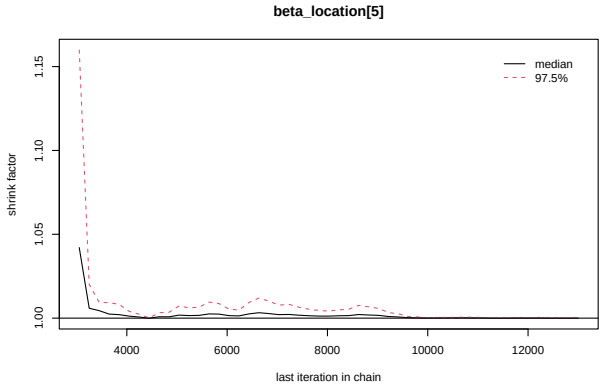
```
model
{
    for (i in 1:J) {
        Y[i] ~ dbin(p[i], N[i])
        logit(p[i]) <- beta1 * btn133_400_pc[i] + beta2 * great_400pc[i] +
            beta_location[location[i]]
    }
    beta1 ~ dnorm(0.00000E+00, 0.001)
    beta2 ~ dnorm(0.00000E+00, 0.001)
    for (j in 1:J_locations) {
        beta_location[j] ~ dnorm(0.00000E+00, 0.001)
    }
}
```

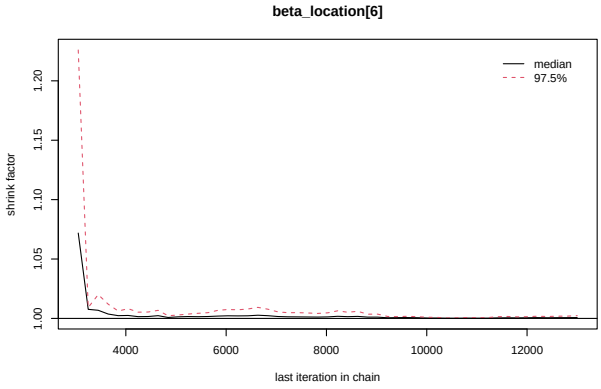
Gelman-Rubin Convergence Diagnostic

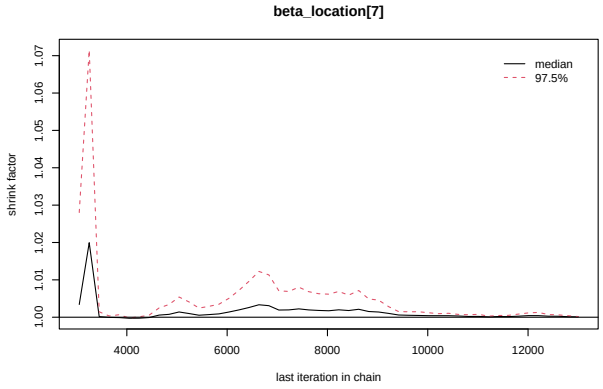
```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## beta1                     1          1
```

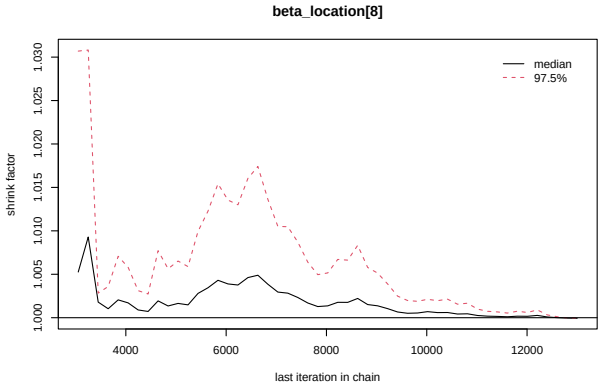
```
## beta2                    1          1
## beta_location[1]         1          1
## beta_location[2]         1          1
## beta_location[3]         1          1
## beta_location[4]         1          1
## beta_location[5]         1          1
## beta_location[6]         1          1
## beta_location[7]         1          1
## beta_location[8]         1          1
## beta_location[9]         1          1
##
## Multivariate psrf
##
## 1
```

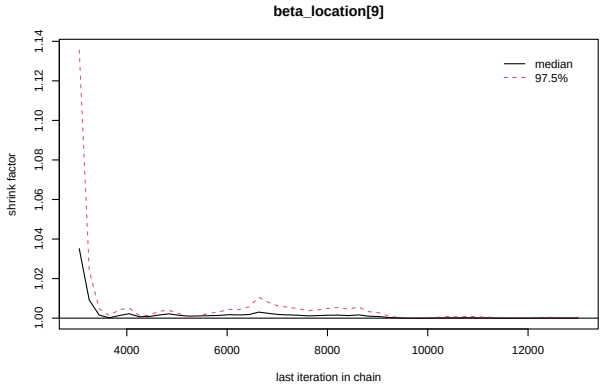



beta_location[5]
median
97.5%
shrink factor
1.00
1.05
1.10
1.15
4000
6000
8000
10000
12000
last iteration in chain

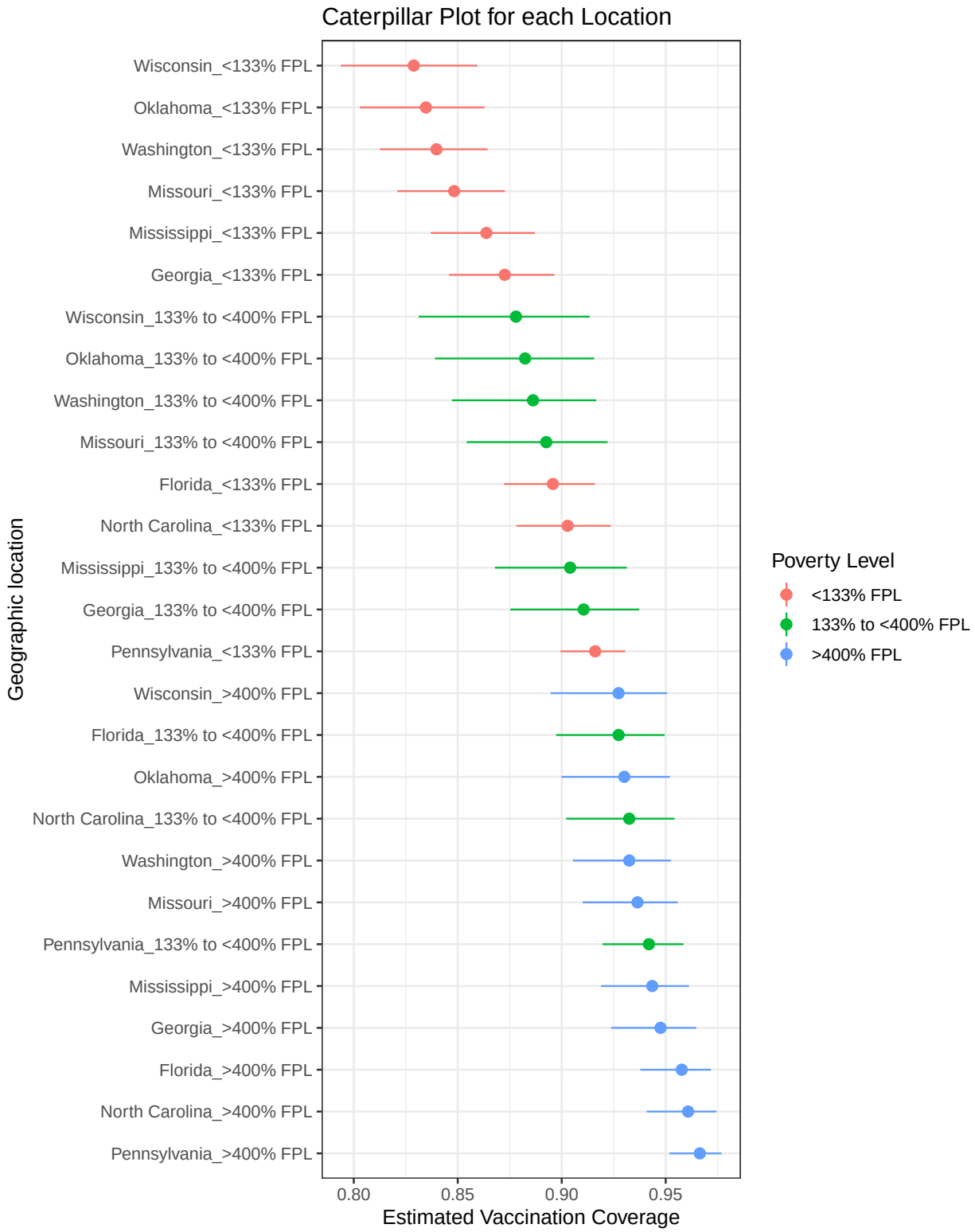

beta_location[6]
median
97.5%
shrink factor
1.00
1.05
1.10
1.15
1.20
4000
6000
8000
10000
12000
last iteration in chain


beta_location[7]
median
97.5%
shrink factor
1.00
1.01
1.02
1.03
1.04
1.05
1.06
1.07
4000
6000
8000
10000
12000
last iteration in chain


beta_location[8]
median
97.5%
shrink factor
1.000
1.005
1.010
1.015
1.020
1.025
1.030
4000
6000
8000
10000
12000
last iteration in chain


beta_location[9]
median
97.5%
shrink factor
1.00
1.02
1.04
1.06
1.08
1.10
1.12
1.14
4000
6000
8000
10000
12000
last iteration in chain

Question 7



This caterpillar plot displays the expected coverage for each geographic region within our data. The point represents the meancoverage while the error bars represent the 95% credible interval. The plot shows that the expected coverage for each region is distinct.

Question 8

Poverty	Estimate
133-400% FPL	90
>400% FPL	94
<133% FPL	86
Total	270

This table shows the estimated number of vaccinated children in Mississippi by poverty group. The estimated number of vaccinated children from households with income less than 133% FPL is 86, from households with income between 133% and 400% FPL is 90,and from households with income greater than 400% FPL is 94.

Appendix

R Code

```
if (!require(pacman)) install.packages("pacman")
p_load(rjags, coda, nimble, R2OpenBUGS, ggplot2, here, dplyr, ggpubr, tidyr)

projdata <- as.data.frame(read.csv(here("data/projectdata.txt")))|>
  mutate(less_133pc = ifelse(Poverty == "<133% FPL", 1,0),
         btn133_400_pc = ifelse(Poverty == "133% to <400% FPL", 1,0),
         great_400pc = ifelse(Poverty == ">400% FPL", 1,0))
# Data prep for bugs
model_data <- list(
  Y = projdata$Vaccinated,
  N = projdata$Sample.Size,
  btn133_400_pc = projdata$btn133_400_pc,
  great_400pc = projdata$great_400pc,
  J = nrow(projdata)
)

model_inits <- list(
  list(beta0 = 0, beta1 = 0, beta2 = 0)
)

parameters <- c("beta0", "beta1", "beta2")

model1 <- function(){
  for (i in 1:J){
    Y[i] ~ dbin(p[i], N[i])
    logit(p[i]) <- beta0 + beta1*btn133_400_pc[i] + beta2*great_400pc[i]
  }
  #priors
  beta0 ~ dnorm(0, 0.001)
  beta1 ~ dnorm(0, 0.001)
  beta2 ~ dnorm(0, 0.001)
}

# Write model to file
write.model(model1, here("models/model1code.txt"))
# View file
file.show(here("models/model1code.txt"))

model.out <- bugs(model_data, model_inits,
                  parameters = parameters, model.file = here("models/model1code.txt"),
                  n.chains = 1, n.iter = 10000, n.burnin = 5000, codaPkg = TRUE,
                  debug = FALSE)

# debug=TRUE opens openBug and displays traceplots and summaries

# Model output
out <- read.bugs(model.out)
summary(out)

# Prep data for density and trace plots
```

```
mcmc_samples <- as.mcmc(out)
mcmc_df <- as.data.frame(mcmc_samples)
mcmc_df$iteration <- 1:nrow(mcmc_df)
mcmc_long <- pivot_longer(mcmc_df, cols = -iteration, names_to = "Parameter",
                          values_to = "Value")

points_data <- data.frame(x = mcmc_df$beta0, x1 = mcmc_df$beta1,x2 = mcmc_df$beta2,
                          x3 = mcmc_df$deviance,y = rep(0, nrow(mcmc_df)))

# Density plots
ggarrange(p_beta0 <- ggplot(mcmc_df, aes(x = beta0)) +
            geom_density(fill = "blue", alpha = 0.1) +
            geom_point(data = points_data, aes(x = x, y = y)) +
            labs(title = "Posterior Distribution of beta0", x = "beta0", y = "Density")+
            theme_bw(base_size = 12)+
            theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()),

          p_beta1 <- ggplot(mcmc_df, aes(x = beta1)) +
            geom_density(fill = "green", alpha = 0.1) +
            geom_point(data = points_data, aes(x = x1, y = y)) +
            labs(title = "Posterior Distribution of beta1", x = "beta1", y = "Density") +
            theme_bw(base_size = 12)+
            theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()),

          p_beta2 <- ggplot(mcmc_df, aes(x = beta2)) +
            geom_density(fill = "red", alpha = 0.1) +
            geom_point(data = points_data, aes(x = x2, y = y)) +
            labs(title = "Posterior Distribution of beta2", x = "beta2", y = "Density")+
            theme_bw(base_size = 12)+
            theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()),

          p_dev <- ggplot(mcmc_df, aes(x = deviance)) +
            geom_density(fill = "yellow", alpha = 0.1) +
            geom_point(data = points_data, aes(x = x3, y = y)) +
            labs(title = "Posterior Distribution of deviance", x = "Deviance", y = "Density")+
            theme_bw(base_size = 12)+
            theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()),
          nrow = 2, ncol = 2)

# Trace plots
ggplot(mcmc_long, aes(x = iteration, y = Value, color = Parameter)) +
  geom_line() +
  scale_color_manual(values = c("blue", "green", "red", "yellow")) +
  facet_wrap(~ Parameter, scales = "free_y") +
  labs(title = "Trace Plots of MCMC Samples", x = "Iteration", y = "Parameter Value") +
  theme_bw(base_size = 12)+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        legend.position = "none")

# Autocorrelation and crosscorrelation plots
crosscorr.plot(out)
autocorr.plot(out)
```