Bayesian Final Project

Tianyi Li, Jade Gu, Jiashu Liu, Jeremy Lu

5/12/2023

```
# Load packages
library(bayesrules)
library(tidyverse)
library(janitor)
library(ggplot2)
library(rstan)
library(bayesplot)
library(broom.mixed)
library(rstanarm)
library(tidybayes)
library(dplyr)
library(purrr)
```

2.1 Proposal

In this project we will be developing Bayesian models in order to identify the effects of mother's characteristics including her education level, race, age, gestational length, smoking status, and BMI, on female new-born's birth weight.

2.2 Attribution

Tianyi Li: Project leader, team management, raw data summary, build statistical model

Jade Gu: Raw data summary, model fitting and selection, assisting in prior predictive simulation

Jiashu Liu: Prior predictive simulation, assisting in model fitting and selection

Jeremy Lu: Presentation preparation, assiting in model selection, summarize key convergence diagnostics

2.3 Raw Data Summary

2.3.1 Data Origin

Below is a link to a compressed CSV file containing US birth data for the year 2018. The data is provided by the National Center for Health Statistics and can be accessed from the National Bureau of Economic Research website.

2.3.2 Data Clean (Include Variable Selection)

This 2018 birth data contains 3801534 obs. of 240 variables. First, based on our research topic, we will keep girls' data only. Second, we will pick 6 most interesting variables including: dbwt(infant's birthweight in grams), mager(mother's years of age), dmar(martial status), cig_rec(cigarette recode), bmi(body mass index), combgest(combined gestation weeks). Then, we will exclude any missing values. After that, we will factor all categorical variables.

2.3.3 Data Sampling

Since this birth data after filter contains 1591008 obs. which is still extremely large, we will get a random sample of births that includes only 0.1% of all obs, which results in 1591 obs.

```
# Load sampled data
birth <- read.csv("D:\\APSTA 2123\\birth.csv")

# Re-factor all categorical variables and covert them into 1 and 0
birth <- birth %>%
  mutate(dmar = factor(dmar, levels = c("2", "1"), labels = c("0", "1")),
         cig_rec = factor(cig_rec, levels = c("N", "Y"), labels = c("0", "1")))
```

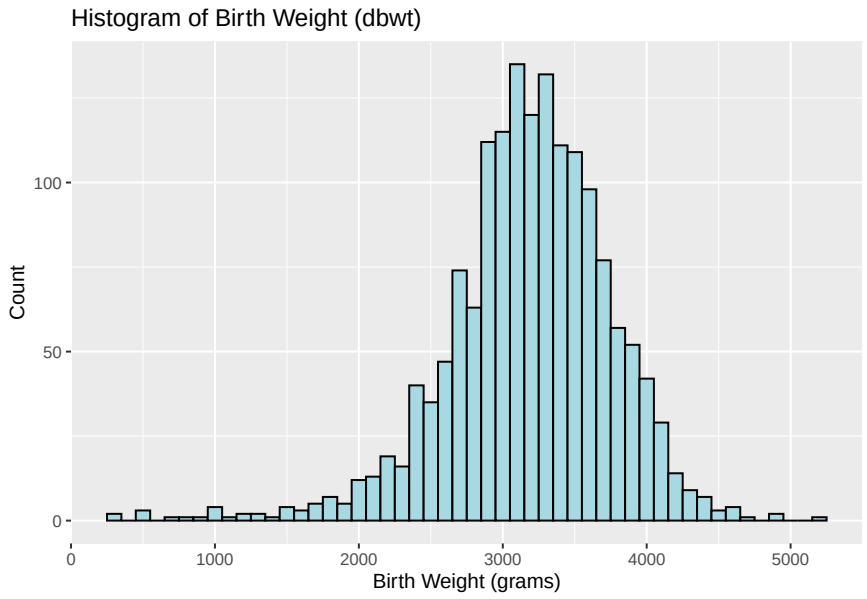
2.3.4 Exploratory data analysis

In the birth data, we have four numerical variables and two factored categorical variables.

Numerical variables:

dbwt (Birth weight)

```
# Create a histogram of the dbwt variable
ggplot(data = birth, aes(x = dbwt)) +
  geom_histogram(binwidth = 100, color = "black", fill = "#a8d8e2") +
  labs(title = "Histogram of Birth Weight (dbwt)", x = "Birth Weight (grams)", y = "Count")
```



Above is a histogram plot of birth weight of female infant babies. Values ranged from 320g (Yes it's possible) to 5220g. We computed some descriptive statistics: Median = 3220; Mean = 3183; SD = 574.74.

mager (Mom's age) Mother's integer years of age. Values ranged from 14 to 47 yrs old. We computed some descriptive statistics: Median = 29; Mean = 28.59; SD = 5.86.

bmi Mother's body mass index. Values ranged from 13.60 to 69.10. We computed some descriptive statistics: Median = 25.70; Mean = 27.29; SD = 7.00. **combgest** Mother's combined gestation in weeks. Values ranged from 20.00 to 47.00. We computed some descriptive statistics: Median = 39.00; Mean = 38.55; SD = 2.49.

Categorical variables:

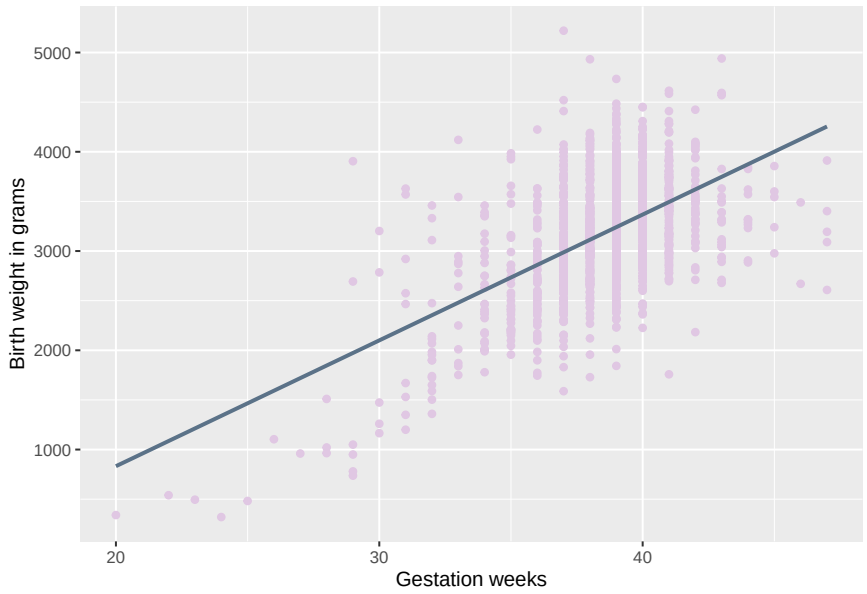
dmar Mother's martial status. "1" stands for married, and "0" stands for unmarried. According to the birth data, 956 out of 1591 observations (60.09%) are married, and the remaining 635 observations (39.91%) are unmarried. The mean birth weight of married moms is 3260g and median is 3280g, while the mean and median birth weight of unmarried is 3068g and 3105g.

cig_rec Whether the mother smokes or not. "1" stands for smoke, and "0" stands for non-smoke. According to the birth data, 106 out of 1591 observations (6.66%) are smokers, and the remaining 1481 observations (93.34) are non-smokers. The mean birth weight of moms who smoke is 2999g and median is 3023g, while the mean and median birth weight of mom who don't smoke is 3197g and 3232g.

2.3.5 More visualization

```
ggplot(birth, aes(x = combgest, y = dbwt)) +
  geom_point(color = "#e0c7e3") +
  geom_smooth(method = "lm", se = FALSE, color = "#5b7288") +
  labs(x = "Gestation weeks", y = "Birth weight in grams")
```

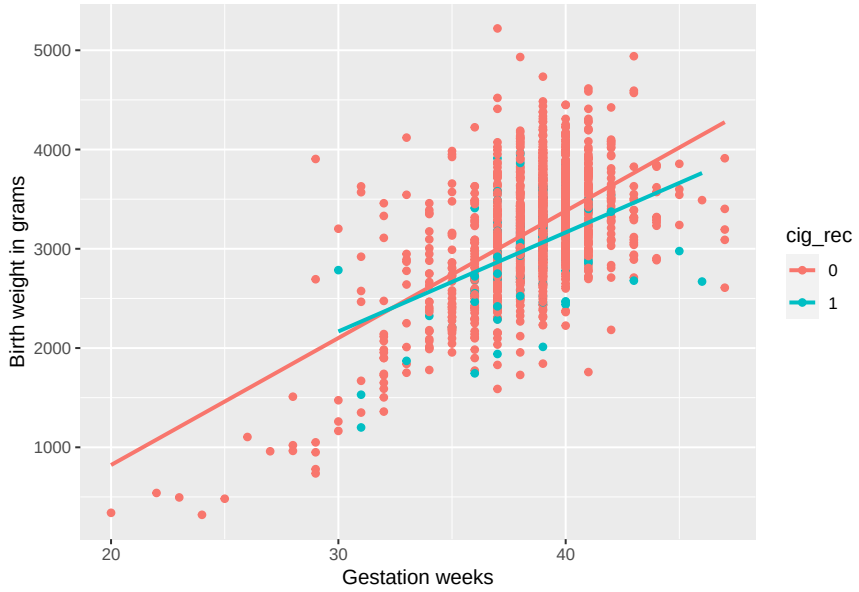
```
## `geom_smooth()` using formula = 'y ~ x'
```



We can observe a positive linear relationship between the mother's combined gestation length and the female newborn's birth weight. This is expected, mothers with longer combined gestation are expected to get heavier baby girls. However, there might be some variations and potential outliers. Let's add whether or not the mom smokes into account:

```
ggplot(birth, aes(x = combgest, y = dbwt, color = cig_rec)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  xlab("Gestation weeks") +
  ylab("Birth weight in grams")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



From the scatterplot of birth weight versus gestation weeks of moms who smoke and does not smoke, superimposed with the observed linear relationships in the solid lines, the slope of the lines are different, indicating that there is an interaction effect between the predictor variables and the relationship between gestation weeks and birth weight is different between moms who smoke and those who don't smoke.

2.4 Statistical Model

We aim to model the relationship between the birth weight of female newborns and maternal characteristics, including age, marital status, smoking status, body mass index, and gestation weeks.

From the histogram of birth weight before, we can assume that the birth weight of each female newborn follows a normal distribution. Also, from the scatterplot above, there is a positive linear relationship between the gestation length and birth weight. Let Y_i denotes the birth weight for observation i. Since Y_i is numerical and we assume there is a linear relationship between Y and the predictor variables, a Normal Bayesian regression model is the most appropriate model to use. With 5 predictor variables and 1 outcome variable, our model looks like:

$$\begin{array}{lrll} \text{data:} & Y_i|\beta_0,\beta_1,...,\beta_5,\sigma & \overset{ind}{\sim} N\left(\mu_i,\sigma^2\right) & \text{with} \quad \mu_i=\beta_0+\beta_1X_{i1}+...+\beta_5X_{i5} \\ \text{priors:} & \beta_0 & \sim N\left(m_0,s_0^2\right) & \\ & \beta_1 & \sim N\left(m_1,s_1^2\right) & \\ & ... & & \\ & \beta_5 & \sim N\left(m_5,s_5^2\right) & \\ & \sigma & \sim \text{Exp}(l). & \end{array} \tag{1}$$

For observation of female new-born i, Y_i is her birth weight, X_{i1} is her mother's gestation length in week, X_{i2} is a binary outcome indicating whether her mom smokes, X_{i3} is her mom's bmi index, X_{i4} is her mom's age, X_{i5} is a binary variable of whether her mom is married.

We have to specify the prior mean and sd of each variable. For simplicity, we assume that the prior distributions are independent. First, we center all numerical predictor variables by subtracting the value of each observation by its mean. Now the intercept β_0 represents the birth weight of an average female new-born whose mother is unmarried and does not smoke. According to Medical News Today, the average birth weight of a full-term female is 3.2 kg, which normally ranges from 2.5 kg to 4.0 kg; however, it can also be below 1.5 kg or above 4.5 kg. As a result, we take 3200 and 750 as prior mean and standard deviation for β_0.

For β_1 of X_1, the gestation week, from prior scientific understanding, on average, one week longer in gestation length is associated with an increase in birth weight by 10 to 225 grams, so we take 108 and 54 as mean and sd for β_1.

For β_2 of X_2, whether mom smokes or not, from prior scientific research, on average, the birth weight of female baby whose mother smokes could be 200 to 300 grams lower than baby whose mother doesn't smoke. We take -250 and 25 for prior mean and sd of β_2.

For β_3 of X_3, the bmi index of baby's mother, from prior scientific research findings, on average, one unit increase in the bmi index is associated with 20 to 30 grams increase in birth weight. We take 25 and 2.5 as prior mean and sd for β_3.

For β_4 of X_4, the mom's age. we employ weakly informative criteria and set 0 and 2.5 as prior mean and sd.

For β_5 of X_5, the marriage status of baby's mom, prior scientific study suggests that the average birth weight of female infants born to married mothers may be around 50 grams to 100 grams higher. We take 75 and 12.5 as prior mean and sd for β_5

Finally, for the error term, a plausible exponential rate parameter is 0.003. We have $\sigma \sim Exp(0.003)$.

2.5 Prior Predictive Simulation

Now let's start simulate some data based on our prior parameters. First we define our prior model function in R.

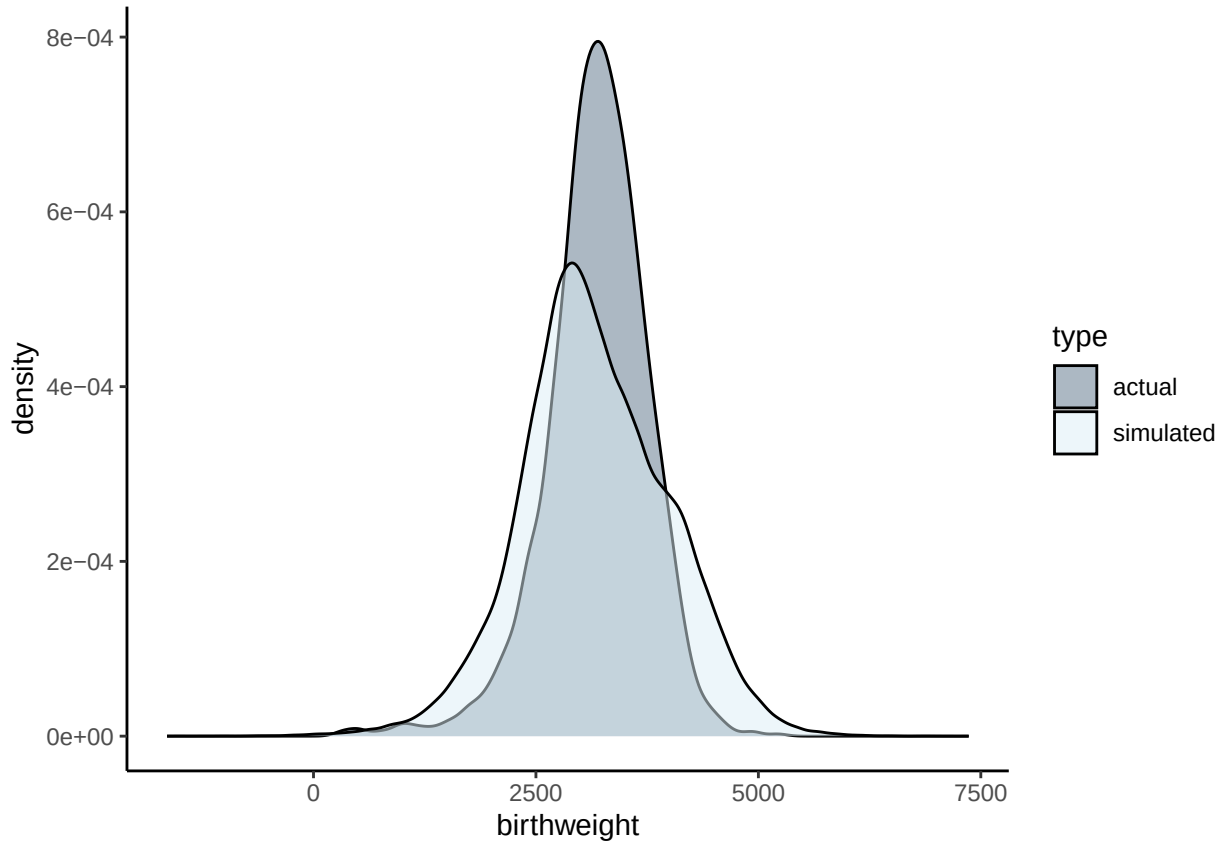
```
set.seed(84735)

# Define function
prior_pred <- function(data) {
  beta_0 <- rnorm(1, 3200, 750)
  beta_1 <- rnorm(1, 108, 54)
  beta_2 <- rnorm(1, -250, 25)
  beta_3 <- rnorm(1, 25, 2.5)
  beta_4 <- rnorm(1, 0, 2.5)
  beta_5 <- rnorm(1, 75, 12.5)
  sigma <- rexp(1, 0.003)
  l <- nrow(data)
  y <- numeric(l)
  for (i in 1:l) {
    # Convert factors to numeric
    cig_rec_num <- as.numeric(data$cig_rec[i])
    dmar_num <- as.numeric(data$dmar[i])

    mu <- beta_0 + beta_1 * data$centered_combgest[i] + beta_2 * cig_rec_num +
          beta_3 * data$centered_bmi[i] + beta_4 * data$centered_mager[i] + beta_5 * dmar_num
```

```
    y[i] <- rnorm(1, mu, sigma)
  }
  return(y)
}

# Simulate 100 times
n <- 100
pr_p <- replicate(n = n, prior_pred(birth))
dim(pr_p)
```

```
## [1] 1591  100
```

To get a sense of how our prior parameters perform, we draw a density plot of the original data and our simulated data.

```
# Extract actual data
actual_data <- birth$dbwt

# Generate prior predictive simulation data
simulated_data <- pr_p

# Combine data into a data frame
df <- data.frame(birthweight = c(actual_data, simulated_data),
                 type = rep(c("actual", "simulated"), c(length(actual_data),
                                                        length(simulated_data))))

# Plot the distribution of the actual data and the prior predictive simulation
ggplot(df, aes(x = birthweight, fill = type)) +
  geom_density(alpha = 0.5) +
  scale_fill_manual(values = c("#5b7288", "#ddeff6")) +
  theme_classic()
```



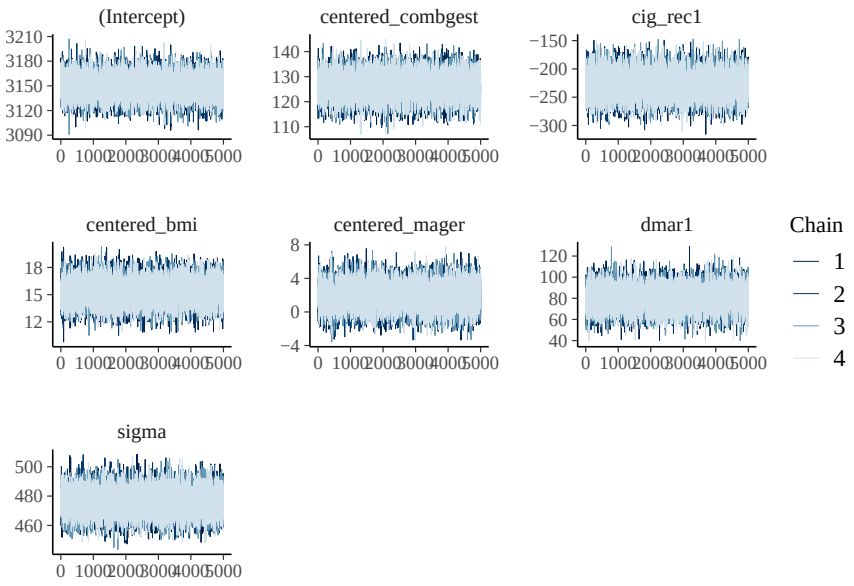
The model looks good. However, the simulated data indicates a higher variability compared to the actual data. So we decide to lower the sd of our prior parameters.

2.6 Modeling Fitting, PPCs, and Model Selection

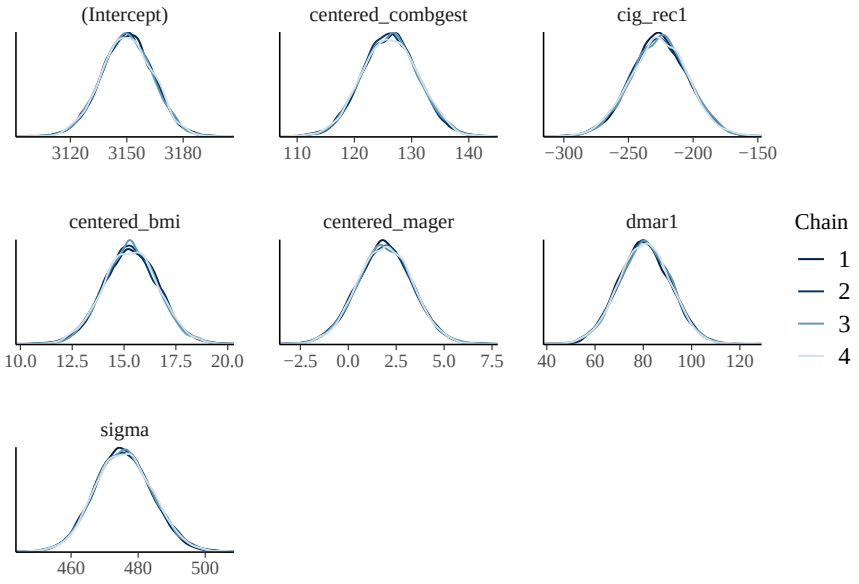
```
# Set prior
prior_intercept <- normal(3200, 750, autoscale = FALSE)
prior_beta <- normal(c(108, -250, 25, 0, 75),
                     c(50, 25, 2, 2, 12),
                     autoscale = FALSE)
prior_sigma <- exponential(0.003)

# Fit the model
birth_model <- stan_glm(dbwt ~ centered_combgest + cig_rec + centered_bmi + centered_mager + dmar,
                        data = birth,
                        prior_intercept = prior_intercept,
                        prior = prior_beta,
                        prior_aux = prior_sigma,
                        chains = 4,
                        iter = 5000*2,
                        seed = 84735)


# Summarize the posterior distribution
summary(birth_model)
```

Then, we will create and interpret both visual and numerical diagnostics of our MCMC simulation.

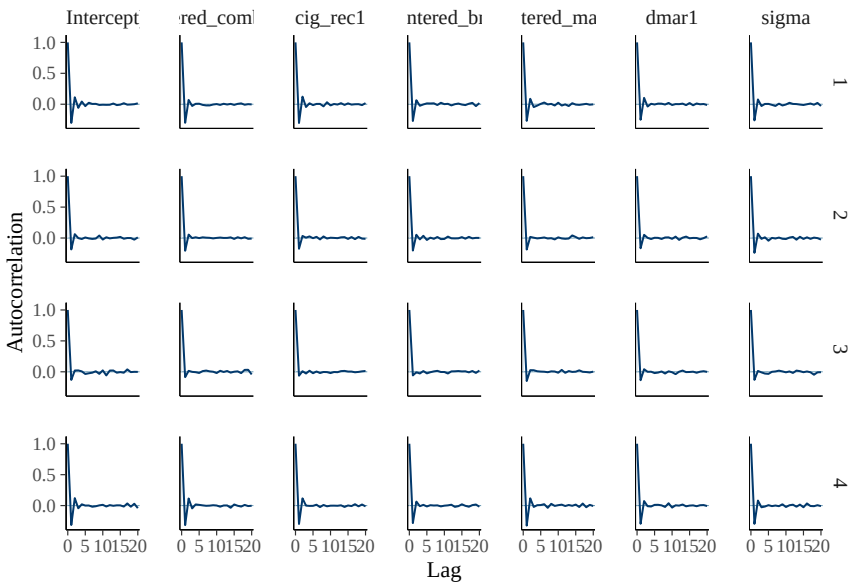
```
# Visual diagnostics of MCMC simulation
mcmc_trace(birth_model, size = 0.1)
```




```
mcmc_dens_overlay(birth_model)
```




```
mcmc_acf(birth_model)
```




```
# Numerical diagnostics of MCMC simulation
neff_ratio(birth_model)
```

```
##       (Intercept) centered_combgest          cig_rec1      centered_bmi
##           1.53915           1.52705           1.44035           1.57925
##    centered_mager             dmar1             sigma
##           1.54545           1.44910           1.61220
```

```
rhat(birth_model)
```

```
##       (Intercept) centered_combgest          cig_rec1      centered_bmi
##         0.9999017         0.9998276         1.0000752         0.9998445
##    centered_mager             dmar1             sigma
##         0.9998644         0.9998943         0.9999645
```

Regarding the assessment of our MCMC simulation, the visual diagnostics provide valuable insight into the behavior of the Markov chains. In particular, we have examined the trace plots, density plots, and autocorrelation plots for each parameter. Based on the trace plots, we can observe that the lines for each parameter fluctuate randomly without exhibiting any clear patterns. Additionally, all four chains appear to mix well, which is a positive indication that the simulation is properly exploring the parameter space. Furthermore, the density plots show that each parameter has a smooth, bell-shaped distribution, which is a desirable property of a well-fitting model. Moreover, the autocorrelation plots exhibit a quick decay in correlation as the lag increases for each parameter, which is indicative of good autocorrelation properties.

Moving onto the numerical diagnostics of our MCMC simulation, we have calculated the neff_ratio and R-hat statistics for each parameter. In terms of numerical diagnostics, the neff_ratio statistic shows that all parameters have a ratio larger than 0.5, indicating a good effective sample size. Additionally, the R-hat statistic shows that all parameters have a value close to 1, indicating good convergence of the chains. These numerical diagnostics further confirm that our posterior simulation has stabilized.

Taken together, the results of both the visual and numerical diagnostics provide strong evidence that our posterior simulation has sufficiently stabilized, and that we can trust the results of our Bayesian analysis.

2.6.2 Tidy summary

Next, we will produce a tidy() summary of this model. In addition, we will interpret the non-intercept coefficients' posterior median values in context.

```
tidy(birth_model, effects = c("fixed", "aux"),
     conf.int = TRUE, conf.level = 0.80)
```

```
## # A tibble: 8 x 5
##   term              estimate std.error  conf.low conf.high
##   <chr>                <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)        3150.       13.7  3133.       3168.
## 2 centered_combgest   126.        4.91  120.        133.
## 3 cig_rec1           -226.       22.0  -254.       -198.
## 4 centered_bmi         15.3       1.32   13.7        17.0
## 5 centered_mager        1.90      1.44    0.0602      3.74
## 6 dmar1                80.5      10.8    66.9        94.3
## 7 sigma               476.        8.51  465.        487.
## 8 mean_PPD           3183.       16.8  3162.       3205.
```

Interpretations of coefficients of some key variables:

- The posterior median value of the intercept in the model is 3150, indicating that on average, the expected weight of a female newborn whose mother has average age, bmi, gestation length, is not married and doesn't smoke is 3150 grams.
- The posterior median value of the centered_combgest coefficient is 126, suggesting that for each additional gestational week, the expected weight of the female newborn increases by 126 grams, holding all other predictors in the model constant.
- The posterior median value of the cig_rec1 coefficient is -226, indicating that mothers who smoke during pregnancy are expected have female newborns that weigh around 226 grams less on average than non-smoking mothers, holding all other predictors in the model constant.

2.6.3 Posterior Summary

Here, we will use posterior_interval() to produce 95% credible intervals for the model parameters. After that, we will check any association between independent variables and dependent variables.

```
# posterior summarizes
posterior_interval(birth_model, prob = 0.95)
```

```
##                           2.5%       97.5%
## (Intercept)       3123.2620344 3176.441993
## centered_combgest  116.7684188  135.822340
## cig_rec1          -269.8335651 -182.977102
## centered_bmi        12.7569316   17.902514
## centered_mager      -0.9424505    4.735552
## dmar1               59.4766916  101.441087
## sigma              459.3819380  492.589002
```

When controlling for the other predictors in the model, the 95% posterior credible intervals for the coefficient of centered_combgest, centered_bmi, and dmar1 lie entirely above 0, suggesting that centered_combgest,

comtered_bmi, and dmar1 have significant positive associations with dbwt. The 95% posterior credible intervals for the coefficient of centered_mager contains 0, suggesting that centered_mager has no association with dbwt. The 95% posterior credible intervals for the coefficient of cig_rec1 lie entirely below 0, suggesting that cig_rec1 has significant negative association with dbwt. In other words, the model suggests that increasing the gestational period (centered_combgest) and maternal BMI (centered_bmi) are associated with higher birthweights of female newborns, as is being married (dmar1), and smoking (cig_rec1) is associated with a significant decrease in the birthweight.

2.6.4 More models

Next, we will bring more models and see which ones work the best. For our new model birth_model2, we are going to add two interaction terms: cig_rec&bmi and mager&dmar. We believe that mother's age and her marital status should be correlated, and smokers probably would have lower BMI. To fit birth_model2, we will use weakly informative priors throughout.

```
# Use weakly informative criteria throughout
# Add interaction: cig_rec*centered_bmi, centered_mager*dmar
birth_model2 <- stan_glm(dbwt ~ centered_combgest + cig_rec + centered_bmi + centered_mager + dmar + cig
                         data = birth,
                         family = gaussian,
                         prior_intercept = normal(3200, 750),
                         prior = normal(0, 2.5, autoscale = TRUE),
                         prior_aux = exponential(1, autoscale = TRUE),
                         chains = 4, iter = 5000*2, seed = 84735,
                         prior_PD = FALSE)
```

Next, to determine the necessity of interaction terms, we will check the 80% posterior credible interval for interaction coefficients.

```
# Posterior summary statistics
tidy(birth_model2, effects = c("fixed", "aux"))
```

```
## # A tibble: 10 x 3
##    term                   estimate std.error
##    <chr>                     <dbl>     <dbl>
##  1 (Intercept)             3145.       20.9
##  2 centered_combgest        126.        4.75
##  3 cig_rec1                -140.       48.1
##  4 centered_bmi               7.71      1.77
##  5 centered_mager             7.70      3.27
##  6 dmar1                     90.6      26.0
##  7 cig_rec1:centered_bmi      6.26      7.09
##  8 centered_mager:dmar1      -7.07      4.39
##  9 sigma                    473.        8.50
## 10 mean_PPD                3183.       16.8
```

```
# Posterior credible interval for the interaction terms
posterior_interval(birth_model2, prob = 0.80,
                   pars = "cig_rec1:centered_bmi")
```

```
##                              10%      90%
## cig_rec1:centered_bmi -2.781384 15.31775
```

```
posterior_interval(birth_model2, prob = 0.80,
                   pars = "centered_mager:dmar1")
```

```
##                             10%       90%
## centered_mager:dmar1 -12.6148 -1.407235
```

The 80% posterior credible interval for interaction coefficient $\beta 6$ contains 0, suggesting that the association between smoking and BMI is not significant. Thus, this interaction term is not necessary. The 80% posterior credible interval for interaction coefficient $\beta 7$ is entirely and well below 0, suggesting a negative association between mother's age and marriage status. And we believe that we should include this interaction term in our new model birth_model3.

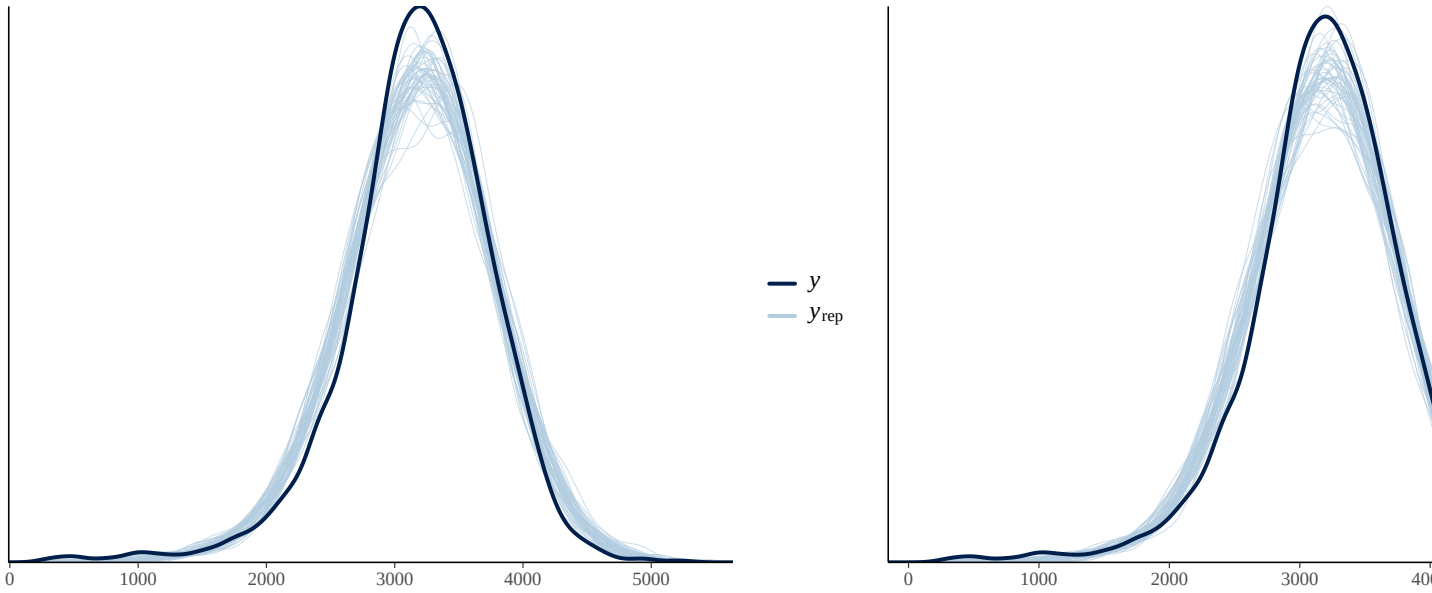
```
# Use weakly informative criteria throughout
# Add interaction: centered_mager*dmar
birth_model3 <- stan_glm(dbwt ~ centered_combgest + cig_rec + centered_bmi + centered_mager + dmar + cen
                         data = birth,
                         family = gaussian,
                         prior_intercept = normal(3200, 750),
                         prior = normal(0, 2.5, autoscale = TRUE),
                         prior_aux = exponential(1, autoscale = TRUE),
                         chains = 4, iter = 5000*2, seed = 84735,
                         prior_PD = FALSE)
```

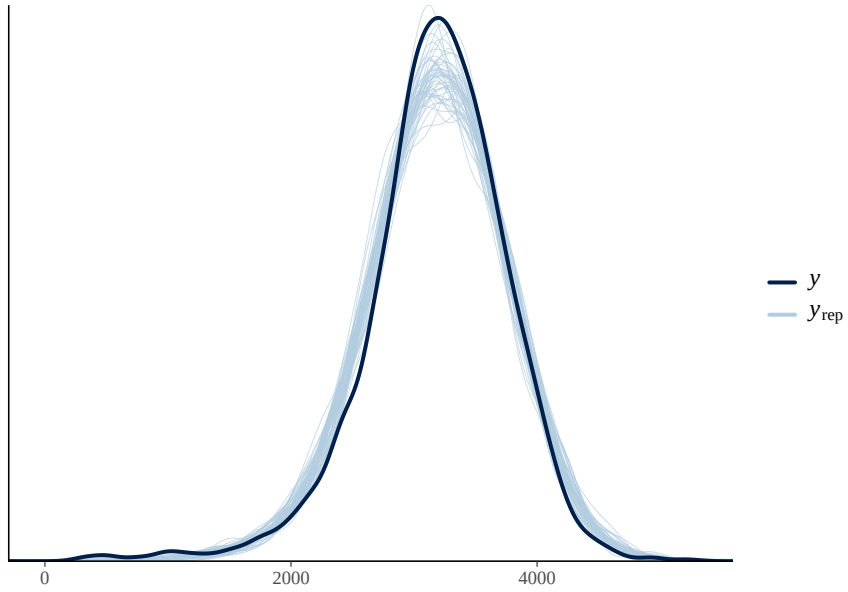
2.6.5 Model Comparison

To compare our 3 models and decide which one is the best, we will use three different approaches, including posterior predictive checks, cross-validation, and ELPD.

The first method is to evaluate predictive accuracy using pp_check.

```
pp_check(birth_model)
pp_check(birth_model2)
pp_check(birth_model3)
```





According to pp_check() output, all three models preform pretty good. Although the simulations are not perfect, but they do reasonably capture the features of the observed data.

The second method is to evaluate predictive accuracy using cross-validation.

```
set.seed(84735)

# Run a 10-fold cross-validation
cv_procedure1 <- prediction_summary_cv(model = birth_model, data = birth, k = 10)
cv_procedure2 <- prediction_summary_cv(model = birth_model2, data = birth, k = 10)
cv_procedure3 <- prediction_summary_cv(model = birth_model3, data = birth, k = 10)

# Compare 3 model's mean
cv_procedure1$cv
```

```
##        mae mae_scaled within_50 within_95
## 1 319.9283  0.6723858 0.5116234 0.9522131
```

```
cv_procedure2$cv
```

```
##        mae mae_scaled within_50 within_95
## 1 311.2236  0.6550167 0.5116156  0.948467
```

```
cv_procedure3$cv
```

```
##        mae mae_scaled within_50 within_95
## 1 309.1355  0.6509468 0.5172799 0.9522327
```

Based on the cross-validated metrics provided, birth_model3 has the lowest mae, lowest mae_scaled, highest within_50, and highest within_95 than the other two models. Its lowest mae and mae_scaled value indicate that the this model on average makes most accurate predictions, and its highest within_50 and within_95 value indicate that the this model is more likely to make accurate predictions within the selected interval. Therefore, the best model is birth_model3.

The third method is to evaluate predictive accuracy using ELPD.

```r
# Calculate ELPD for the 3 models
set.seed(84735)
loo_1 <- loo(birth_model)
loo_2 <- loo(birth_model2)
loo_3 <- loo(birth_model3)

# Results
c(loo_1$estimates[1], loo_2$estimates[1], loo_3$estimates[1])
```

```
## [1] -12068.71 -12060.68 -12060.05
```

```r
# Compare the ELPD for the 3 models
loo_compare(loo_1, loo_2, loo_3)
```

```
##              elpd_diff se_diff
## birth_model3  0.0       0.0
## birth_model2 -0.6       0.9
## birth_model  -8.7       5.2
```

According to loo_compare() output, birth_model3 has the highest ELPD, indicating it is the best model out of three.

2.7 Discussion