Lab5

21/02/23

```r
library(tidyverse)
library(here)
# for bayes stuff
library(rstan)
library(bayesplot)
library(loo)
library(tidybayes)
library("fdrtool")
ds <- read_rds(("births_2017_sample.RDS"))

theme_set(theme_bw())
```

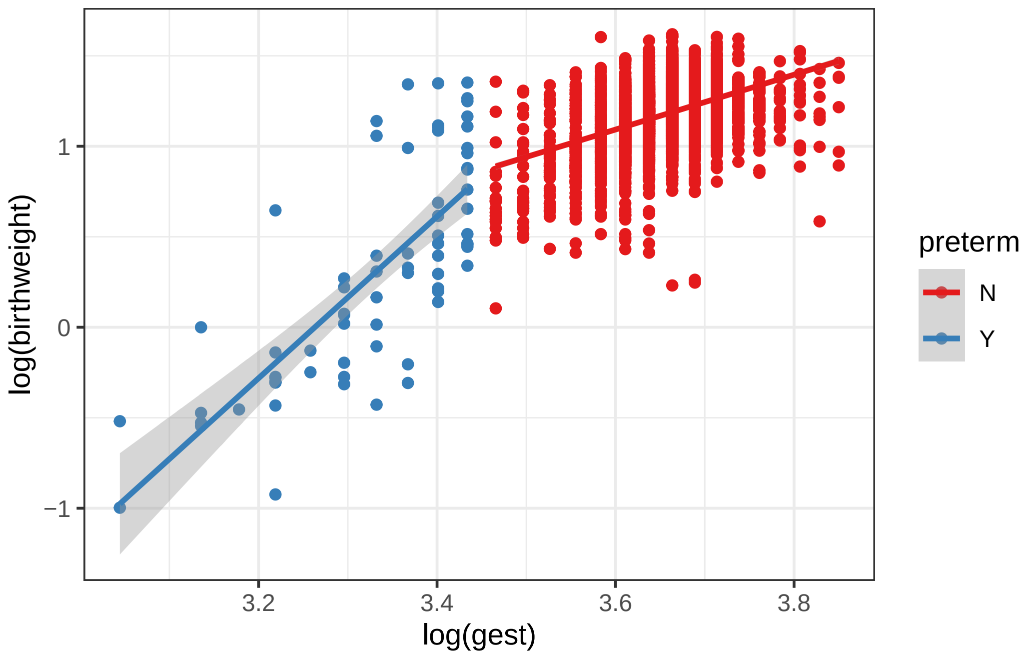
```r
ds <- ds %>%
  rename(birthweight = dbwt, gest = combgest) %>%
  mutate(preterm = ifelse(gest<32, "Y", "N")) %>%
  filter(ilive=="Y",gest< 99, birthweight<9.999)

ds$log_weight <- log(ds$birthweight)
ds$log_gest_c <- (log(ds$gest) - mean(log(ds$gest)))/sd(log(ds$gest))
```

Question 1

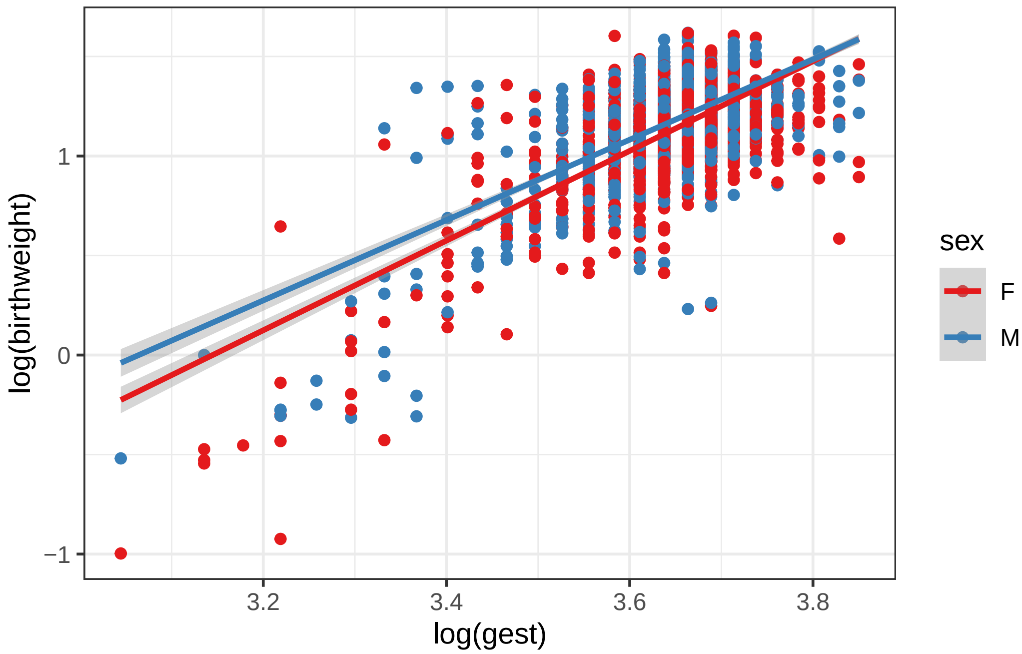
1. From the first scatterplot, we note that there is a linear relationship between the log birthweight and log gestational age, but the slope of the linear relationship differs whether the child was born preterm or not.
2. We further investigated whether the linear relationship between the log birthweight and log gestational age differs by the sex of the child. The second scatterplot shows that the slopes are similar.

3. Finally, we investigated whether the log birthweight differs by sex. We found that the birthweight between male and female babies differ the most when born premature.

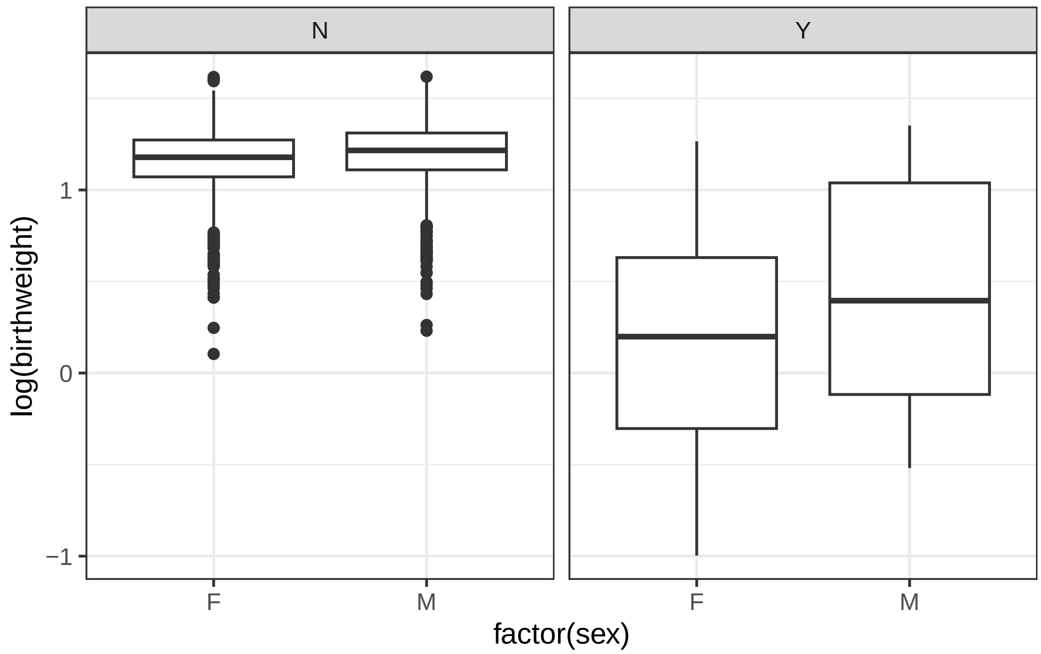
```
ds %>%
  ggplot(aes(log(gest), log(birthweight), color = preterm)) +
  geom_point() + geom_smooth(method = "lm") +
  scale_color_brewer(palette = "Set1")
```




```
ds %>%
  ggplot(aes(log(gest), log(birthweight), color = sex)) +
  geom_point() + geom_smooth(method = "lm") +
  scale_color_brewer(palette = "Set1")
```



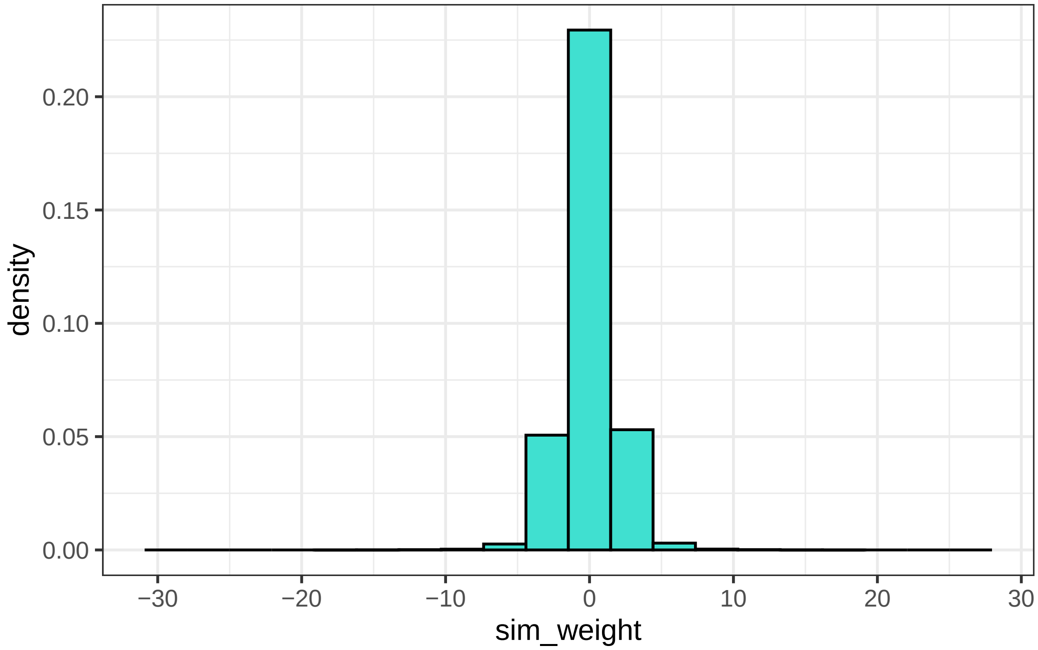

```
ds%>%
  ggplot(aes(x = factor(sex), y = log(birthweight))) +
  geom_boxplot() +
  facet_wrap(~preterm)
```



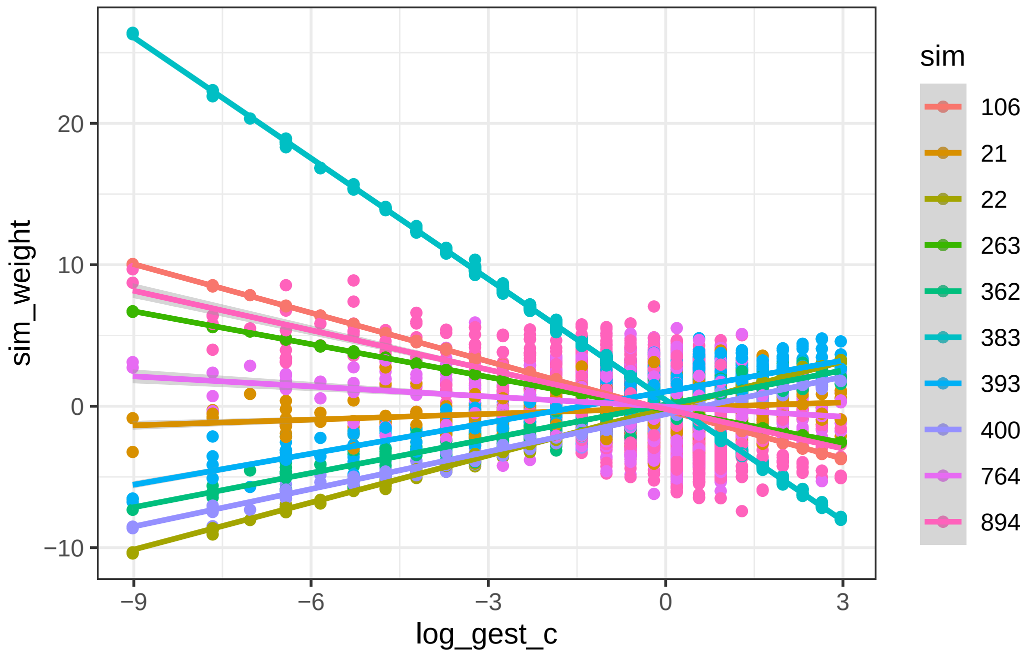
Question 2

```
set.seed(123)

# Simulated log birth weights

nsims <- 1000
sigma <- abs(rnorm(nsims, 0, 1))
beta0 <- rnorm(nsims, 0, 1)
beta1 <- rnorm(nsims, 0, 1)

dsims <- tibble(log_gest_c = (log(ds$gest)-mean(log(ds$gest)))/sd(log(ds$gest)))

for(i in 1:nsims){
  this_mu <- beta0[i] + beta1[i]*dsims$log_gest_c
  dsims[paste0(i)] <- this_mu + rnorm(nrow(dsims), 0, sigma[i])
}

dsl <- dsims %>%
  pivot_longer(`1`:`1000`, names_to = "sim", values_to = "sim_weight")

# Plot the histogram of simulated weights
```

```
dsl %>%
  ggplot(aes(sim_weight)) + geom_histogram(aes(y = ..density..), bins = 20, fill = "turquo
```




```
# Plot simulated weights against gestational age

dsl %>%filter(sim %in% sample(dsl$sim, 10)) %>%
  ggplot(aes(x = log_gest_c, y = sim_weight, color = sim)) +
  geom_point() + geom_smooth(method = "lm")
```




```
# Fit Model 1

stan_data <- list(N = nrow(ds),
                  log_weight = ds$log_weight,
                  log_gest = ds$log_gest_c)

mod1 <- stan(data = stan_data,
             file = ("simple_weight.stan"),
             iter = 500,
             seed = 243)

summary(mod1)$summary[c("beta[1]", "beta[2]", "sigma"),]
```

Question 3

Based on model 1, give an estimate of the expected birthweight of a baby who was born at a gestational age of 37 weeks.

```
m <- mean(log(ds$gest))
s <- sd(log(ds$gest))
exp(1.1626250 + 0.1436183*(log(37)-m)/s)
```

```
[1] 2.93654
```

Question 4

Write a stan model to run Model 2, and run it.

```
# Create preterm factor: If preterm = Y, set to 1

ds <- ds %>%
  mutate(preterm_factor = ifelse(preterm == "Y", 1,  0))


stan_data2 <- list(N = nrow(ds),
                  log_weight = ds$log_weight,
                  log_gest = ds$log_gest_c,
                  preterm = ds$preterm_factor,
                  log_gest_preterm = ds$log_gest_c * ds$preterm_factor)

# Fit model 2

my.mod2 <- stan(data = stan_data2,
             file = ("simple_weight_2.stan"),
             iter = 500,
             seed = 243)
```

Question 5

```
load(("mod2.Rda"))
summary(mod2)$summary[c(paste0("beta[", 1:4, "]"), "sigma"),]
```

```
             mean      se_mean          sd       2.5%        25%       50%
beta[1] 1.1697241 1.385590e-04 0.002742186 1.16453578 1.16767109 1.1699278
beta[2] 0.5563133 5.835253e-03 0.058054991 0.43745504 0.51708255 0.5561553
beta[3] 0.1020960 1.481816e-04 0.003669476 0.09459462 0.09997153 0.1020339
beta[4] 0.1967671 1.129799e-03 0.012458398 0.17164533 0.18817091 0.1974114
sigma   0.1610727 9.950037e-05 0.001782004 0.15784213 0.15978020 0.1610734
              75%     97.5%     n_eff      Rhat
beta[1] 1.1716235 1.1750167 391.67359 1.0115970
beta[2] 0.5990427 0.6554967  98.98279 1.0088166
```

```
beta[3] 0.1044230 0.1093843 613.22428 0.9978156
beta[4] 0.2064079 0.2182454 121.59685 1.0056875
sigma   0.1623019 0.1646189 320.75100 1.0104805
```

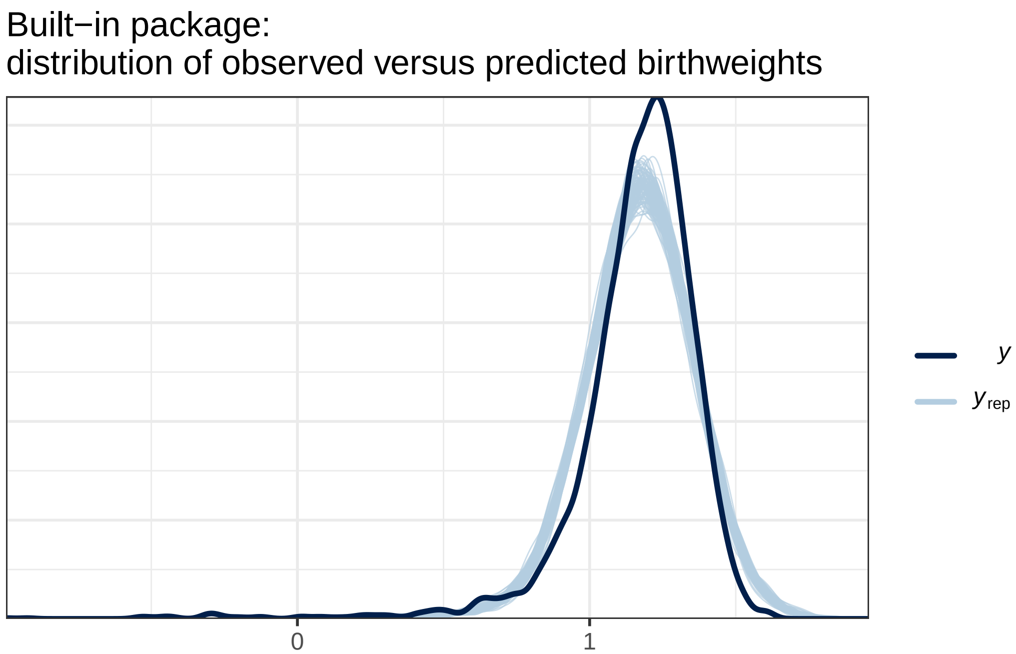
```r
summary(my.mod2)$summary[c(paste0("beta[", 1:4, "]"), "sigma"),]
```

```
             mean      se_mean          sd       2.5%        25%       50%
beta[1] 1.1695474 7.775215e-05 0.002730748 1.16432439 1.16774539 1.1694545
beta[2] 0.1020646 1.333365e-04 0.003540319 0.09526538 0.09965927 0.1020458
beta[3] 0.5634542 4.555121e-03 0.066073353 0.42870273 0.52058743 0.5622696
beta[4] 0.1983429 9.280153e-04 0.013709344 0.17026957 0.18956716 0.1986937
sigma   0.1611931 7.237822e-05 0.001813104 0.15776604 0.16004710 0.1610828
              75%     97.5%     n_eff     Rhat
beta[1] 1.1714191 1.1746801 1233.4982 1.000121
beta[2] 0.1044228 0.1088863  704.9956 1.008594
beta[3] 0.6072238 0.6902591  210.4035 1.031391
beta[4] 0.2076363 0.2246584  218.2343 1.027843
sigma   0.1624286 0.1648346  627.5230 1.000475
```

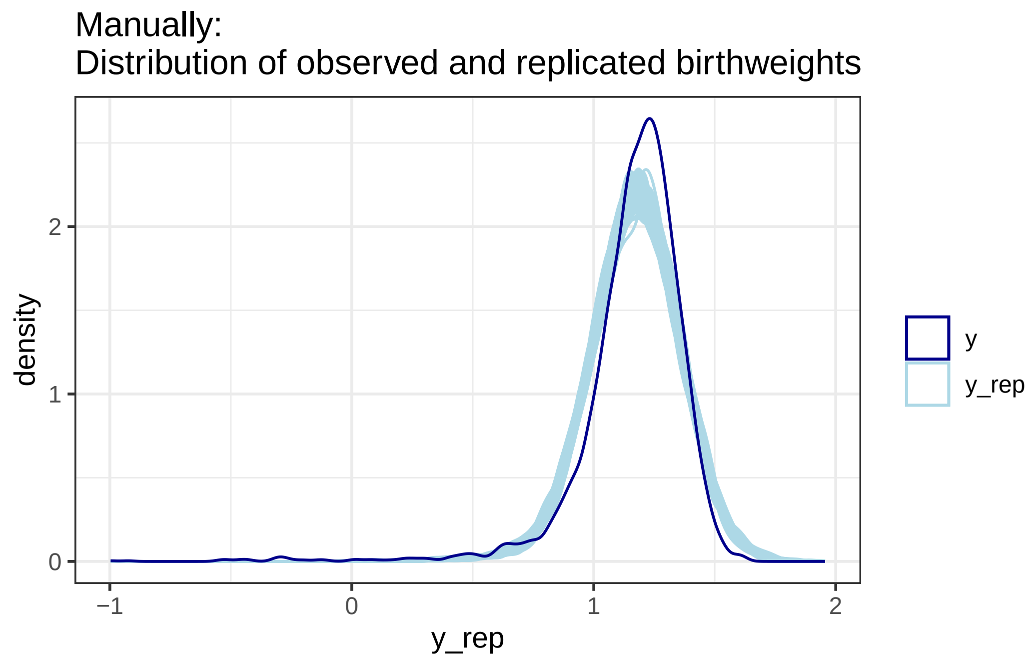
The results are similar.

Question 6

```r
# Posterior predictive checks

set.seed(1856)
y <- ds$log_weight
yrep1 <- extract(mod1)[["log_weight_rep"]]
yrep2 <- extract(my.mod2)[["log_weight_rep"]]
samp100 <- sample(nrow(yrep2), 100)

# From Bayes package
ppc_dens_overlay(y, yrep2[samp100, ])  +
  ggtitle("Built-in package: \ndistribution of observed versus predicted birthweights")
```




```
# From 'scratch'

N <- nrow(ds)
rownames(yrep2) <- 1:nrow(yrep2)
dr <- as_tibble(t(yrep2))
dr <- dr %>% bind_cols(i = 1:N, log_weight_obs = log(ds$birthweight))

# turn into long format; easier to plot
dr <- dr %>%
  pivot_longer(-(i:log_weight_obs), names_to = "sim", values_to ="y_rep")

# filter to just include 100 draws and plot!
dr %>%
  filter(sim %in% samp100) %>%
  ggplot(aes(y_rep, group = sim)) +
  geom_density(alpha = 0.2, aes(color = "y_rep")) +
  geom_density(data = ds %>% mutate(sim = 1),
               aes(x = log(birthweight), col = "y")) +
  scale_color_manual(name = "",
                     values = c("y" = "darkblue",
                                "y_rep" = "lightblue")) +
  ggtitle("Manually: \nDistribution of observed and replicated birthweights")
```

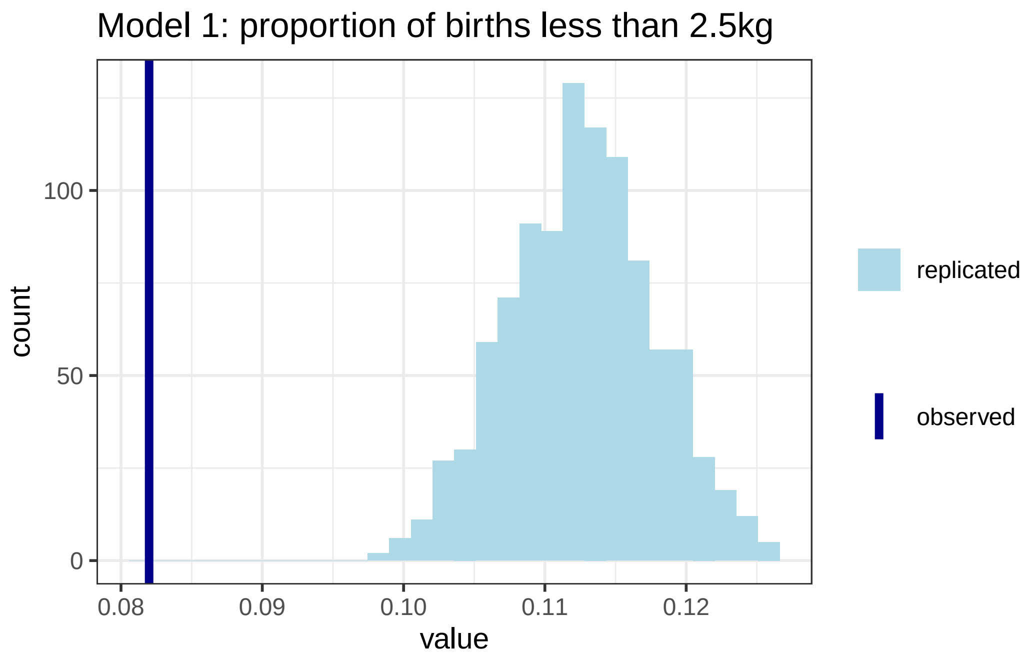


Question 7

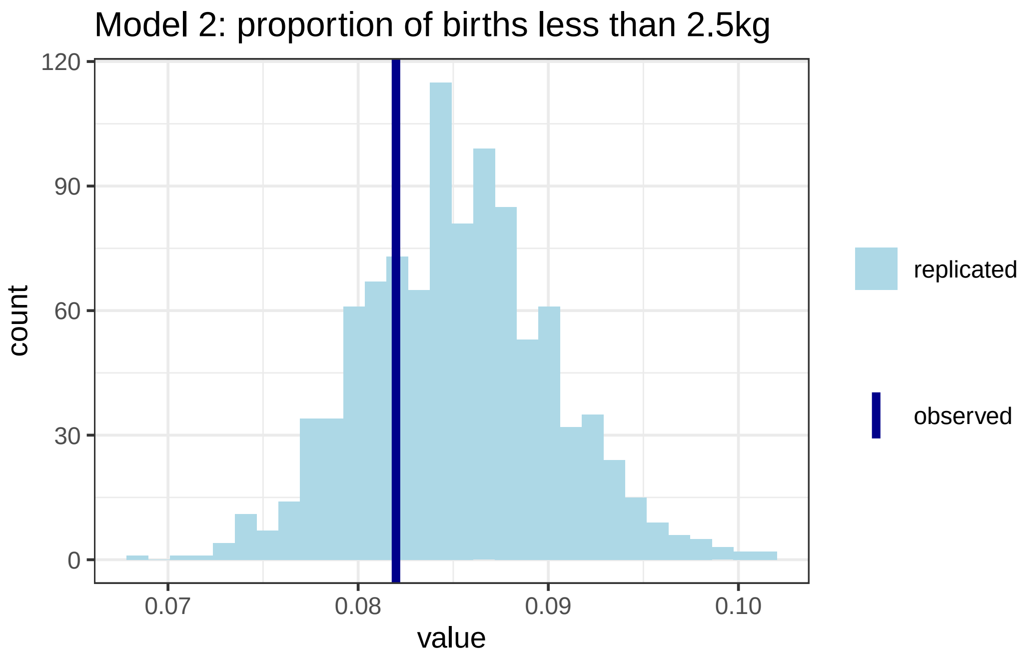
Use a test statistic of the proportion of births under 2.5kg. Calculate the test statistic for the data, and the posterior predictive samples for both models, and plot the comparison (one plot per model).

```
# Calculate test statistic

#ppc_stat_grouped(ds$log_weight, yrep2, group = ds$birthweight <2.5, stat = 'mean')


t_y <- mean(y<=log(2.5))
t_y_rep_1<- sapply(1:nrow(yrep1), function(i) mean(yrep1[i,]<=log(2.5)))
t_y_rep_2 <- sapply(1:nrow(yrep2), function(i) mean(yrep2[i,]<=log(2.5)))


ggplot(data = as_tibble(t_y_rep_1), aes(value)) +
    geom_histogram(aes(fill = "replicated")) +
    geom_vline(aes(xintercept = t_y, color = "observed"), lwd = 1.5) +
  ggtitle("Model 1: proportion of births less than 2.5kg") +
  scale_color_manual(name = "",
                     values = c("observed" = "darkblue"))+
```

```
  scale_fill_manual(name = "",
                    values = c("replicated" = "lightblue"))
```




```
ggplot(data = as_tibble(t_y_rep_2), aes(value)) +
    geom_histogram(aes(fill = "replicated")) +
    geom_vline(aes(xintercept = t_y, color = "observed"), lwd = 1.5) +
  ggtitle("Model 2: proportion of births less than 2.5kg") +
  scale_color_manual(name = "",
                    values = c("observed" = "darkblue"))+
  scale_fill_manual(name = "",
                    values = c("replicated" = "lightblue"))
```



We note that for model 1, the posterior proportions of births less than 2.5kg are all above the observed proportion. However in model 2, the mean posterior proportion of births is close to the observed proportion.

Question 8

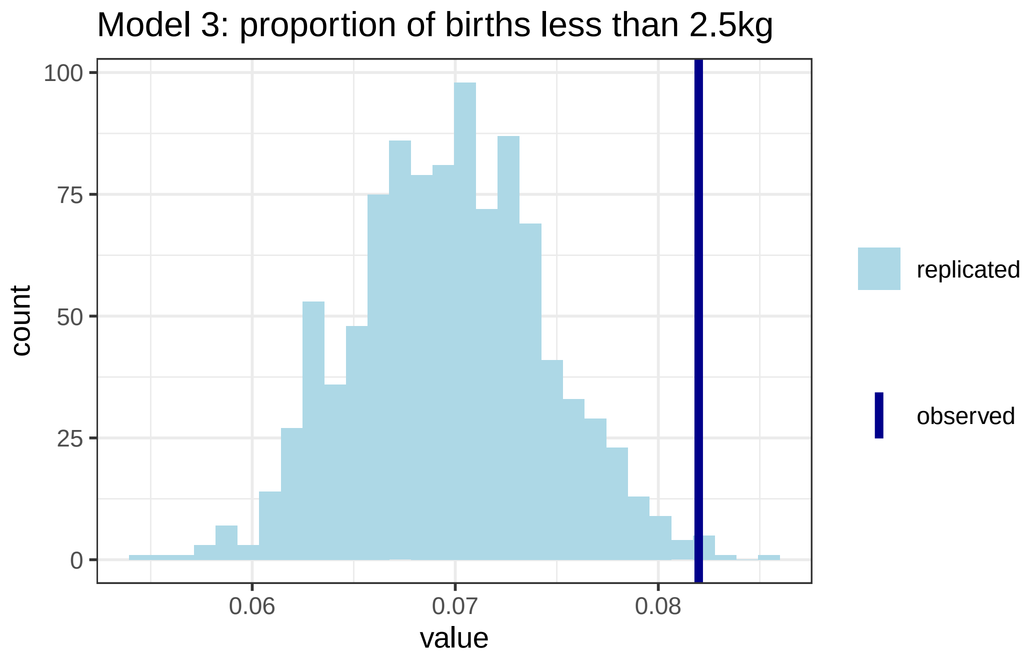
Based on the EDA in (a), we add sex and the interaction effect between sex and preterm to model 2.

Model 3 does not perform better than model 2 since the posterior proportions of births > 2.5kg mostly lie below the observed proportion. Hence the predictions produced by model 3 are not better than model 2.

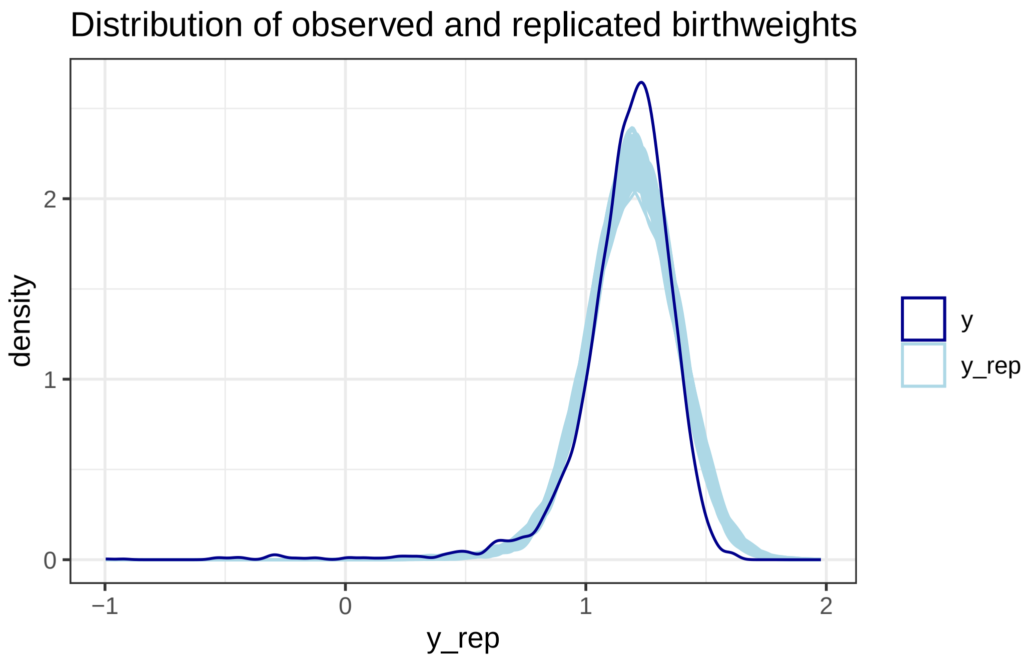
```
ds <- ds %>%
  mutate(sex_factor = ifelse(sex == "F", 1,  0))

# put into a list
stan_data3 <- list(N = nrow(ds),
                   log_weight = ds$log_weight,
                   log_gest = ds$log_gest_c,
                   preterm = ds$preterm_factor,
                   log_gest_preterm = ds$log_gest_c * ds$preterm_factor,
                   sex = ds$sex_factor,
```

```
                  sex_preterm = ds$sex_factor*ds$preterm_factor)

# Fit model 3

my.mod3 <- stan(data = stan_data3,
             file = ("simple_weight_3.stan"),
             iter = 500,
             seed = 243)
```

```
# Posterior predictive checks

yrep3 <- extract(my.mod3)[["log_weight_rep"]]
t_y_rep_3 <- sapply(1:nrow(yrep3), function(i) mean(yrep3[i,]<=log(2.5)))

# Proportion under 2.5kg

ggplot(data = as_tibble(t_y_rep_3), aes(value)) +
    geom_histogram(aes(fill = "replicated")) +
    geom_vline(aes(xintercept = t_y, color = "observed"), lwd = 1.5) +
  ggtitle("Model 3: proportion of births less than 2.5kg") +
  scale_color_manual(name = "",
                     values = c("observed" = "darkblue"))+
  scale_fill_manual(name = "",
                     values = c("replicated" = "lightblue"))
```




```
# Density of posterior predictive samples

dr3 <- as_tibble(t(yrep3))
dr3 <- dr3 %>% bind_cols(i = 1:N, log_weight_obs = log(ds$birthweight))

# turn into long format; easier to plot
dr3 <- dr3 %>%
  pivot_longer(-(i:log_weight_obs), names_to = "sim", values_to ="y_rep")

# filter to just include 100 draws and plot!
dr3 %>%
  filter(sim %in% paste0("V",samp100)) %>%
  ggplot(aes(y_rep, group = sim)) +
  geom_density(alpha = 0.2, aes(color = "y_rep")) +
  geom_density(data = ds %>% mutate(sim = 1),
               aes(x = log(birthweight), col = "y")) +
  scale_color_manual(name = "",
                     values = c("y" = "darkblue",
                                "y_rep" = "lightblue")) +
  ggtitle("Distribution of observed and replicated birthweights")
```



Distribution of observed and replicated birthweights
density
2
1
0
−1
0
1
2
y_rep
y
y_rep