Lab Week 6

Rosa Fallahpour

```
library(tidyverse)
library(here)
library(rstan)
library(bayesplot)
library(loo)
library(tidybayes)
```

```
ds <- read_rds(here("births_2017_sample.RDS"))
head(ds)
```

```
## # A tibble: 6 x 8
##   mager mracehisp meduc   bmi sex   combgest  dbwt ilive
##   <dbl>     <dbl> <dbl> <dbl> <chr>    <dbl> <dbl> <chr>
## 1    16         2     2  23   M           39  3.18 Y
## 2    25         7     2  43.6 M           40  4.14 Y
## 3    27         2     3  19.5 F           41  3.18 Y
## 4    26         1     3  21.5 F           36  3.40 Y
## 5    28         7     2  40.6 F           34  2.71 Y
## 6    31         7     3  29.3 M           35  3.52 Y
```

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

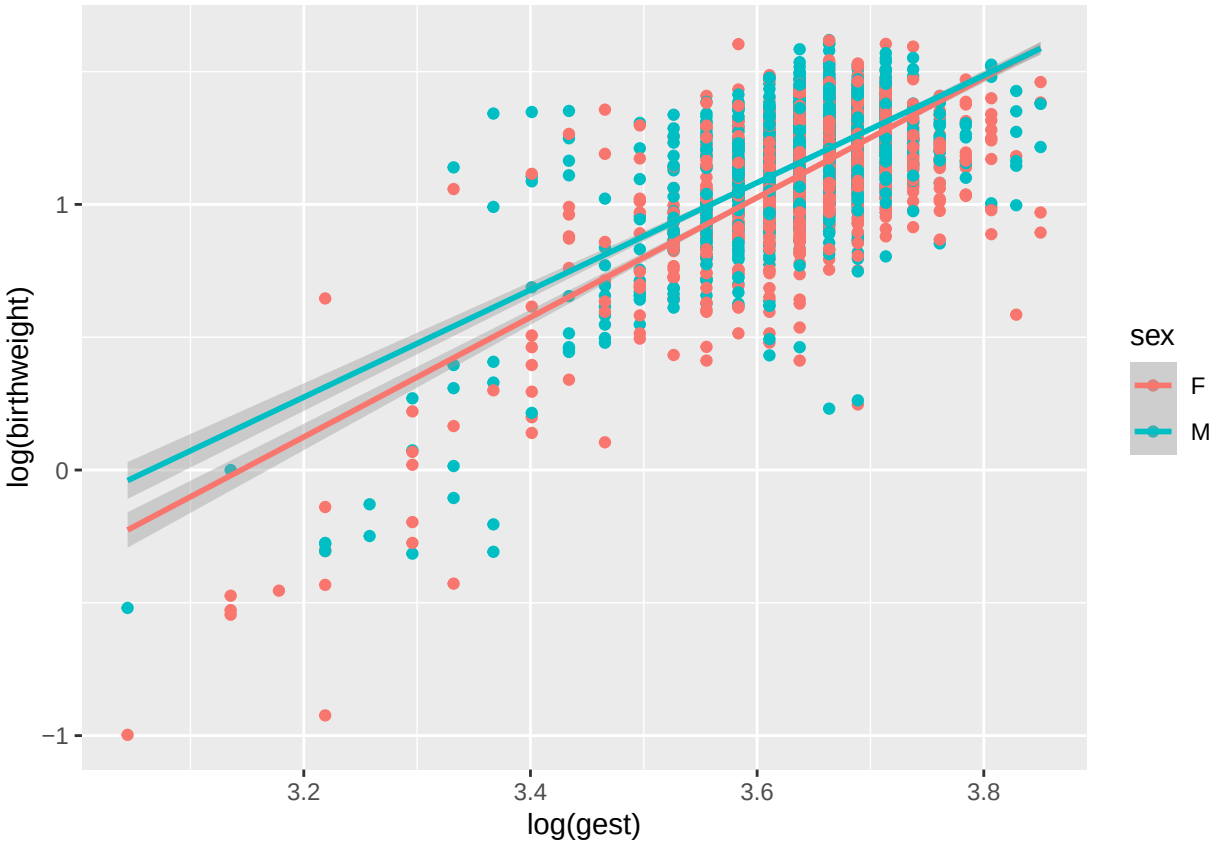
```
## # A tibble: 6 x 9
##   mager mracehisp meduc   bmi sex    gest birthweight ilive preterm
##   <dbl>     <dbl> <dbl> <dbl> <chr> <dbl>       <dbl> <chr> <chr>
## 1    16         2     2  23   M        39        3.18 Y     N
## 2    25         7     2  43.6 M        40        4.14 Y     N
## 3    27         2     3  19.5 F        41        3.18 Y     N
## 4    26         1     3  21.5 F        36        3.40 Y     N
## 5    28         7     2  40.6 F        34        2.71 Y     N
## 6    31         7     3  29.3 M        35        3.52 Y     N
```

Question 1

The following plot displays the relationship between log birth weight and log gestational age for both genders. As we can see there is a positive relationship between these two variables. As the gestational age increases, the birth weight also increases. It also shows the similar behavior for both genders. However, we see a

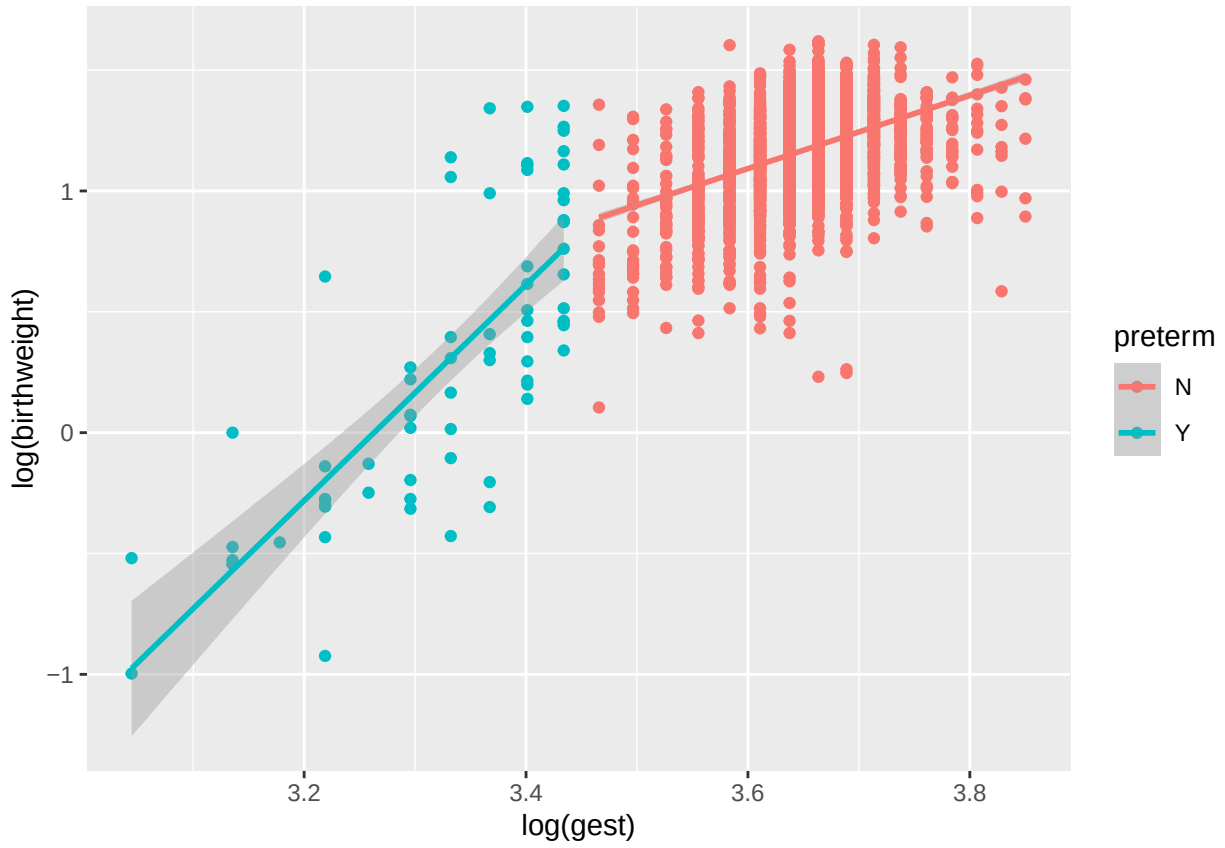
higher birth weight for baby boy rather than baby girls in the same gestational age at the early stages. This difference decreases as the gestational age increases.

```
p1 <- ds |> ggplot(aes(x=log(gest), y=log(birthweight), color=sex))+geom_point()+geom_smooth(method = lm
p1
```



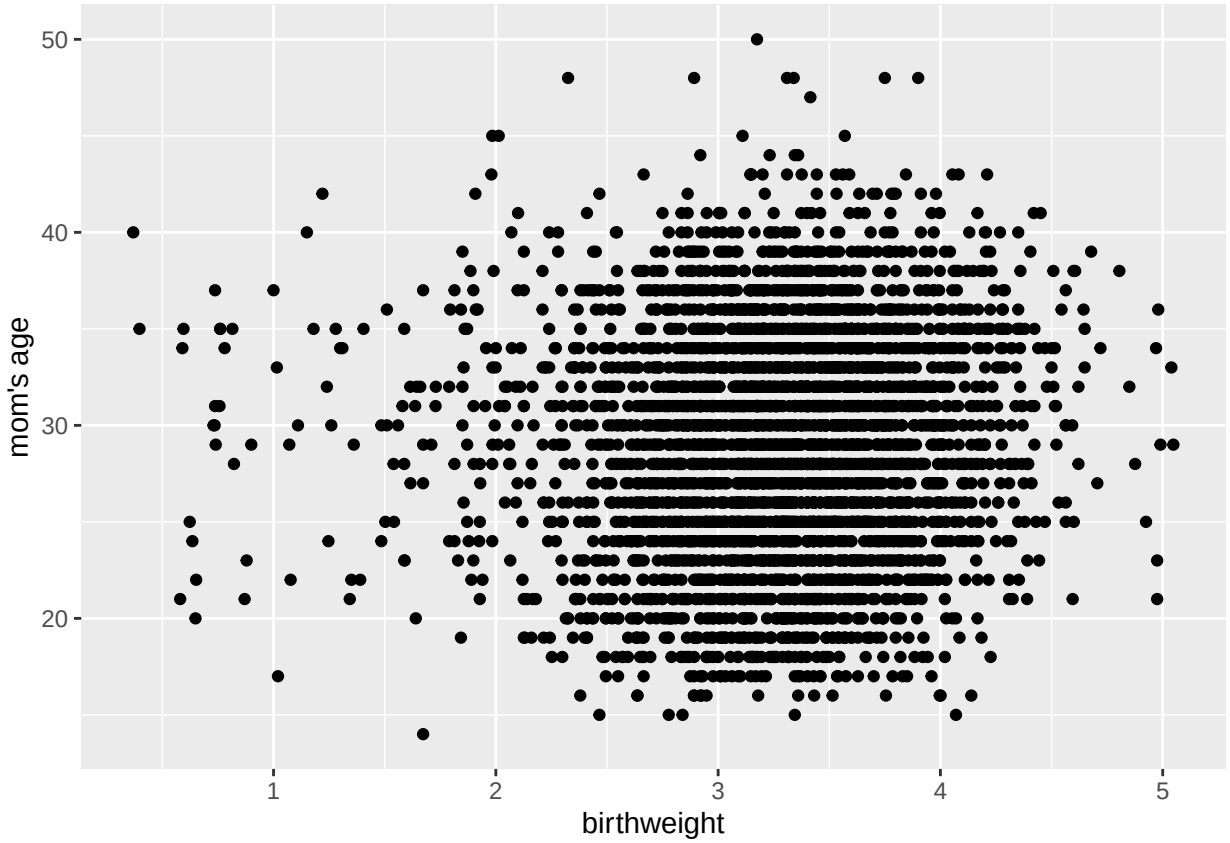
Below represents the graph for relation between log birth weight and the gestational age considering two groups whether gestational age is at least 32 weeks or not. As we can see there is a positive relationship between birth weight and gestational age for both categories. It should be noted that the relationship between these two variables is much stronger for the group with gestational age less than 32 weeks than group with greater than 32 weeks.

```
p2 <- ds |> ggplot(aes(x=log(gest), y=log(birthweight), color=preterm))+geom_point()+geom_smooth(method
p2
```



The following plot represents the relationship between birth weight and mom's age. It is interesting to see that the birth weight for different mom's ages is in an almost similar range.

```
p3 <- ds |> ggplot(aes(x=birthweight, y=mager))+labs(y="mom's age")+geom_point()
p3
```

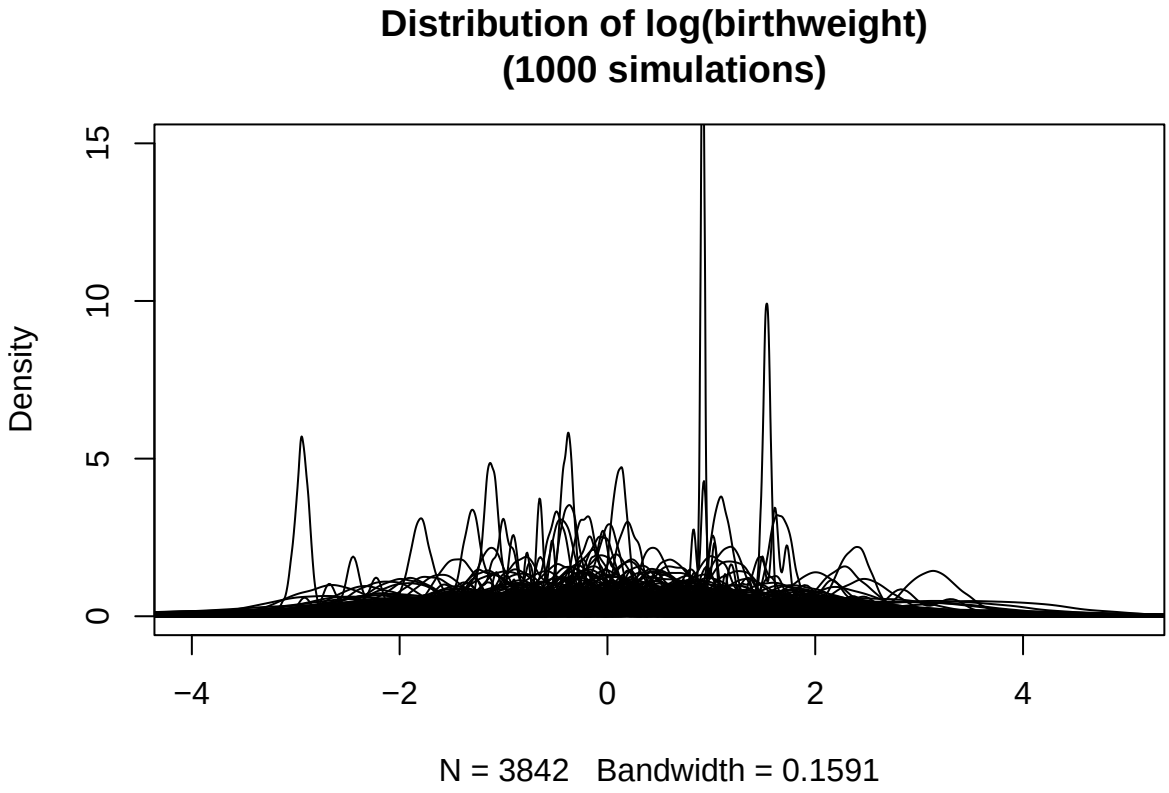


Question 2

The following graph represents the distribution of simulated (log) birth weights:

```
y <- c()
for(i in 1:1000){
  beta1 <- rnorm(1, 0,1)
  beta2 <- rnorm(1, 0,1)
  sigma <- abs(rnorm(1, 0,1))
  gest_cs <- (log(ds$gest)-mean(log(ds$gest)))/sd(log(ds$gest))
  mu <- beta1 + beta2 *gest_cs
  yi <- rnorm(length(mu), mean = mu, sd = sigma)
  y <- c(y, yi)
}
```

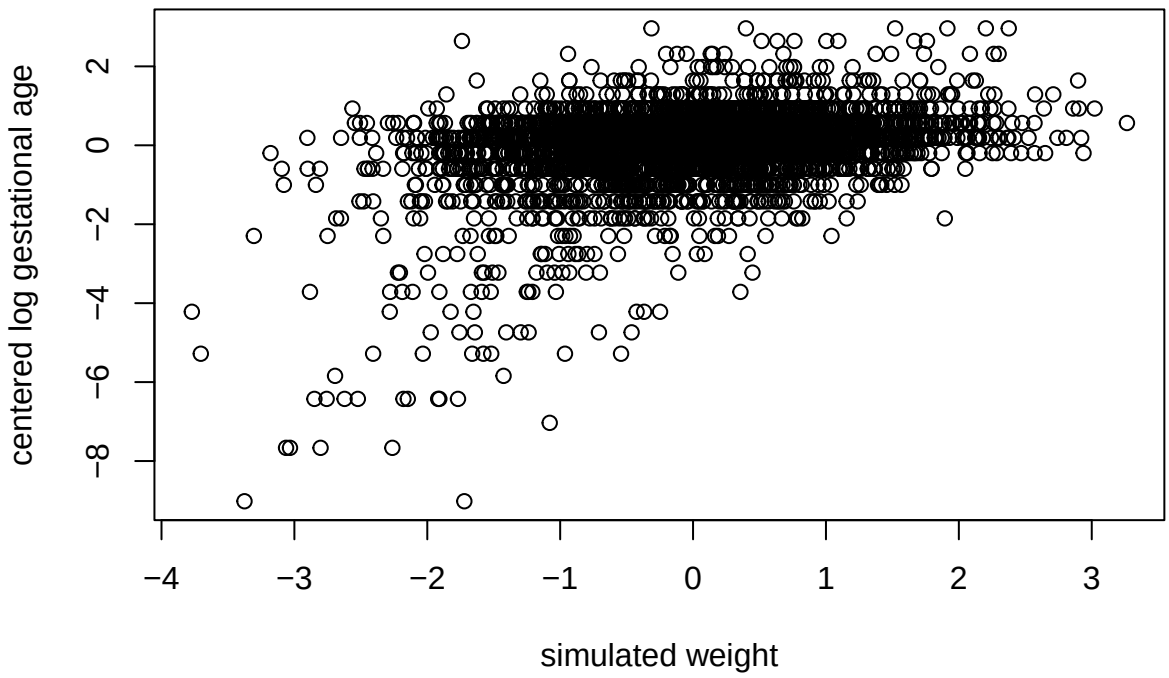
```
plot(density(y[1:3842]), xlim=c(-4,5),ylim=c(0,15), main = "Distribution of log(birthweight) \n (1000 s
for(i in 1:999){
    lines(density(y[(i*3842+1):((i+1)*3842)]))
}
```



Plot shown below is ten simulations of (log) birthweight against gestational age:

```
y <- c()
for(i in 1:10){
  beta1 <- rnorm(1, 0,1)
  beta2 <- rnorm(1, 0,1)
  sigma <- abs(rnorm(1, 0,1))
  gest_cs1 <- (log(ds$gest)-mean(log(ds$gest)))/sd(log(ds$gest))
  mu <- beta1 + beta2 *gest_cs1
  yi <- rnorm(length(mu), mean = mu, sd = sigma)
  y <- c(y, yi)
}
```

```
plot(y[1:3842], gest_cs1[1:3842],xlab="simulated weight" ,ylab="centered log gestational age",main = ""
for(i in 1:9){
    lines(y[(i*3842+1):((i+1)*3842)], gest_cs1[(i*3842+1):((i+1)*3842)])
}
```



Run the model

```
ds$log_weight <- log(ds$birthweight)
ds$log_gest_c <- (log(ds$gest) - mean(log(ds$gest)))/sd(log(ds$gest))
stan_data <- list(N = nrow(ds),
                  log_weight = ds$log_weight,
                  log_gest = ds$log_gest_c)
```

We fit model 1, below is the summary:

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

```
##              mean      se_mean          sd      2.5%       25%       50%
## beta[1] 1.1624783 8.160385e-05 0.002856578 1.1570200 1.1604786 1.1625011
## beta[2] 0.1437529 8.295075e-05 0.002912236 0.1381284 0.1416970 0.1436747
## sigma   0.1690330 1.113724e-04 0.001902828 0.1652694 0.1677842 0.1690763
##               75%     97.5%     n_eff      Rhat
## beta[1] 1.1644669 1.1681028 1225.3801 0.9978044
## beta[2] 0.1456716 0.1495180 1232.5721 0.9998714
## sigma   0.1702528 0.1727953  291.9066 1.0146111
```

Question 3

An estimate of the expected birthweight of a baby who was born at a gestational age of 37 weeks is as below:

```
x_new <- (log(37) - mean(log(ds$gest)))/sd(log(ds$gest))
postsample <- rstan ::extract(mod1)
beta1_hat <- median(postsample[["beta"]][,1])
beta2_hat <- median(postsample[["beta"]][,2])
exp(beta1_hat+beta2_hat*x_new)
```

```
## [1] 2.936078
```

Question 4

The summary of model 2 is as below:

```
ds$z <- ifelse(ds$preterm == "Y", 1, 0)
ds$z_log_gest <- ds$z * ds$log_gest_c
stan_data_2 <- list(N = nrow(ds),
                    log_weight = ds$log_weight,
                    log_gest = ds$log_gest_c,
                    z = ds$z,
                    z_log_gest = ds$z_log_gest)
```

```
summary(my_mod_2)$summary[c("beta[1]", "beta[2]", "beta[3]", "beta[4]", "sigma"),]
```

```
##              mean      se_mean          sd       2.5%        25%       50%
## beta[1] 1.1696129 7.671989e-05 0.002653883 1.16442098 1.16774305 1.1696337
## beta[2] 0.1019609 1.162120e-04 0.003685496 0.09474375 0.09944551 0.1021107
## beta[3] 0.5613108 3.194235e-03 0.062128569 0.43739411 0.51901011 0.5619125
## beta[4] 0.1980210 6.485573e-04 0.012644381 0.17316581 0.18926424 0.1975173
## sigma   0.1612369 7.878029e-05 0.001815088 0.15746513 0.16005849 0.1611716
##               75%     97.5%     n_eff     Rhat
## beta[1] 1.1713966 1.1745912 1196.5962 0.999004
## beta[2] 0.1045111 0.1091967 1005.7489 1.003601
## beta[3] 0.6021147 0.6879296  378.3110 1.008267
## beta[4] 0.2061977 0.2233707  380.1005 1.005890
## sigma   0.1624537 0.1647784  530.8360 1.008802
```

Question 5

The comparison between reference model and my model shows that beta2 and beta3 has switched their values. Otherwise, the results are very close.

```
load(here("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
```

PPCs

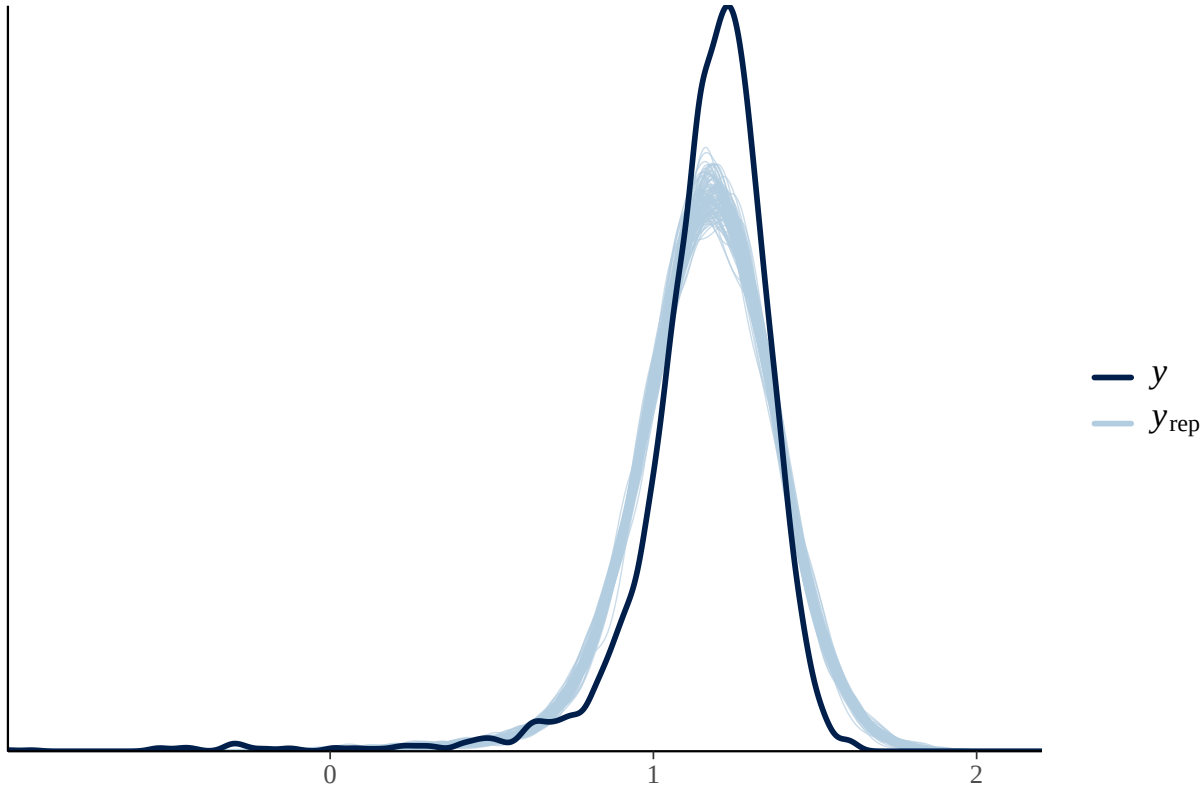
The distribution of data (y) against 100 different datasets drawn from the posterior predictive distribution:

```
set.seed(1856)
y <- ds$log_weight
yrep1 <- extract(mod1)[["log_weight_rep"]]
dim(yrep1)
```

```
## [1] 1000 3842
```

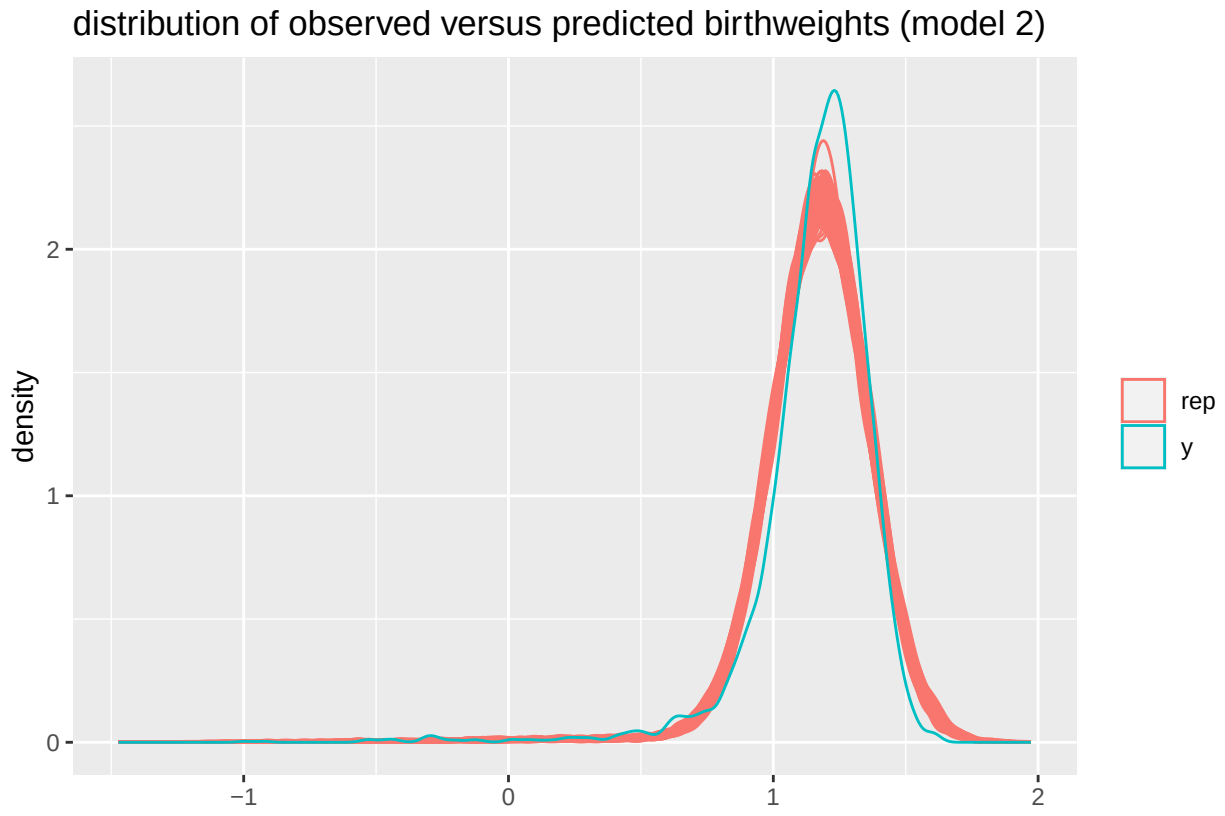
```
samp100 <- sample(nrow(yrep1), 100)
ppc_dens_overlay(y, yrep1[samp100, ])  + ggtitle("distribution of observed versus predicted birthweights
```

distribution of observed versus predicted birthweights



Question 6

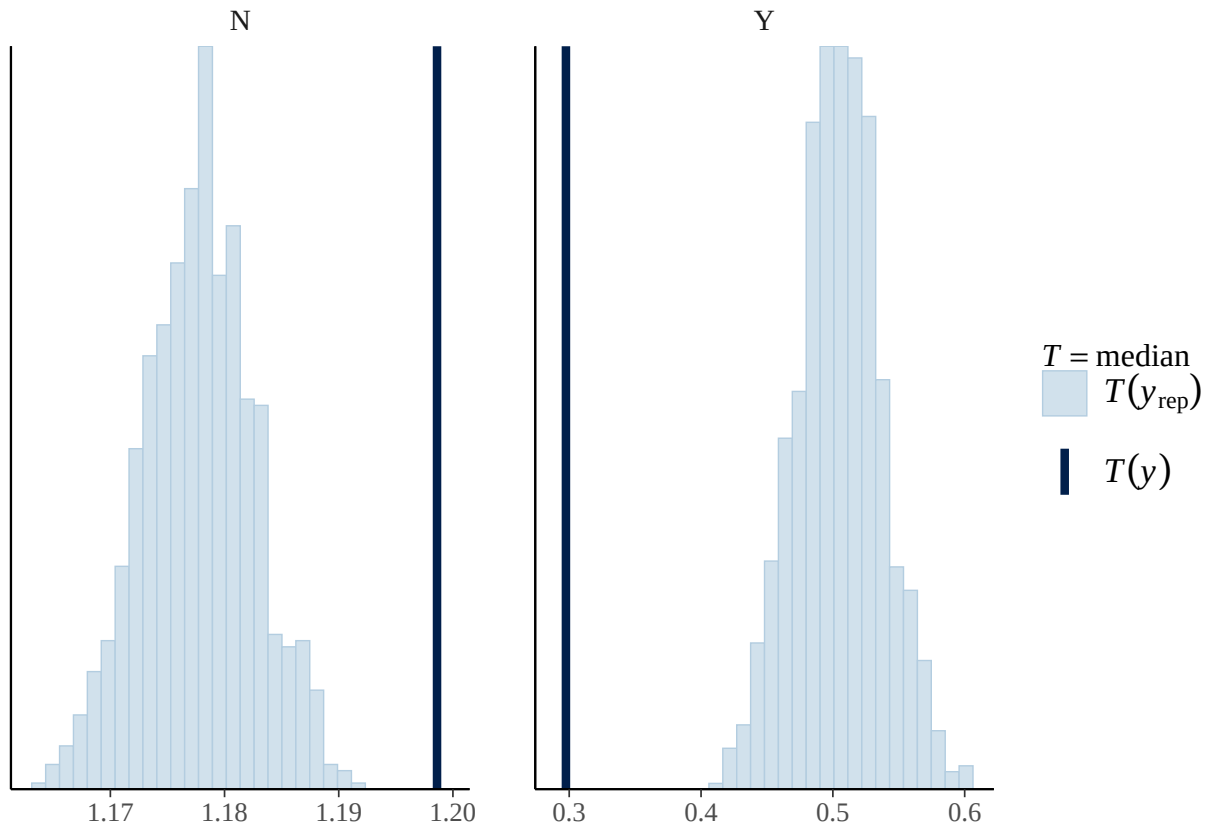
```
y <- ds$log_weight
yrep2 <- extract(mod2)[["log_weight_rep"]]
samp2 <- yrep2[sample(nrow(yrep2), 100),]
df <- data.frame(rbind(samp2, y))
df$index <- 1:nrow(df)
df$type <- c(rep("rep",100), "y")
df <- df %>% gather(key = "key", value = "value", X1:X3842)
df %>% ggplot(aes(x=value, group=index, color = type)) + geom_density() + theme(legend.title=element_bla
```



Test statistics

Medians by prematurity for Model 1:

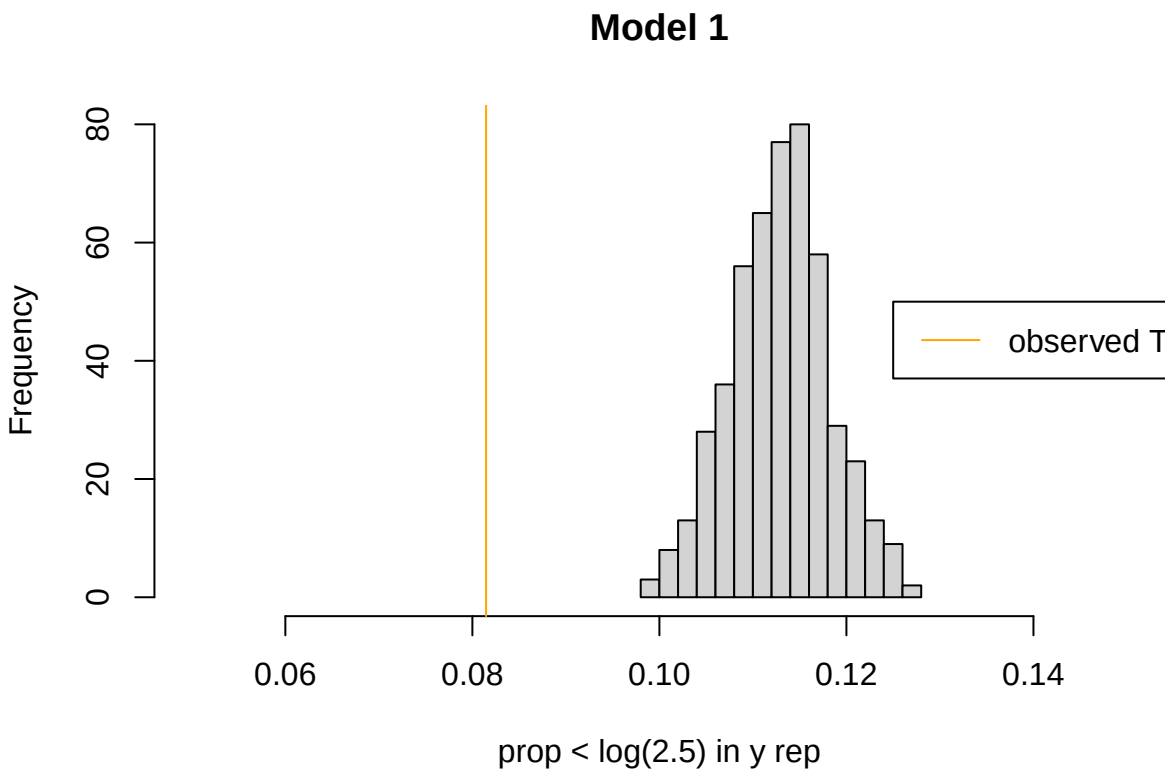
```
ppc_stat_grouped(ds$log_weight, yrep1, group = ds$preterm, stat = 'median')
```



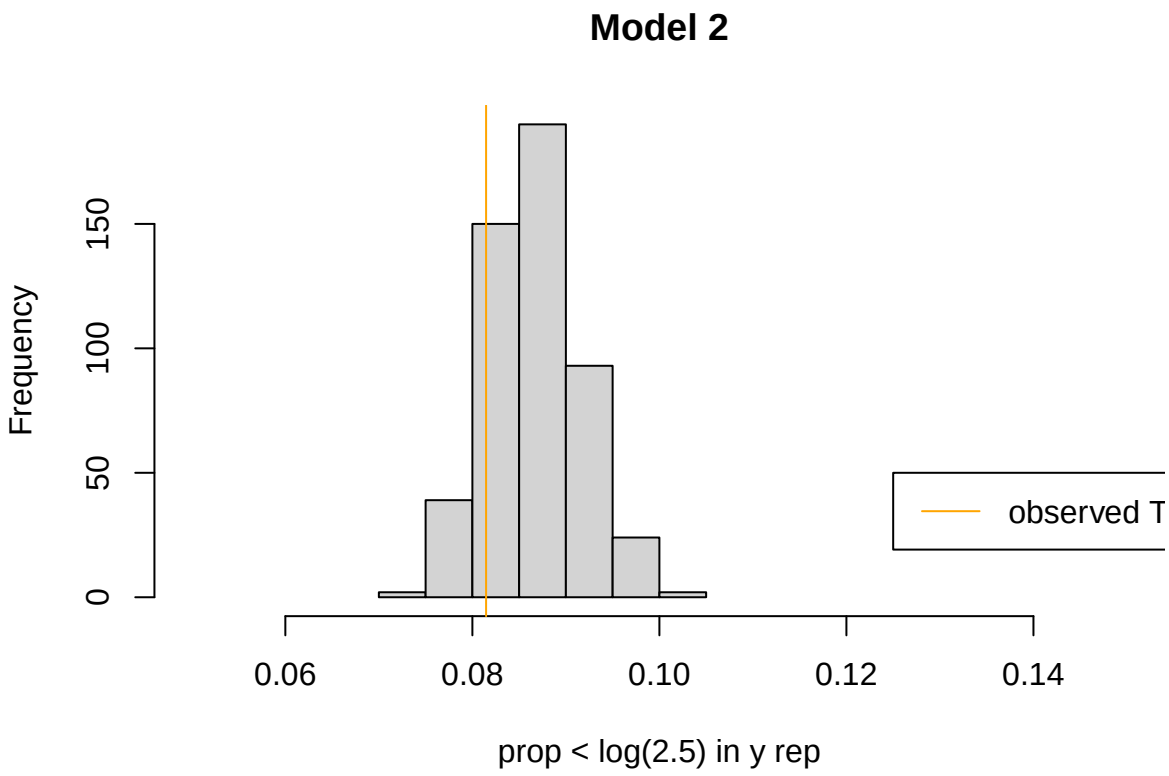
Question 7

We calculate the test statistic for the data, and the posterior predictive samples for both models, and plot the comparisons as below:

```
tstat.data <- sum(y < log(2.5))/ length(y)
tstat.mod1 <- c()
tstat.mod2 <- c()
for (i in 1:500) {
  tstat.mod1 <- c(tstat.mod1, sum(yrep1[i,]<log(2.5))/length(yrep1[i,]))
  tstat.mod2 <- c(tstat.mod2, sum(yrep2[i,]<log(2.5))/length(yrep2[i,]))
}
hist(tstat.mod1, xlim = c(0.05, 0.15), main = "Model 1", xlab = "prop < log(2.5) in y rep")
abline(v = tstat.data, col= "orange")
legend(0.125, 50, legend = "observed T", col = "orange", lty=1)
```




```
hist(tstat.mod2, xlim = c(0.05, 0.15), main = "Model 2", xlab = "prop < log(2.5) in y rep")
abline(v = tstat.data, col = "orange")
legend(0.125, 50, legend = "observed T", col = "orange", lty=1)
```



LOO

Finally let's calculate the LOO elpd for each model and compare. The first step of this is to get the point-wise log likelihood estimates from each model:

```
loo1 <- loo(loglik1, save_psis = TRUE)
loo2 <- loo(loglik2, save_psis = TRUE)
```

```
loo1
```

```
##
## Computed from 1000 by 3842 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   1377.0  72.4
## p_loo         9.8   1.4
## looic     -2754.1 144.8
## ------
## Monte Carlo SE of elpd_loo is 0.1.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

```
loo2
```

```
##
## Computed from 500 by 3842 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   1552.8  70.0
## p_loo        14.8   2.3
## looic     -3105.6 139.9
## ------
## Monte Carlo SE of elpd_loo is 0.2.
##
## All Pareto k estimates are good (k < 0.5).
## See help('pareto-k-diagnostic') for details.
```

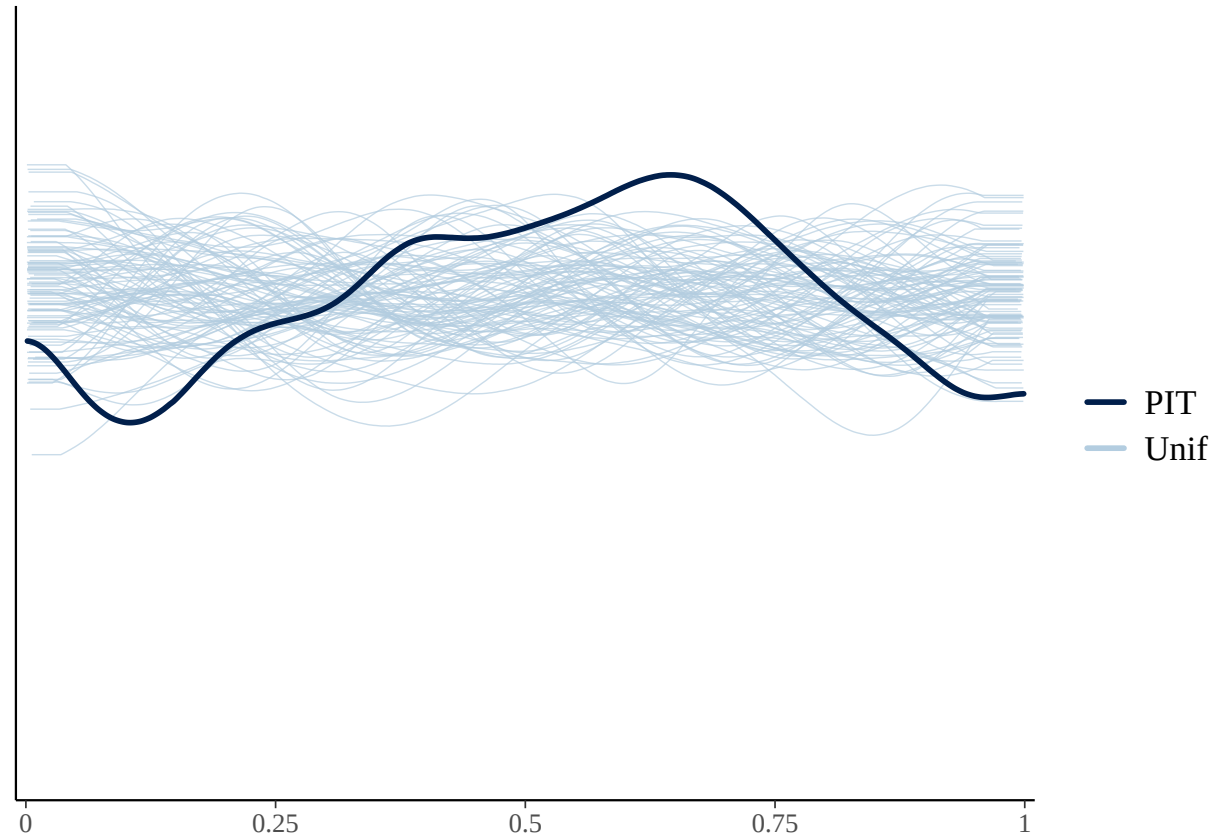
Comparing the two models tells us Model 2 is better:

```
loo_compare(loo1, loo2)
```

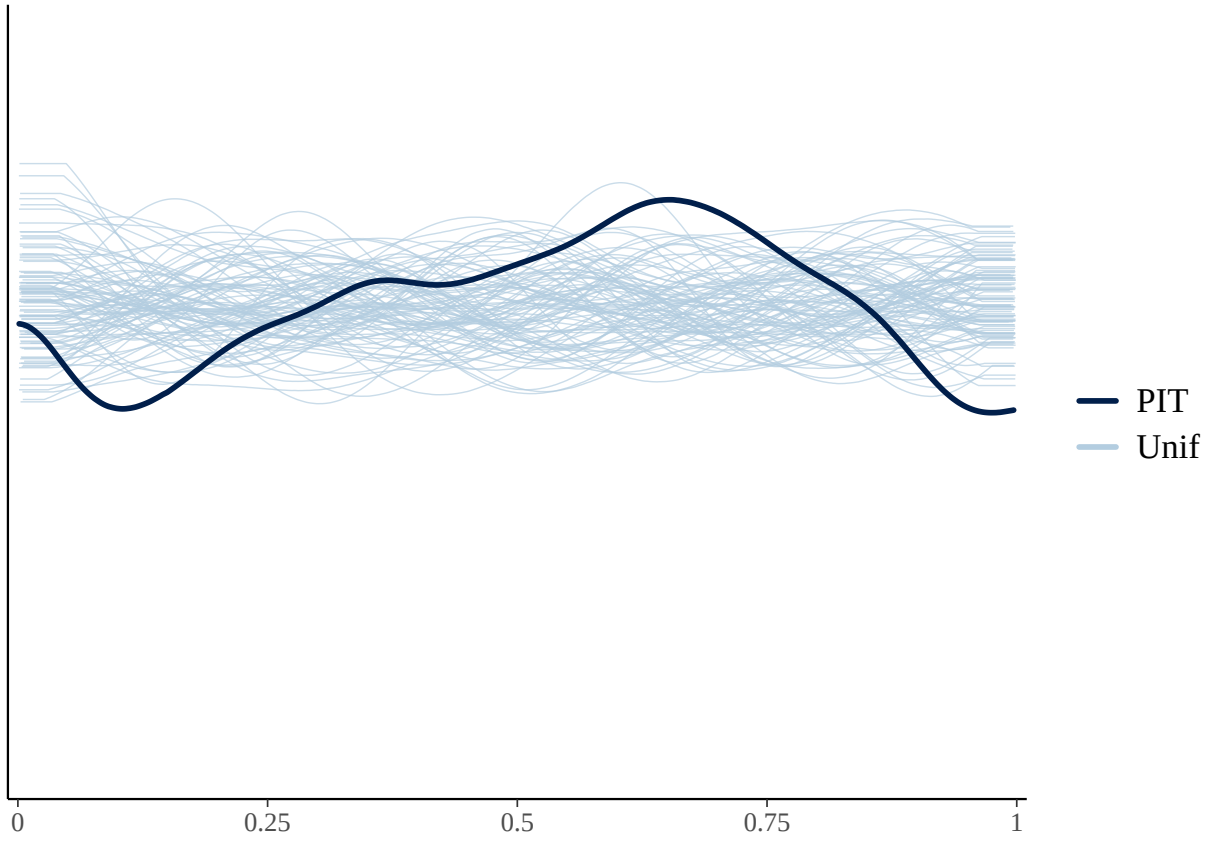
```
##        elpd_diff se_diff
## model2    0.0       0.0
## model1 -175.8      36.2
```

We can also compare the LOO-PIT of each of the models to standard uniforms. The both do pretty well.

```
ppc_loo_pit_overlay(yrep = yrep1, y = y, lw = weights(loo1$psis_object))
```



```
ppc_loo_pit_overlay(yrep = yrep2, y = y, lw = weights(loo2$psis_object))
```



Question 8

I will add the sex variable to model one and then make a comparison with model 2. Results suggest that model 2 performs better.

```
ds$s <- ifelse(ds$sex == "M", 1, 0)
stan_data_3 <- list(N = nrow(ds),
                    log_weight = ds$log_weight,
                    log_gest = ds$log_gest_c,
                    s = ds$s)
```

```
summary(my_mod_3)$summary[c(paste0("beta[", 1:3, "]"), "sigma"),]
```

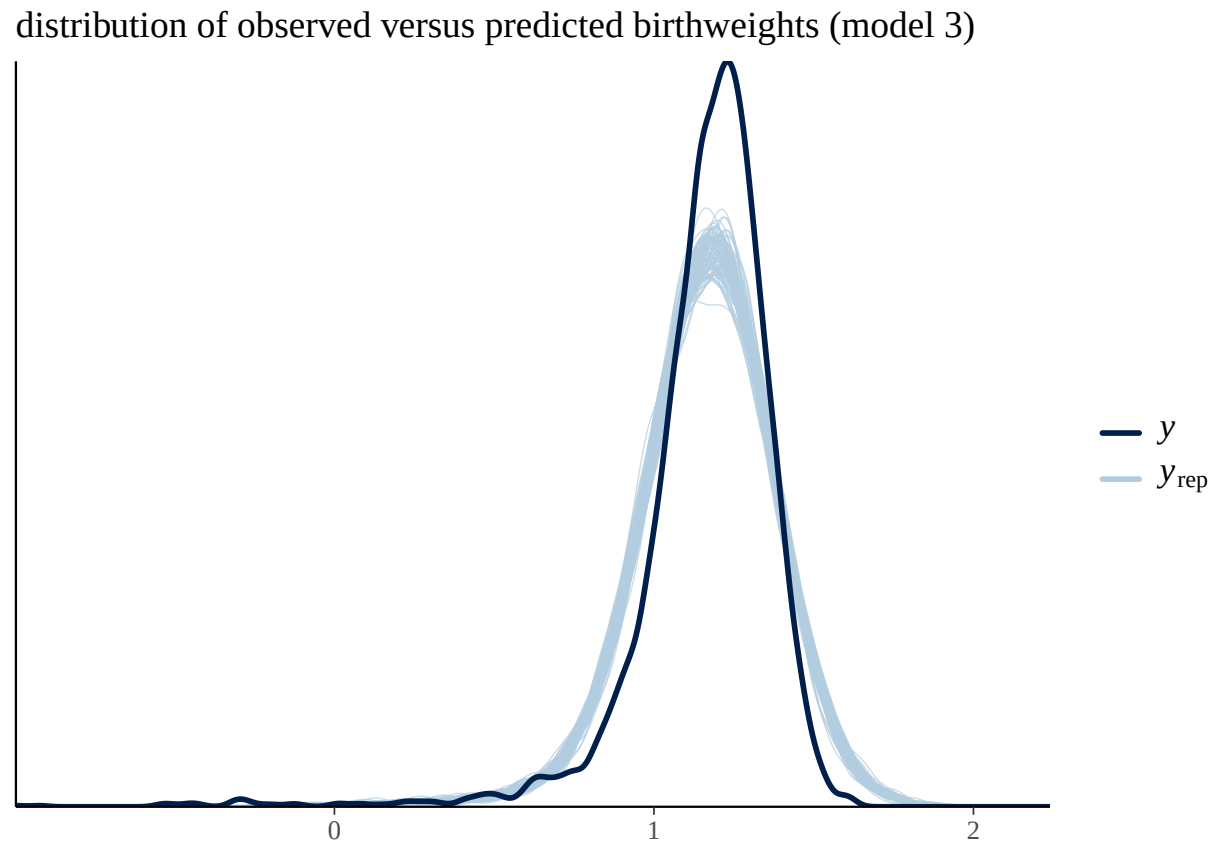
```
##               mean      se_mean          sd       2.5%        25%        50%
## beta[1] 1.14028996 1.493403e-04 0.003754554 1.13285962 1.13779512 1.14018814
## beta[2] 0.14414166 9.454262e-05 0.002773616 0.13860941 0.14230469 0.14404814
## beta[3] 0.04436215 2.183887e-04 0.005392706 0.03392679 0.04069144 0.04438335
## sigma   0.16739711 1.090949e-04 0.002013758 0.16359394 0.16609752 0.16731626
##                75%      97.5%    n_eff      Rhat
## beta[1] 1.14287688 1.14764264 632.0665 0.9998937
## beta[2] 0.14592509 0.15018203 860.6716 1.0007940
```

```
## beta[3] 0.04802888 0.05480015 609.7522 1.0006674
## sigma   0.16868116 0.17155964 340.7265 0.9974656
```

```
yrep3 <- extract(my_mod_3)[["log_weight_rep"]]
```

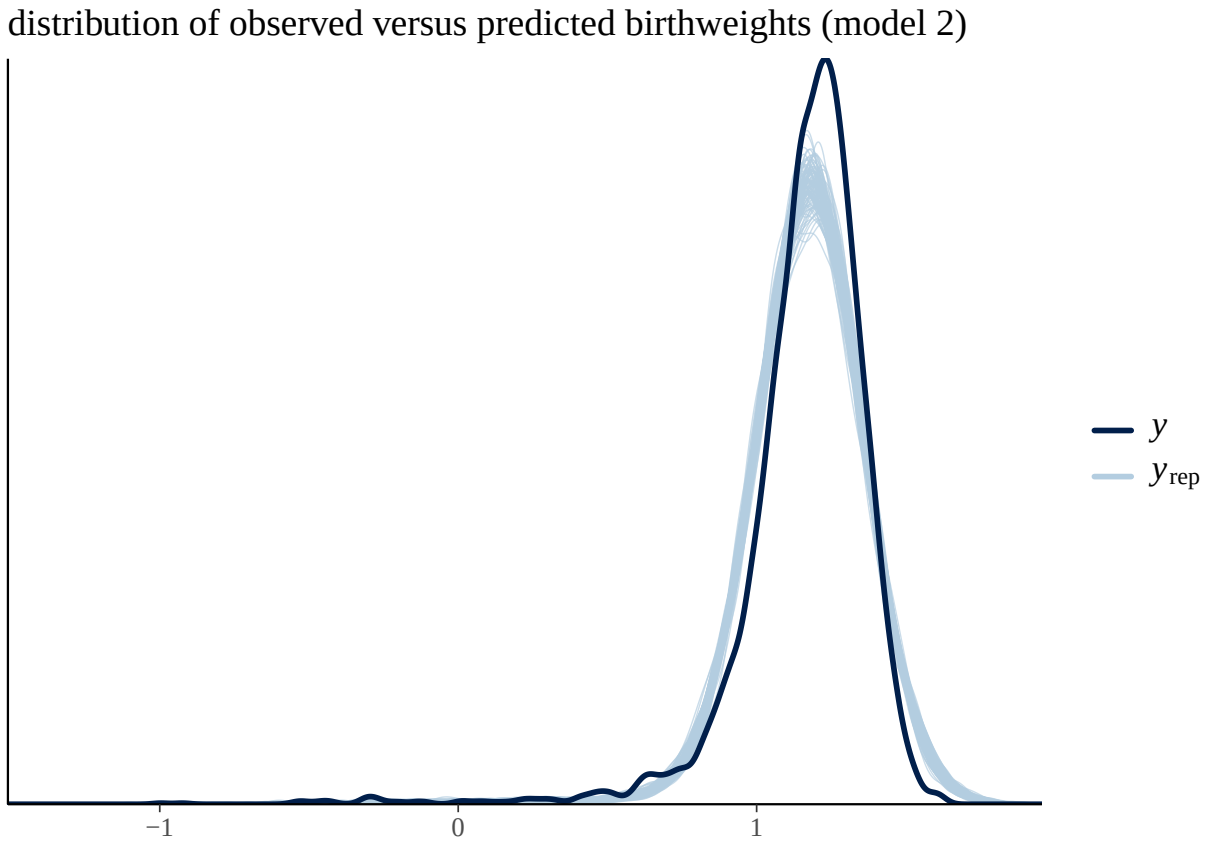
Plot shown below is related to model 3:

```
samp100 <- sample(nrow(yrep3), 100)
ppc_dens_overlay(y, yrep3[samp100, ])  + ggtitle("distribution of observed versus predicted birthweights
```



Let's do for model 2:

```
samp100 <- sample(nrow(yrep2), 100)
ppc_dens_overlay(y, yrep2[samp100, ])  + ggtitle("distribution of observed versus predicted birthweights
```



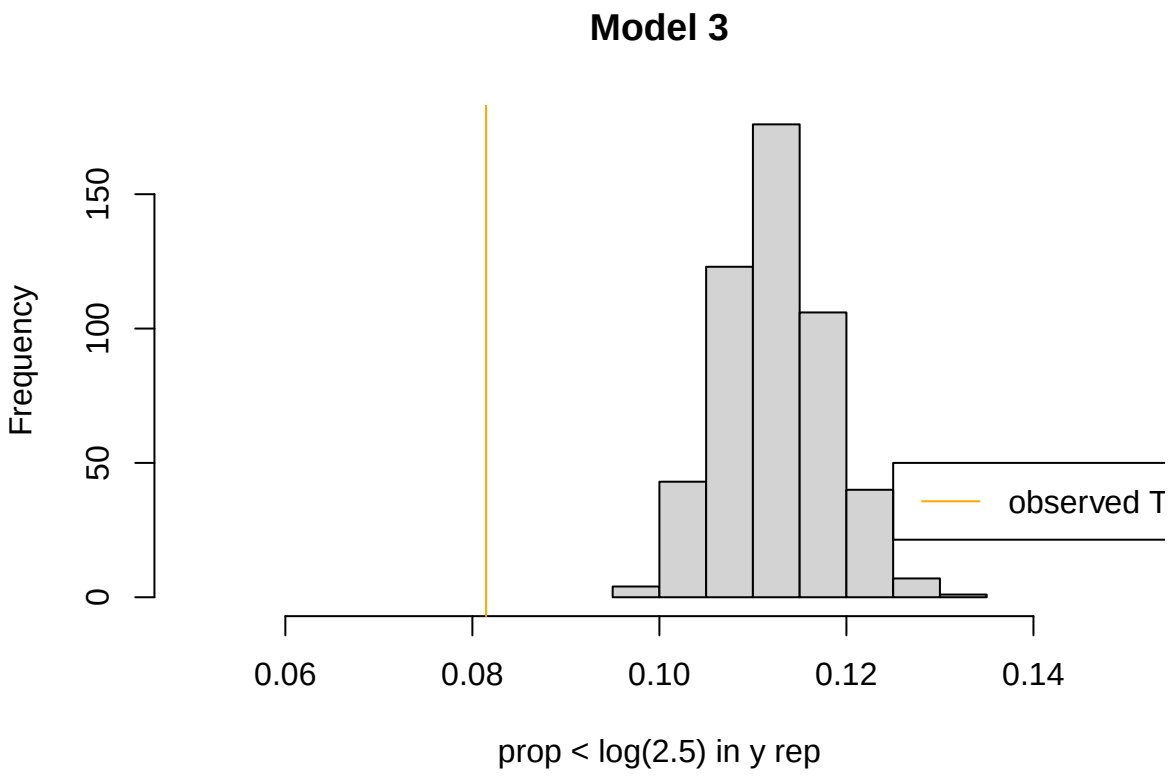
let's do another comparison between model 2 and model 3:

```
loglik3 <- extract(my_mod_3)[["log_lik"]]
loo3 <- loo(loglik3, save_psis = TRUE)
loo_compare(loo2, loo3)
```

```
##        elpd_diff se_diff
## model1    0.0       0.0
## model2 -143.1      36.4
```

Now we carry out a test statistic of the proportion of births under 2.5kg. We calculate the test statistic for the data, and the posterior predictive samples for model 3, and plot the comparison.

```
tstat.data <- sum(y < log(2.5))/ length(y)
tstat.mod3 <- c()
for (i in 1:500) {
  tstat.mod3 <- c(tstat.mod3, sum(yrep3[i,]<log(2.5))/length(yrep3[i,]))
}
hist(tstat.mod3, xlim = c(0.05, 0.15), main = "Model 3", xlab = "prop < log(2.5) in y rep")
abline(v = tstat.data, col= "orange")
legend(0.125, 50, legend = "observed T", col = "orange", lty=1)
```


Model 3
Frequency
0
50
100
150
0.06
0.08
0.10
0.12
0.14
observed T
prop < log(2.5) in y rep