STA2201 Lab #6

Timothy Jordan Regis

23/02/2023

Introduction

This lab will be looking at trying to replicate some of the visualizations in the lecture notes, involving prior and posterior predictive checks, and LOO model comparisons.

The dataset is a 0.1% of all births in the US in 2017. I've pulled out a few different variables, but as in the lecture, we'll just focus on birth weight and gestational age.

The data

Read it in, along with all our packages.

```
## # 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
```

Brief overview of variables:

- `mager` mum's age
- `mracehisp` mum's race/ethnicity see here for codes: https://data.nber.org/natality/2017/natl2017.pdf page 15
- `meduc` mum's education see here for codes: https://data.nber.org/natality/2017/natl2017.pdf page 16
- `bmi` mum's bmi
- `sex` baby's sex
- `combgest` gestational age in weeks
- `dbwt` birth weight in kg
- `ilive` alive at time of report y/n/ unsure

I'm going to rename some variables, remove any observations with missing gestational age or birth weight, restrict just to babies that were alive, and make a preterm variable.

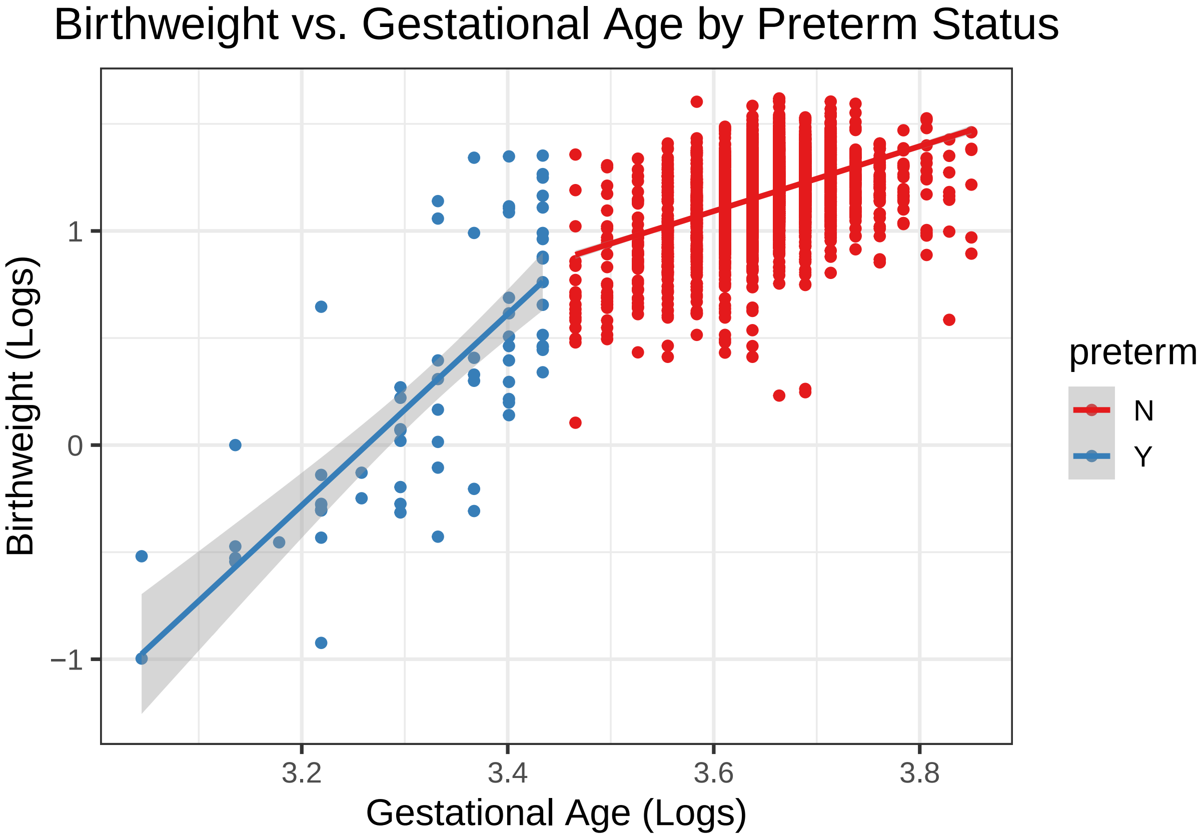
Question 1

Use plots or tables to show three interesting observations about the data. Remember:

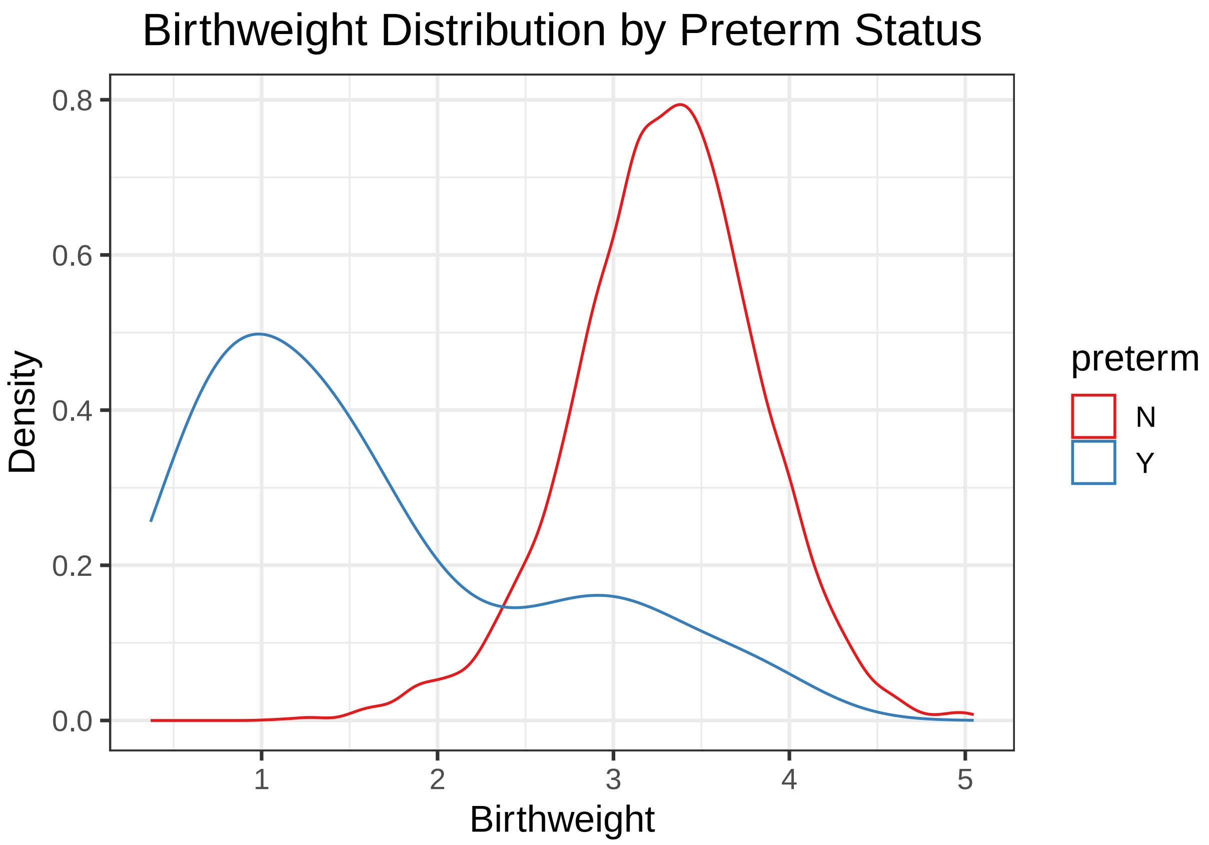
- Explain what your graph/ tables show
- Choose a graph type that's appropriate to the data type
- If you use `geom_smooth`, please also plot the underlying data

Feel free to replicate one of the scatter plots in the lectures as one of the interesting observations, as those form the basis of our models.

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

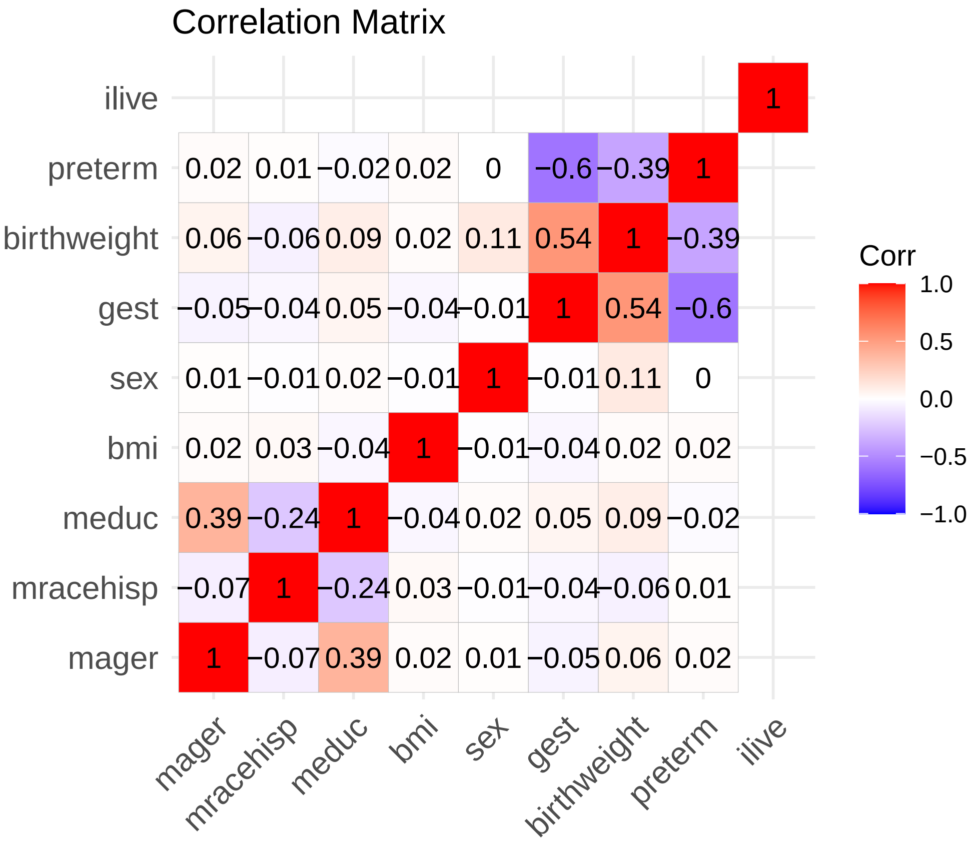


In our first plot, we see the distribution of birthweights by gestational age, both on the log scale, with a colouring by the preterm status. We have also added linear smoothings to each of these groups to better identify the existing trends. As we can see, there is a fairly clear positive relationship between gestational age and birth weight, suggesting longer gestation peiods leading to heavier babies, possibly due to increased nutrition in the womb. Furthermore, there is a difference of effects based on the preterm status, where we see preterm babies having a much stronger positive relationship than babies born after the preterm period.



Our second plot displays the distribution of birthweights in the data, grouped again by the preterm status. Here, we can observe that there is a significant difference between the birthweights of preterm and non-preterm babies, where preterm births have a high density at lower weights, peaking at 1, whereras non-preterm babies have a majority of their density at weights above 2, peaking around 3.5. Thus, we can see that not only do other factors have different effects based on preterm status, but that the birthweights themselves are also potentially influenced by the preterm status. This result is faairly sensible, however, as we acknowledge the multitude of difficulties presented in preterm births, and the much greater frequency of complications that arise which can impact the weight of the child. Moreover, preterm babies spend less time in the womb and thus have less time to grow in the final weeks of pregnancy which are often considered to be the most valuable to a baby's development.

```
## Warning in cor(ds %>% mutate(sex = as.numeric(as.factor(sex)), preterm =
## as.numeric(as.factor(preterm)), : the standard deviation is zero
```



Our final plot displays the correlation matrix of our data, which can aid in identifying any interesting links between the raw data presented. As we can see firstly, the column tracking the survival of a baby has no correlation due to our filtering out all deaths. We can also notice strong correlations between a few variables. Firstly, preterm status has a strong negative correlation with both birthweight and gestational period, which agree with the observations we saw in the previous two plots. Furthermore, birthweight has a high positive correlation with gestational age, again agreeing with our previous plots. The education of a mother also has strong correlations with their age and race. With age, we see a positive relationship, which suggests that more educated mothers are often correlated with older mothers. Lastly, with race, this becomes somewhat uninterpretable, as unlike education, which has ordinal characteristics with higher numbers equalling higher education levels, race is purely categorical, so we cannot directly interpret how these variables are related exactly. The remaining variables showed no significant correlations above a magnitude of 0.2.

The model

As in lecture, we will look at two candidate models

Model 1 has log birth weight as a function of log gestational age

$$\log(y_i) \sim N(\beta_1 + \beta_2 \log(x_i), \sigma^2)$$

Model 2 has an interaction term between gestation and prematurity

$$\log(y_i) \sim N(\beta_1 + \beta_2 \log(x_i) + \beta_3 z_i + \beta_4 \log(x_i) z_i, \sigma^2)$$

- y_i is weight in kg

- x_i is gestational age in weeks, CENTERED AND STANDARDIZED
- z_i is preterm (0 or 1, if gestational age is less than 32 weeks)

Prior predictive checks

Let's put some weakly informative priors on all parameters i.e. for the βs

$$\beta \sim N(0, 1)$$

and for σ

$$\sigma \sim N^{+}(0, 1)$$

where the plus means positive values only i.e. Half Normal.

Let's check to see what the resulting distribution of birth weights look like given Model 1 and the priors specified above, assuming we had no data on birth weight (but observations of gestational age).

Question 2

For Model 1, simulate values of βs and σ based on the priors above. Do 1000 simulations. Use these values to simulate (log) birth weights from the likelihood specified in Model 1, based on the set of observed gestational weights. **Remember the gestational weights should be centered and standardized**.

- Plot the resulting distribution of simulated (log) birth weights.
- Plot ten simulations of (log) birthweights against gestational age.

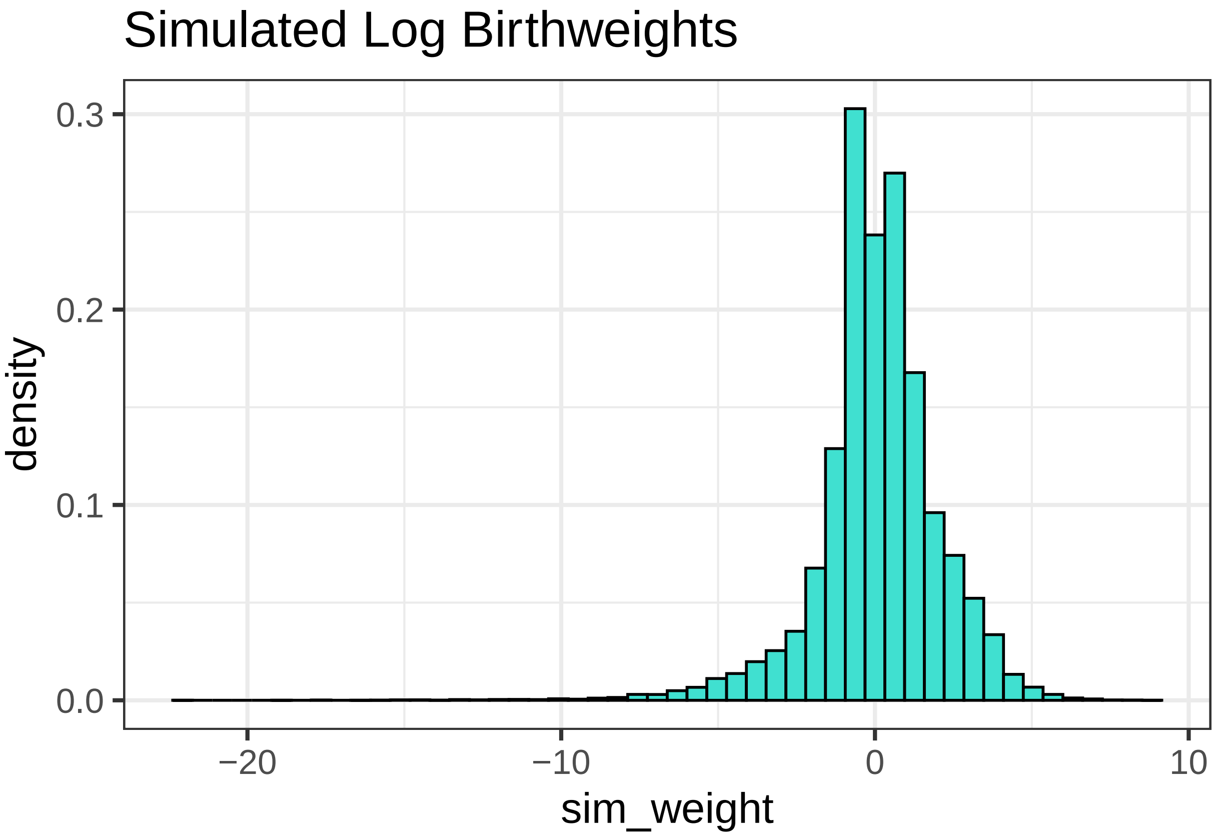
Run the model

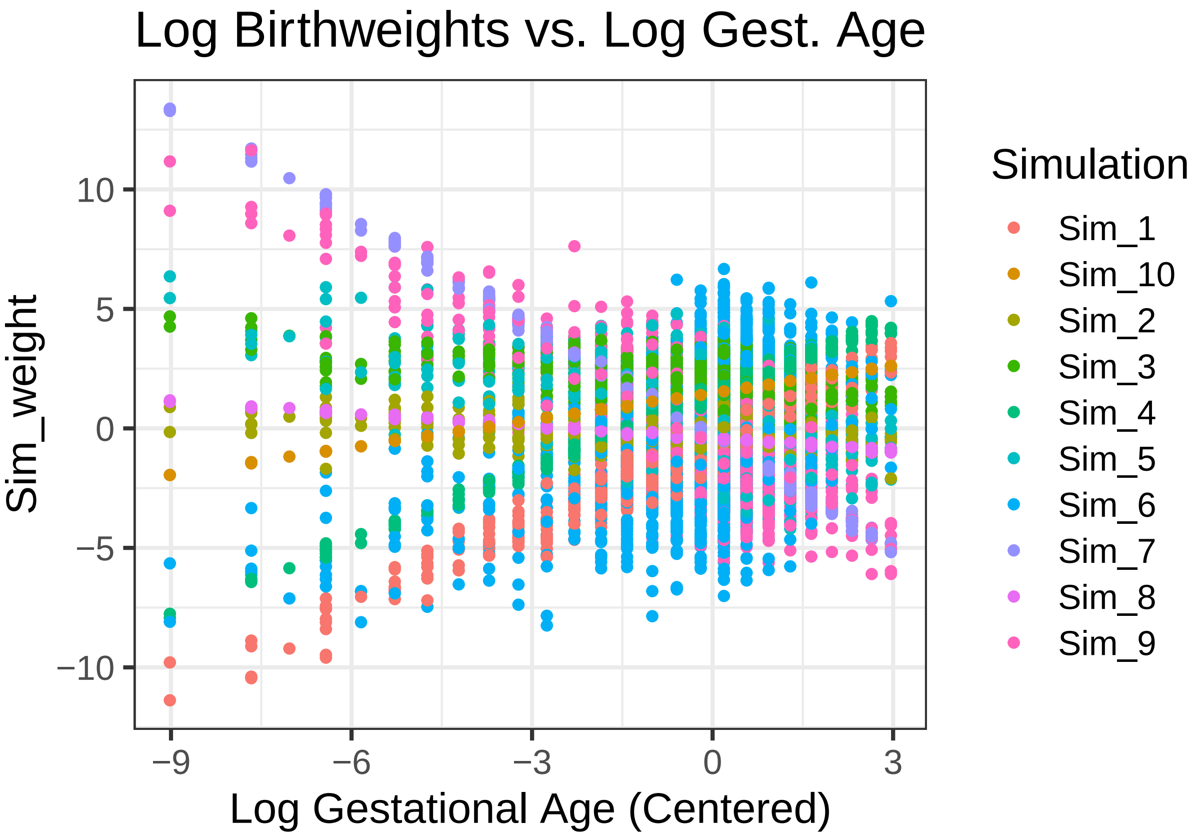
Now we're going to run Model 1 in Stan. The stan code is in the `code/models` folder.

First, get our data into right form for input into stan.

Now fit the model

```
##              mean      se_mean          sd      2.5%       25%       50%
## beta[1] 1.1626250 7.634607e-05 0.002583881 1.1575321 1.1609497 1.1626383
## beta[2] 0.1436183 8.105504e-05 0.002791943 0.1380281 0.1417563 0.1436199
## sigma   0.1689127 1.051837e-04 0.001979909 0.1650908 0.1676042 0.1688619
##               75%     97.5%     n_eff      Rhat
## beta[1] 1.1643919 1.1677313 1145.4383 0.9970543
## beta[2] 0.1455075 0.1489575 1186.4598 0.9984953
## sigma   0.1701148 0.1728405  354.3181 1.0046933
```


Simulated Log Birthweights
density
0.3
0.2
0.1
0.0
−20
−10
0
10
sim_weight



Both plots display a centering of data and observations around 0 for weight which was our initial mean value. No significant issues observed.

Question 3

Based on model 3, give an estimate of the expected birthweight of a baby who was born at a gestational age of 37 weeks.

```
##               mean      se_mean          sd      2.5%       25%       50%
## beta[1] 1.1626250 7.634607e-05 0.002583881 1.1575321 1.1609497 1.1626383
## beta[2] 0.1436183 8.105504e-05 0.002791943 0.1380281 0.1417563 0.1436199
## sigma   0.1689127 1.051837e-04 0.001979909 0.1650908 0.1676042 0.1688619
##               75%     97.5%     n_eff      Rhat
## beta[1] 1.1643919 1.1677313 1145.4383 0.9970543
## beta[2] 0.1455075 0.1489575 1186.4598 0.9984953
## sigma   0.1701148 0.1728405  354.3181 1.0046933
```

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

```
## [1] 2.93654
```

Thus, we see that the expected birthweight of a baby born at 37 weeks is approximately 2.94

Question 4

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

Now fit the model

```
##              mean      se_mean          sd       2.5%        25%       50%
## beta[1] 1.1696059 6.927910e-05 0.002550935 1.16442849 1.16783369 1.1695717
## beta[2] 0.5559712 4.014392e-03 0.069945640 0.41296530 0.51056374 0.5563826
## beta[3] 0.1019253 1.224681e-04 0.003724323 0.09515199 0.09943343 0.1017972
## beta[4] 0.1970114 8.036017e-04 0.014327959 0.16773657 0.18796380 0.1971632
## sigma   0.1612909 8.908049e-05 0.001842882 0.15745754 0.16009865 0.1613438
##               75%     97.5%     n_eff      Rhat
## beta[1] 1.1713054 1.1746635 1355.7962 0.9984824
## beta[2] 0.6065553 0.6850575  303.5860 1.0052147
## beta[3] 0.1043354 0.1093189  924.8010 0.9990344
## beta[4] 0.2068355 0.2242119  317.8974 1.0074509
## sigma   0.1624883 0.1648739  427.9866 1.0112045
```

Question 5

For reference I have uploaded some model 2 results. Check your results are similar.

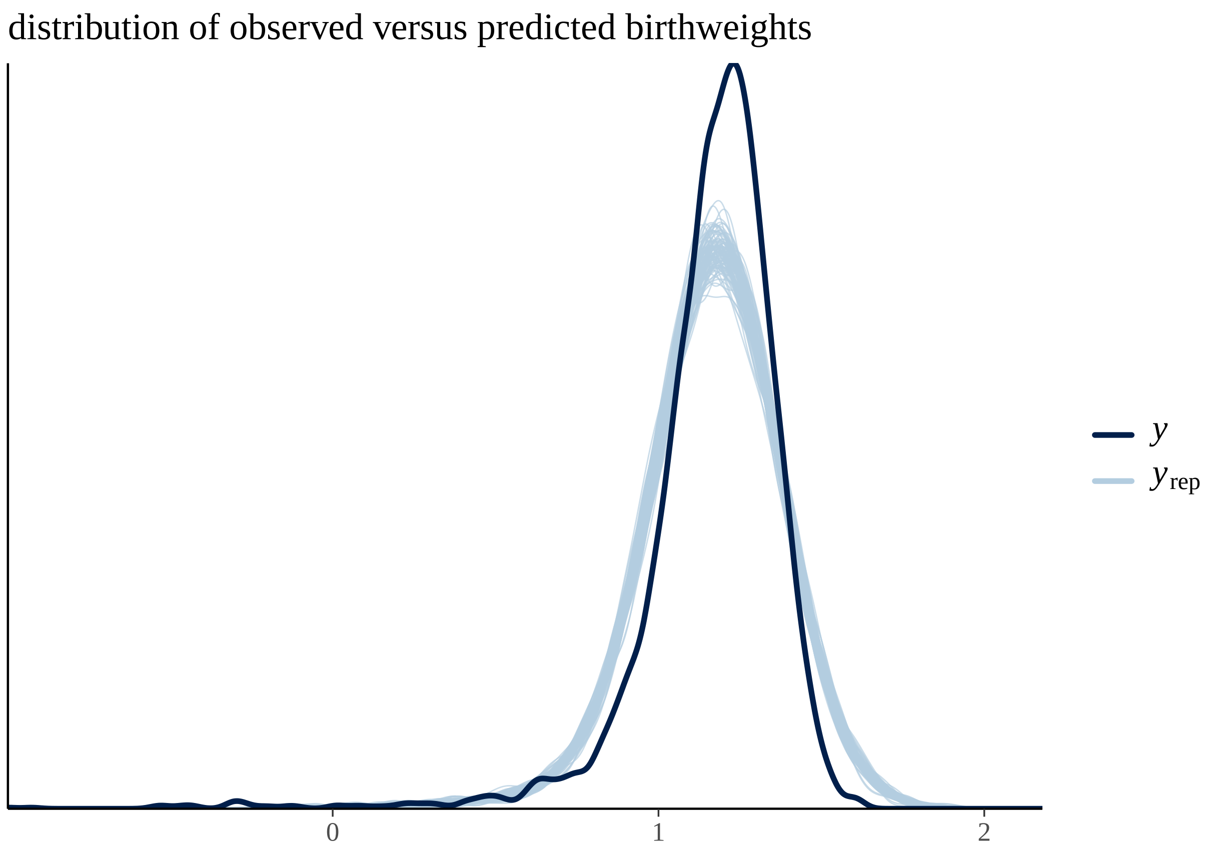
```
##              mean      se_mean          sd       2.5%        25%       50%
## beta[1] 1.1697241 1.385590e-04 0.002742186 1.16453578 1.16767109 1.1699278
## beta[2] 0.5563133 5.835253e-03 0.058054991 0.43745504 0.51708255 0.5561553
## beta[3] 0.1020960 1.481816e-04 0.003669476 0.09459462 0.09997153 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
```

Coefficients are quite similar with minor differences in sd and se.

PPCs

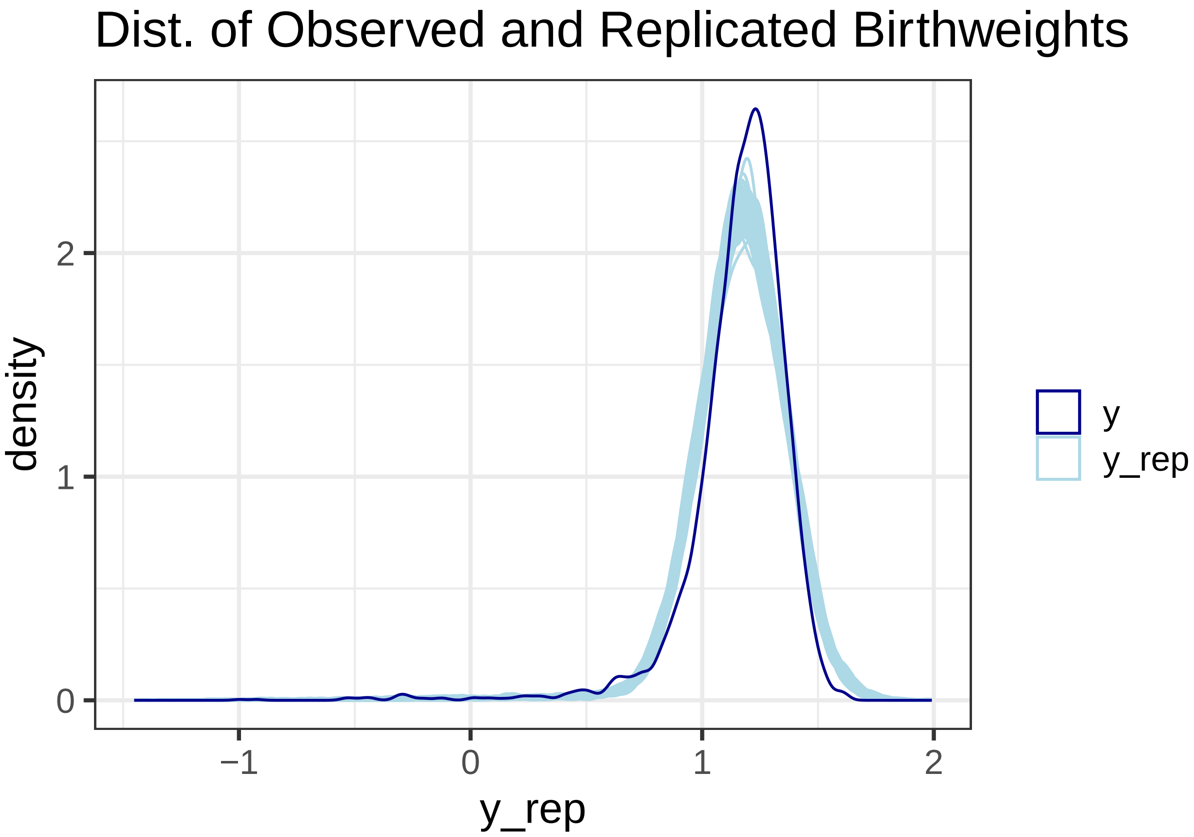
Now we've run two candidate models let's do some posterior predictive checks. The `bayesplot` package has a lot of inbuilt graphing functions to do this. For example, let's plot the distribution of our data (y) against 100 different datasets drawn from the posterior predictive distribution:

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



Question 6

Make a similar plot to the one above but for model 2, and **not** using the bayes plot in built function (i.e. do it yourself just with `geom_density`)

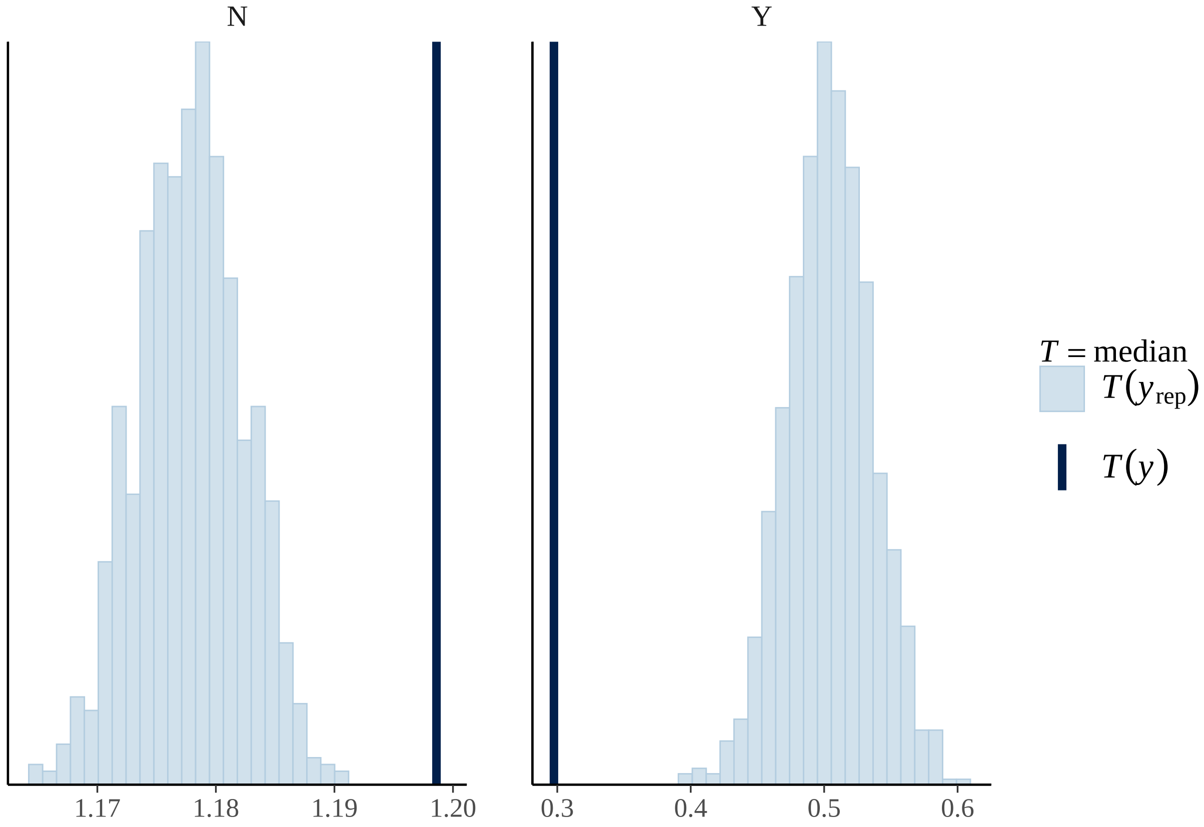


Test statistics

We can also look at some summary statistics in the PPD versus the data, again either using `bayesplot` – the function of interest is `ppc_stat` or `ppc_stat_grouped` – or just doing it ourselves using ggplot.

E.g. medians by prematurity for Model 1

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

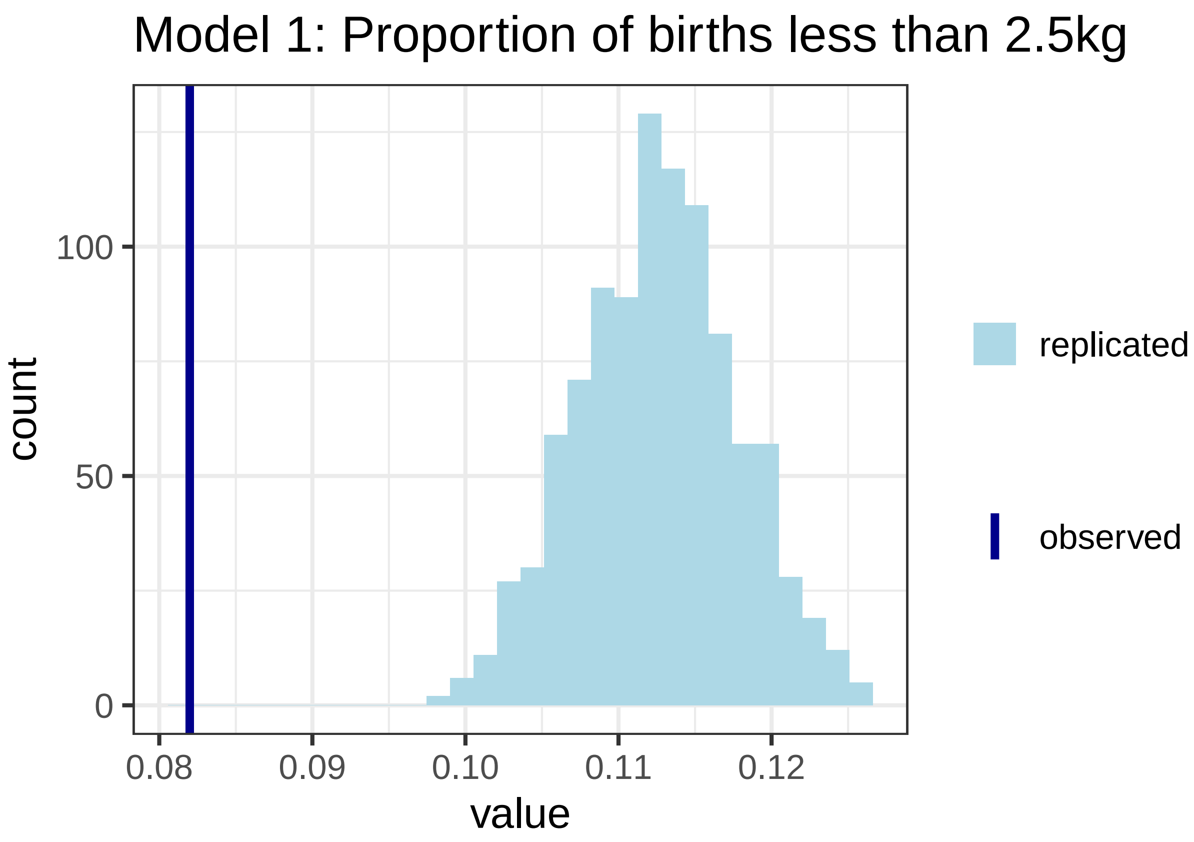


Question 7

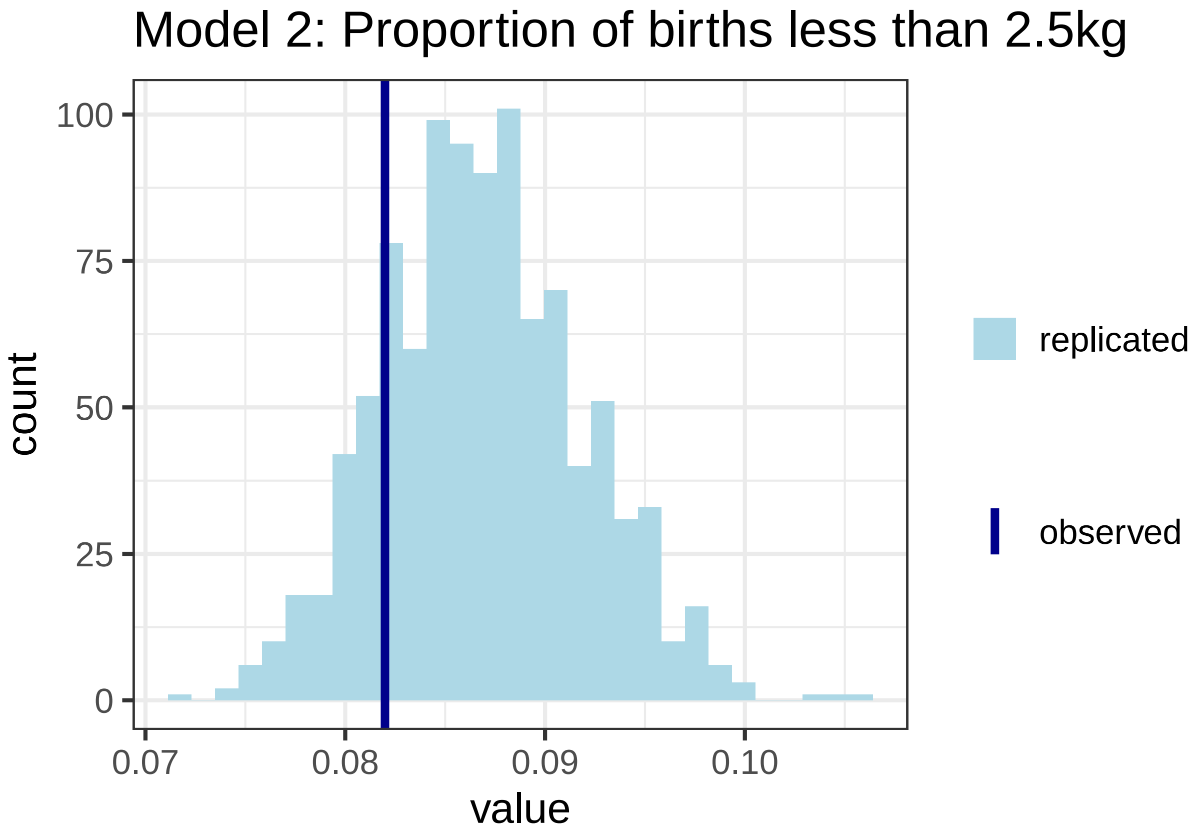
Use a test statistic of the proportion of births under 2.5kg. Calculate the test statistic for the data, and the posterior predictive samples for both models, and plot the comparison (one plot per model).

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



As we can see, Model 2 has a higher proportion of its values around the observed value, suggesting an improvement

Question 8

Based on the original dataset, choose one (or more) additional covariates to add to the linear regression model. Run the model in Stan, and compare with Model 2 above on at least 2 posterior predictive checks.

For our model we have chosen to add in the mother's age, on the log scale, and we first standardize this value

```r
# Given Model
summary(mod2)$summary[c("beta[1]", "beta[2]", "beta[3]", "beta[4]", "sigma"),]
```

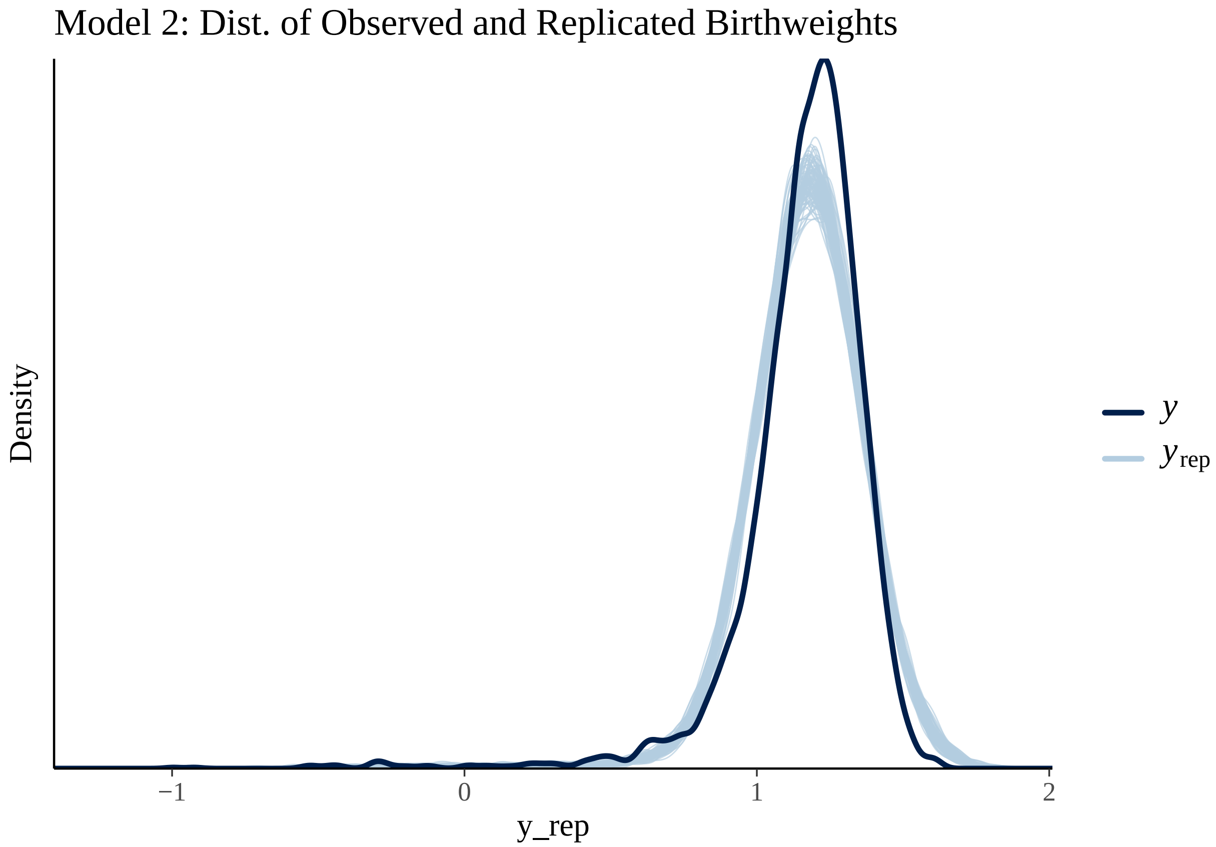
```
##              mean      se_mean          sd       2.5%        25%       50%
## beta[1] 1.1697241 1.385590e-04 0.002742186 1.16453578 1.16767109 1.1699278
## beta[2] 0.5563133 5.835253e-03 0.058054991 0.43745504 0.51708255 0.5561553
## beta[3] 0.1020960 1.481816e-04 0.003669476 0.09459462 0.09997153 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(mod_2)$summary[c("beta[1]", "beta[2]", "beta[3]", "beta[4]", "sigma"),]
```

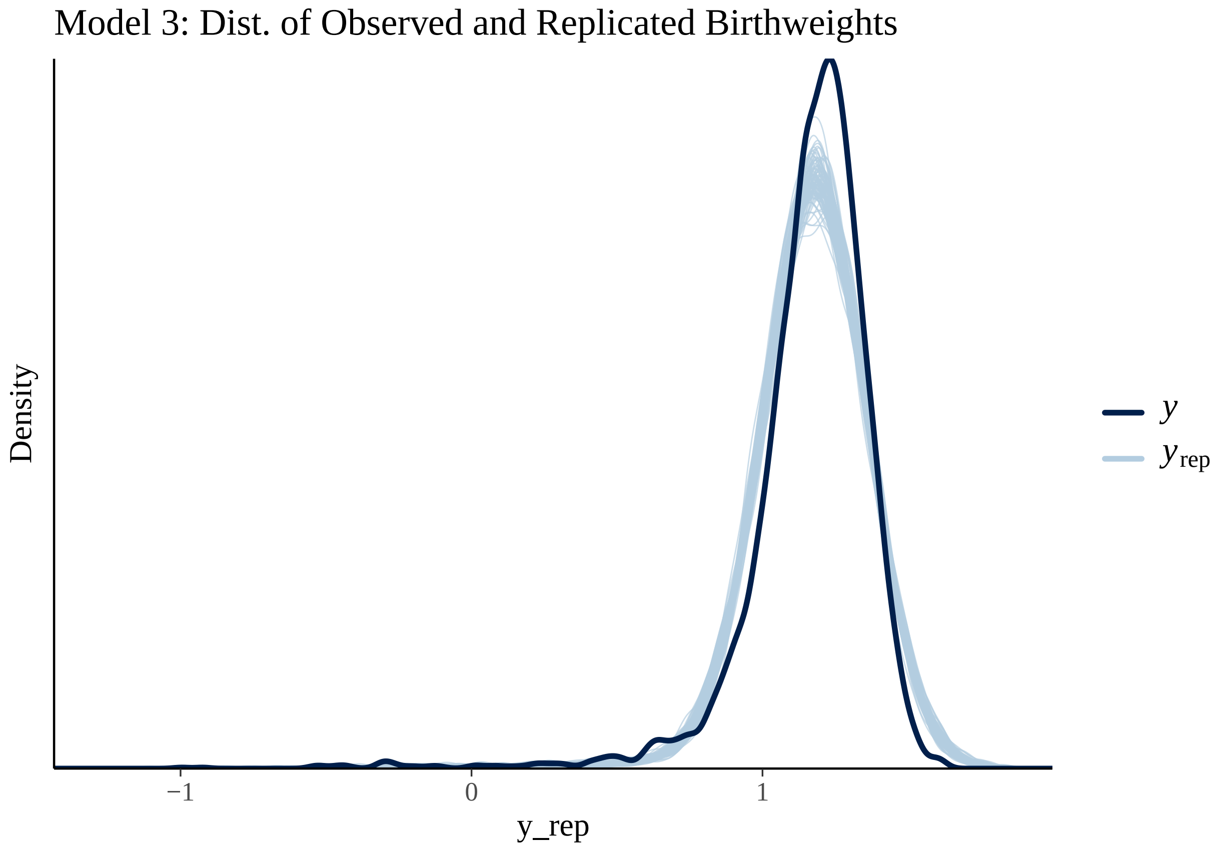
```
##              mean      se_mean          sd       2.5%        25%       50%
## beta[1] 1.1696059 6.927910e-05 0.002550935 1.16442849 1.16783369 1.1695717
## beta[2] 0.5559712 4.014392e-03 0.069945640 0.41296530 0.51056374 0.5563826
## beta[3] 0.1019253 1.224681e-04 0.003724323 0.09515199 0.09943343 0.1017972
## beta[4] 0.1970114 8.036017e-04 0.014327959 0.16773657 0.18796380 0.1971632
## sigma   0.1612909 8.908049e-05 0.001842882 0.15745754 0.16009865 0.1613438
##               75%     97.5%     n_eff      Rhat
## beta[1] 1.1713054 1.1746635 1355.7962 0.9984824
## beta[2] 0.6065553 0.6850575  303.5860 1.0052147
## beta[3] 0.1043354 0.1093189  924.8010 0.9990344
## beta[4] 0.2068355 0.2242119  317.8974 1.0074509
## sigma   0.1624883 0.1648739  427.9866 1.0112045
```

```r
summary(mod_3)$summary[c("beta[1]", "beta[2]", "beta[3]", "beta[4]", "beta[5]", "sigma"),]
```

```
##               mean      se_mean          sd       2.5%        25%        50%
## beta[1] 1.16960052 7.256712e-05 0.002595540 1.16466267 1.16783051 1.16962576
## beta[2] 0.55970688 4.101224e-03 0.067294419 0.42706107 0.51568040 0.56050700
## beta[3] 0.10258601 1.118174e-04 0.003362266 0.09572013 0.10032792 0.10261581
## beta[4] 0.19740761 8.230791e-04 0.013620868 0.17022763 0.18821312 0.19740915
## beta[5] 0.01512897 7.192421e-05 0.002562922 0.01025135 0.01335834 0.01514334
## sigma   0.16058147 8.128039e-05 0.001819348 0.15703120 0.15940689 0.16050725
##                75%      97.5%     n_eff      Rhat
## beta[1] 1.17136418 1.17452890 1279.3096 0.9975628
## beta[2] 0.60168672 0.69493026  269.2347 1.0023771
## beta[3] 0.10485540 0.10931755  904.1607 0.9983826
## beta[4] 0.20632396 0.22536467  273.8586 1.0021816
## beta[5] 0.01688166 0.02043173 1269.7566 0.9993463
## sigma   0.16171749 0.16441456  501.0258 1.0007859
```



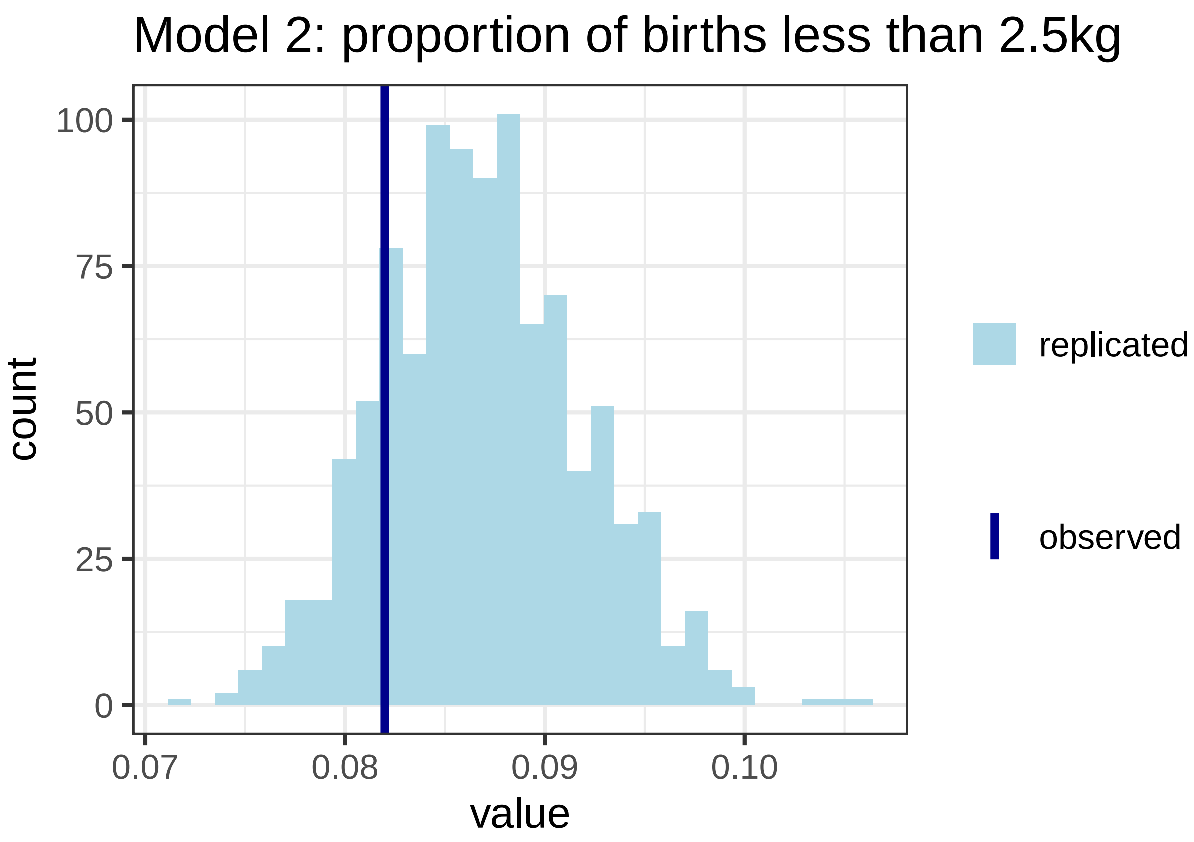
Model 2: Dist. of Observed and Replicated Birthweights



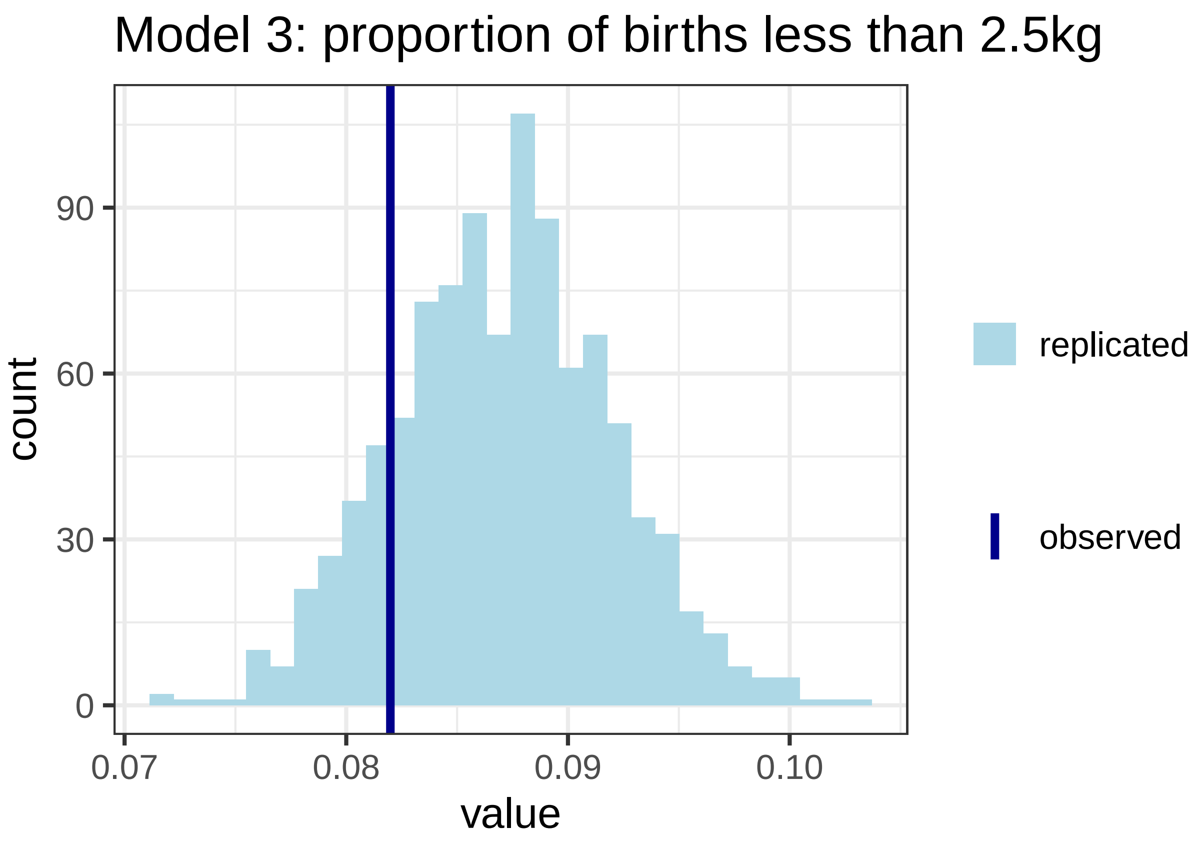
As we can see, there is very little difference between the two models, with both fitting quite close to the desired curve.

We can also observe the test statistic plots of each model.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



Again, however, we still see quite similar results with no clear winner.

```
## Warning: Relative effective sample sizes ('r_eff' argument) not specified.
## For models fit with MCMC, the reported PSIS effective sample sizes and
## MCSE estimates will be over-optimistic.

## Warning: Relative effective sample sizes ('r_eff' argument) not specified.
## For models fit with MCMC, the reported PSIS effective sample sizes and
## MCSE estimates will be over-optimistic.
```

```
##
## Computed from 1000 by 3842 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   1552.2  69.8
## p_loo        16.0   2.5
## looic     -3104.4 139.6
## ------
## 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.
```

```
##
## Computed from 1000 by 3842 log-likelihood matrix
##
```

```
##          Estimate    SE
## elpd_loo   1568.4  70.5
## p_loo        16.7   2.4
## looic     -3136.8 141.1
## ------
## 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.
```

```
##        elpd_diff se_diff
## model2   0.0       0.0
## model1 -16.2       5.7
```

Finally, to numerically assess the results, we can compare the two models via `loo_compare` which tells us that the new model, mod_3, is a little bit better than the earlier mod_2.

Note:

Helper code for this lab taken from: https://www.monicaalexander.com/posts/2020-28-02-bayes_viz/