Week 5: Bayesian linear regression and introduction to Stan

Hainan Xu

15/02/24

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

```
kidiq <- readRDS("~/Documents/2024/STA2201-Methods-of-Applied-Statistics/labs/kidiq.RDS")
```

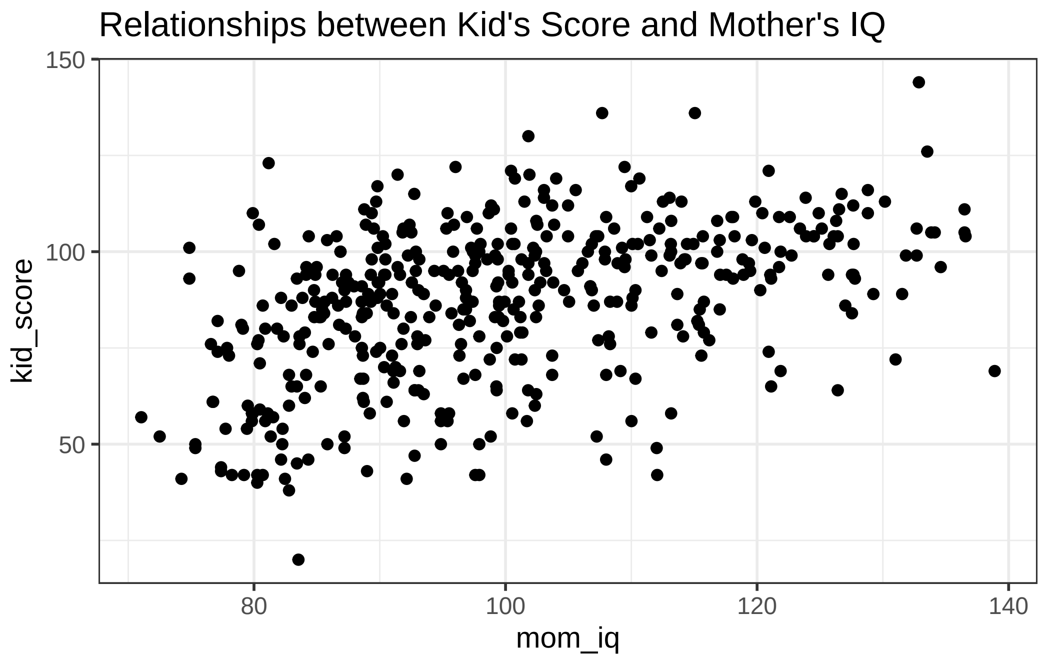
Question 1

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

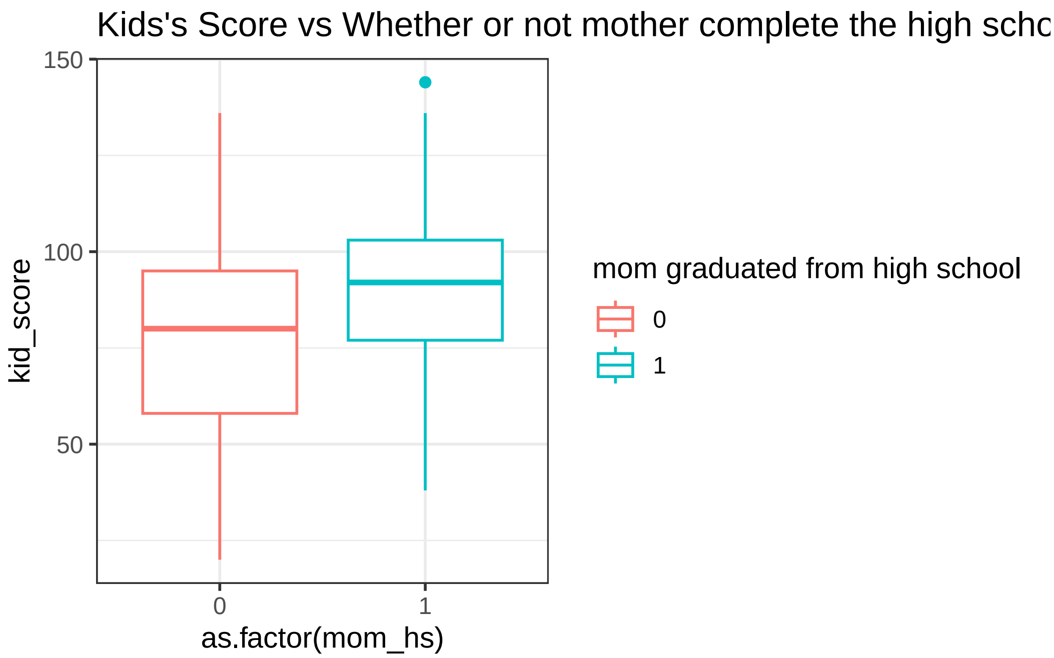
The scatterplot below shows the relationships between kid's score and mom's IQ, we observe that the kid's score and mom's IQ seems to have a positive relationship.

```
#relation ship between kids' score and mother's IQ
ggplot(data=kidiq)+geom_point(aes(x=mom_iq,y=kid_score),fill="lightblue")+theme_bw()+ggtit
```



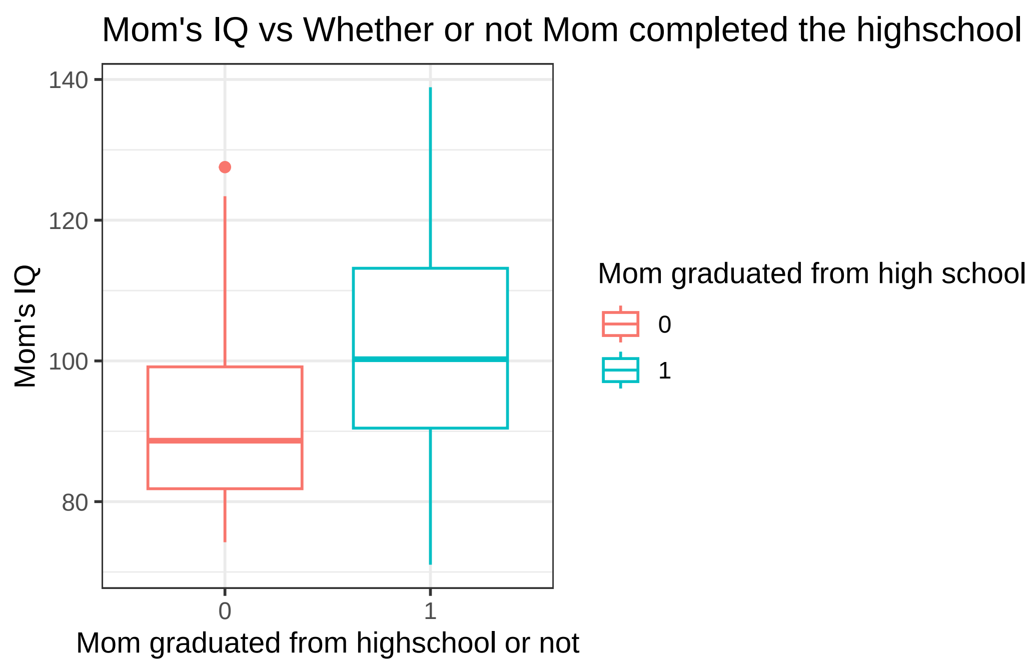
The boxplot below shows the distribution of kids kid's score catergorized by whether or not the mom is graduated from highschool. We observe that the mean of kid's IQ whose graduated from highschool is higher than that of not graduated from highschool.

```
# relationships between kids'score and whether or not mother complete the high school
ggplot(data=kidiq)+geom_boxplot(aes(x=as.factor(mom_hs),y=kid_score,color=as.factor(mom_hs
```



The boxplot below shows the distribution of mom's iq catergrized by whether or not the shei s graduated from highschool. We observe that the mean of mom's IQ who graduated from highschool is higher than that of not graduated from highschool. Interestingly, fomr the given data, we observe that the IQ of the highschool graduated mom can still be lower than those who did not graduate from highschool.

```
#relationships between mom's IQ and whether or not they completed high school
ggplot(data=kidiq)+geom_boxplot(aes(x=as.factor(mom_hs),y=mom_iq,color=as.factor(mom_hs)))
```



Original Model

In order to compare the density plot as required in question2, we keep original model fitted with `sigma0=01`.

```
y <- kidiq$kid_score
mu0 <- 80
sigma0 <- 10

# named list to input for stan function
data <- list(y = y,
             N = length(y),
             mu0 = mu0,
             sigma0 = sigma0)
```

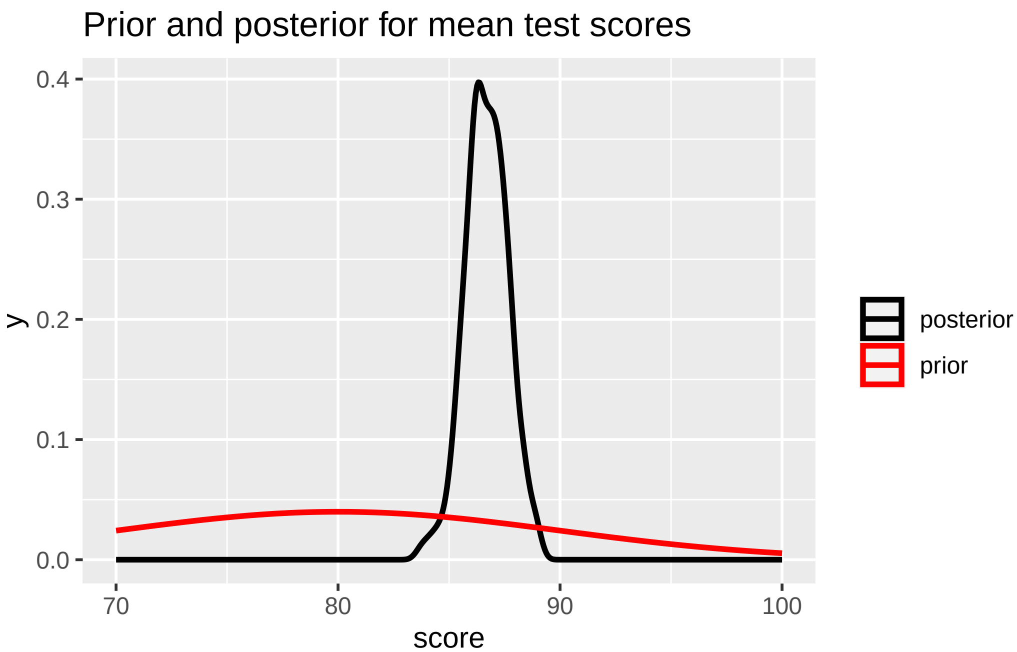
Now we can run the model:

```
fit <- stan(file = here("labs/kids2.stan"),
            data = data,
            chains = 3,
            iter = 500)
```

```r
dsamples <- fit  |>
  gather_draws(mu, sigma) # gather = long format
dsamples
```

```
# A tibble: 1,500 x 5
# Groups:   .variable [2]
   .chain .iteration .draw .variable .value
    <int>      <int> <int> <chr>      <dbl>
 1      1          1     1 mu          86.2
 2      1          2     2 mu          87.0
 3      1          3     3 mu          85.9
 4      1          4     4 mu          88.2
 5      1          5     5 mu          87.9
 6      1          6     6 mu          86.3
 7      1          7     7 mu          87.3
 8      1          8     8 mu          88.1
 9      1          9     9 mu          87.5
10      1         10    10 mu          86.9
# ... with 1,490 more rows
```

```r
dsamples |>
  filter(.variable == "mu") |>
  ggplot(aes(.value, color = "posterior")) + geom_density(size = 1) +
  xlim(c(70, 100)) +
  stat_function(fun = dnorm,
        args = list(mean = mu0,
                    sd = sigma0),
        aes(colour = 'prior'), size = 1) +
  scale_color_manual(name = "", values = c("prior" = "red", "posterior" = "black")) +
  ggtitle("Prior and posterior for mean test scores") +
  xlab("score")
```



Question 2

Change the prior to be much more informative (by changing the standard deviation to be 0.1). Rerun the model. Do the estimates change? Plot the prior and posterior densities.

The estimates changed after changing the standard deviation as follows.`mu` decreased from 86 to around 80. `sigma` increased from 20 to 21.

```
y <- kidiq$kid_score
mu0 <- 80
sigma0 <- 0.1

# named list to input for stan function
data <- list(y = y,
             N = length(y),
             mu0 = mu0,
             sigma0 = sigma0)
```

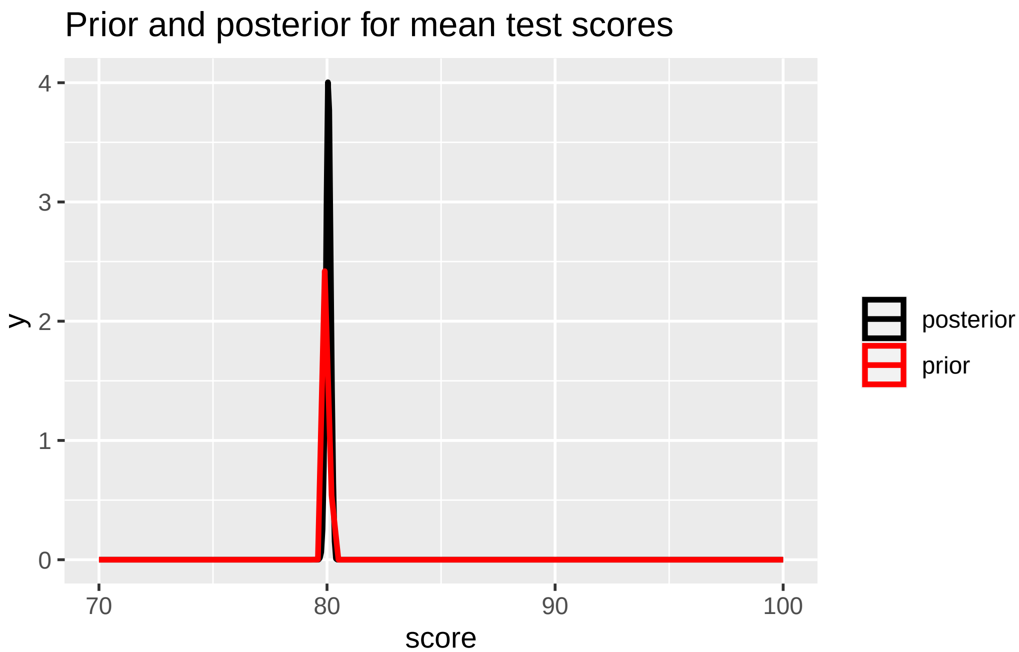
```
#check the model fit
fit <- stan(file = here("~/Documents/2024/STA2201-Methods-of-Applied-Statistics/labs/kids2
            data = data,
            chains = 3,
```

```r
              iter = 500)
```

```r
fit
```

```
Inference for Stan model: anon_model.
3 chains, each with iter=500; warmup=250; thin=1;
post-warmup draws per chain=250, total post-warmup draws=750.

          mean se_mean   sd     2.5%      25%      50%      75%    97.5% n_eff
mu       80.06    0.00 0.10    79.86    80.00    80.06    80.13    80.26   560
sigma    21.38    0.03 0.73    20.05    20.87    21.34    21.85    22.86   810
lp__  -1548.38    0.05 0.98 -1550.92 -1548.85 -1548.05 -1547.65 -1547.39   391
      Rhat
mu       1
sigma    1
lp__     1

Samples were drawn using NUTS(diag_e) at Thu Feb 15 22:31:23 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

```r
dsamples<-fit  |>
  gather_draws(mu, sigma)

dsamples |>
filter(.variable == "mu") |>
ggplot(aes(.value, color = "posterior")) + geom_density(size = 1) +
xlim(c(70, 100)) +
stat_function(fun = dnorm,
      args = list(mean = mu0,
                  sd = sigma0),
      aes(colour = 'prior'), size = 1) +
scale_color_manual(name = "", values = c("prior" = "red", "posterior" = "black")) +
ggtitle("Prior and posterior for mean test scores") +
xlab("score")
```



Adding covariates

Now let's see how kid's test scores are related to mother's education. We want to run the simple linear regression

$$Score = \alpha + \beta X$$

where $X = 1$ if the mother finished high school and zero otherwise.

`kid3.stan` has the stan model to do this. Notice now we have some inputs related to the design matrix X and the number of covariates (in this case, it's just 1).

Let's get the data we need and run the model.

```
X <- as.matrix(kidiq$mom_hs, ncol = 1) # force this to be a matrix
K <- 1

data <- list(y = y, N = length(y),
             X =X, K = K)
fit2 <- stan(file = here("labs/kids3.stan"),
            data = data,
            iter = 1000)
```

Question 3

a) Confirm that the estimates of the intercept and slope are comparable to results from `lm()`

To show that the estimates of the intercept and slope are comparable for the results from `lm()`, we first fit a model using `lm()`. Then, we compare the fitted model with the `fit2` obtained from the bayes regression.

The the intercept term for linear regression is estimated as 77.548, while for bayes regression we have 77.96 as the mean intercept. The two values only differed by 0.412. Given the scale of the data, the means are comparable. We then look at the coefficient term for `mom_hs`. The coefficient for `lm()` is 11.771, for bayes regressoin, the coefficient is 11.25. The second one is just slightly greater than the one estimated from linear models. In addition, the estimated standard errors of these coefficients are similar as well. Thus, the results of the two models are comparable.

```r
fit.lm<-lm(kid_score~mom_hs,data=kidiq)
summary(fit.lm)
```

```

Call:
lm(formula = kid_score ~ mom_hs, data = kidiq)

Residuals:
   Min     1Q Median     3Q    Max 
-57.55 -13.32   2.68  14.68  58.45 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   77.548      2.059  37.670  < 2e-16 ***
mom_hs        11.771      2.322   5.069 5.96e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 19.85 on 432 degrees of freedom
Multiple R-squared:  0.05613,	Adjusted R-squared:  0.05394 
F-statistic: 25.69 on 1 and 432 DF,  p-value: 5.957e-07
```

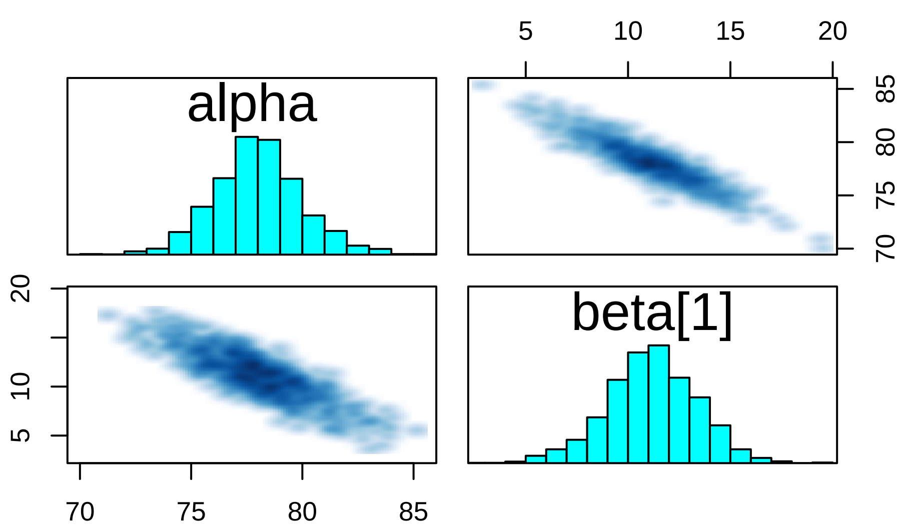
```r
fit2
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.

            mean se_mean   sd     2.5%      25%      50%      75%    97.5%
alpha      78.02    0.07 1.98    74.24    76.79    77.98    79.25    82.18
beta[1]    11.16    0.08 2.20     6.46     9.78    11.18    12.58    15.36
sigma      19.82    0.02 0.70    18.49    19.35    19.80    20.29    21.24
lp__    -1514.43    0.05 1.31 -1517.80 -1515.00 -1514.09 -1513.47 -1512.97
        n_eff Rhat
alpha     848 1.00
beta[1]   854 1.00
sigma    1100 1.00
lp__      727 1.01

Samples were drawn using NUTS(diag_e) at Thu Feb 15 22:32:12 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

b) Do a **pairs** plot to investigate the joint sample distributions of the slope and intercept. Comment briefly on what you see. Is this potentially a problem?

From the pairs plot below, we observe a negative correlation between the intercept and the coefficient. This may not potentially be a problem, since in linear regression, $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x}$, indicating the increase coefficient, the other one decreases.

```
pairs(fit2, pars = c("alpha","beta[1]"))
```



Question 4

Add in mother's IQ as a covariate and rerun the model. Please mean center the covariate before putting it into the model. Interpret the coefficient on the (centered) mum's IQ.

```
kidiq$mom_iq_centered <- kidiq$mom_iq - mean(kidiq$mom_iq)

X <- as.matrix(kidiq[, c("mom_hs", "mom_iq_centered")])
K <- 2

data3 <- list(y = y,
              N = length(y),
              X = X,
              K = K
)
fit3 <- stan(file = here("labs/kids3.stan"),
             data = data3,
             iter = 1000)
```

```
fit3
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.

            mean se_mean   sd     2.5%      25%      50%      75%    97.5%
alpha      82.37    0.06 1.89    78.54    81.15    82.41    83.60    86.03
beta[1]     5.64    0.07 2.14     1.41     4.15     5.66     7.06     9.97
beta[2]     0.57    0.00 0.06     0.45     0.53     0.57     0.61     0.69
sigma      18.10    0.01 0.60    16.90    17.70    18.11    18.47    19.28
lp__    -1474.38    0.06 1.42 -1478.06 -1475.06 -1474.04 -1473.35 -1472.67
        n_eff Rhat
alpha    1098 1.00
beta[1]  1079 1.00
beta[2]  1182 1.00
sigma    1631 1.00
lp__      551 1.01

Samples were drawn using NUTS(diag_e) at Thu Feb 15 22:32:14 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

The coefficient of the centered mom's IQ is estimated as 0.56. This value indicates that, if we keep other variables the same, on average, an unit increase in the mom's IQ increases the kids' score by 0.56.

Question 5

Confirm the results from Stan agree with `lm()`

From the output below, we observe that, the centered mom's IQ from `lm()` is estimated as 0.56 as well, with aligns with our bayes regression estimate.

```r
fit3.lm <- lm(kid_score ~ mom_hs + mom_iq_centered, data=kidiq)
summary(fit3.lm)
```

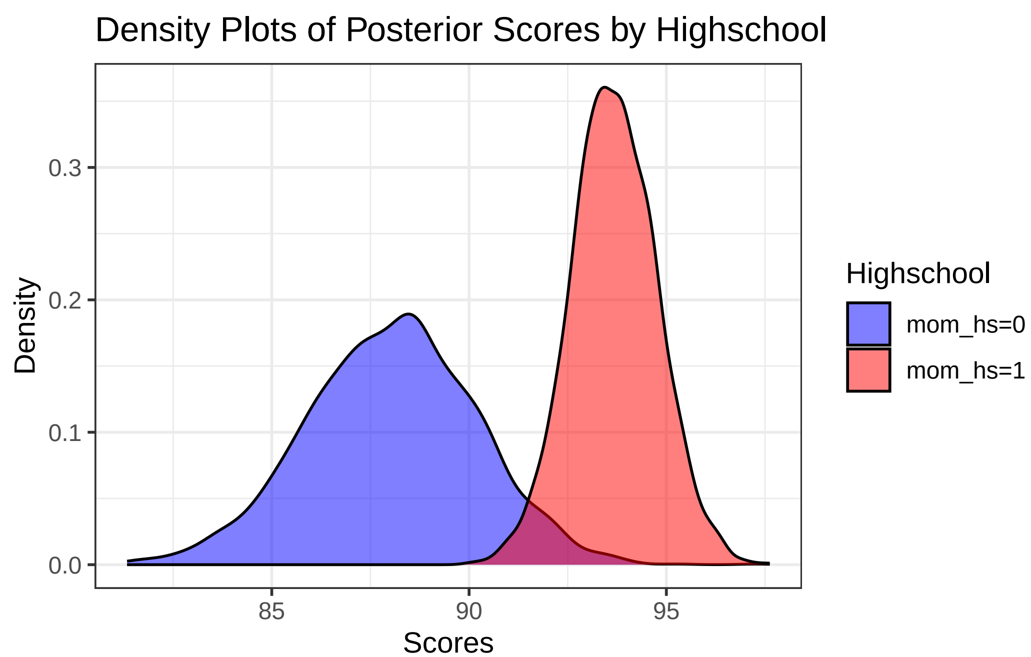
```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq_centered, data = kidiq)

Residuals:
    Min      1Q  Median      3Q     Max
-52.873 -12.663   2.404  11.356  49.545

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     82.12214    1.94370  42.250  < 2e-16 ***
mom_hs           5.95012    2.21181   2.690  0.00742 **
mom_iq_centered  0.56391    0.06057   9.309  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.14 on 431 degrees of freedom
Multiple R-squared:  0.2141,    Adjusted R-squared:  0.2105
F-statistic: 58.72 on 2 and 431 DF,  p-value: < 2.2e-16
```

Question 6

Plot the posterior estimates of scores by education of mother for mothers who have an IQ of 110.

From density plot below, we observe that, for moms who have an IQ of 110, if the moms did not graduate from high school, their kids' IQ is approximately nornally distributed with mean around 87, if the moms has graduated from highschool, the disribution of their kids IQ has a mean around 93.

```
posterior_samples <- extract(fit3)
b0=posterior_samples$alpha
b1=posterior_samples$beta[,1]
b2=posterior_samples$beta[,2]
e=posterior_samples$sigma

posterior.hs0=b0+b2*(110-mean(kidiq$mom_iq))
posterior.hs1=b0+b1+b2*(110-mean(kidiq$mom_iq))

density<- data.frame(
  Scores = c(posterior.hs0, posterior.hs1),
  Highschool = rep(c("mom_hs=0", "mom_hs=1"), each = length(posterior.hs0)))

ggplot(density, aes(x = Scores, fill = Highschool)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plots of Posterior Scores by Highschool",x = "Scores", y = "Densit
```




```
#a<-density|> filter(Highschool=="mom_hs=1")
#a$Scores |>mean()
```

Question 7

Generate and plot (as a histogram) samples from the posterior predictive distribution for a new kid with a mother who graduated high school and has an IQ of 95.

We observe that the estimated kids IQ are approximately normally distributed with a mean around 103 .

```
posterior.hs1.95=b0+b1+b2*(95-mean(kidiq$mom_iq))+e

ggplot()+geom_histogram(aes(x=posterior.hs1.95),fill="lightblue")+theme_bw()+labs(x="IQ",t
```