lab5

13/02/23

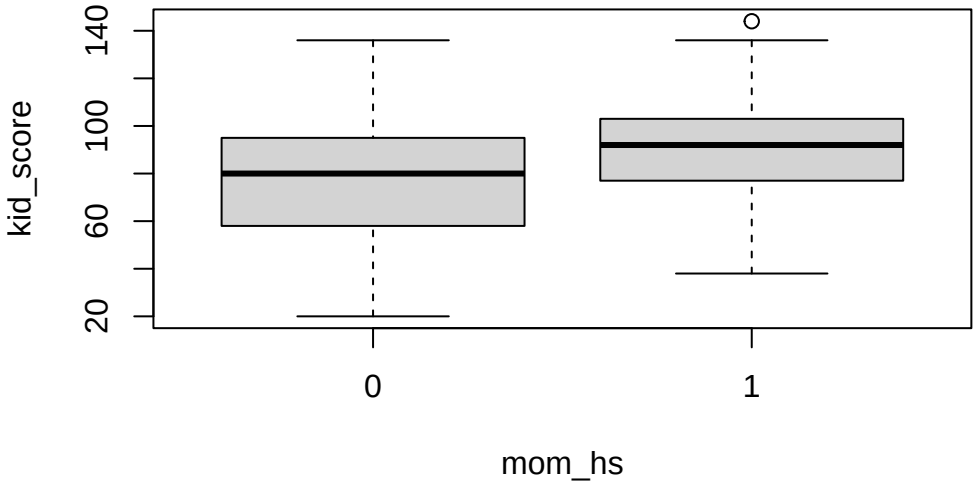
```
library(tidyverse)
library(rstan)
library(tidybayes)
library(here)
```

```
kidiq <- read_rds(here("data","kidiq.RDS"))
kidiq
```

```
# A tibble: 434 x 4
   kid_score mom_hs mom_iq mom_age
       <int>  <dbl>  <dbl>   <int>
 1        65      1  121.        27
 2        98      1   89.4       25
 3        85      1  115.        27
 4        83      1   99.4       25
 5       115      1   92.7       27
 6        98      0  108.        18
 7        69      1  139.        20
 8       106      1  125.        23
 9       102      1   81.6       24
10        95      1   95.1       19
# ... with 424 more rows
```

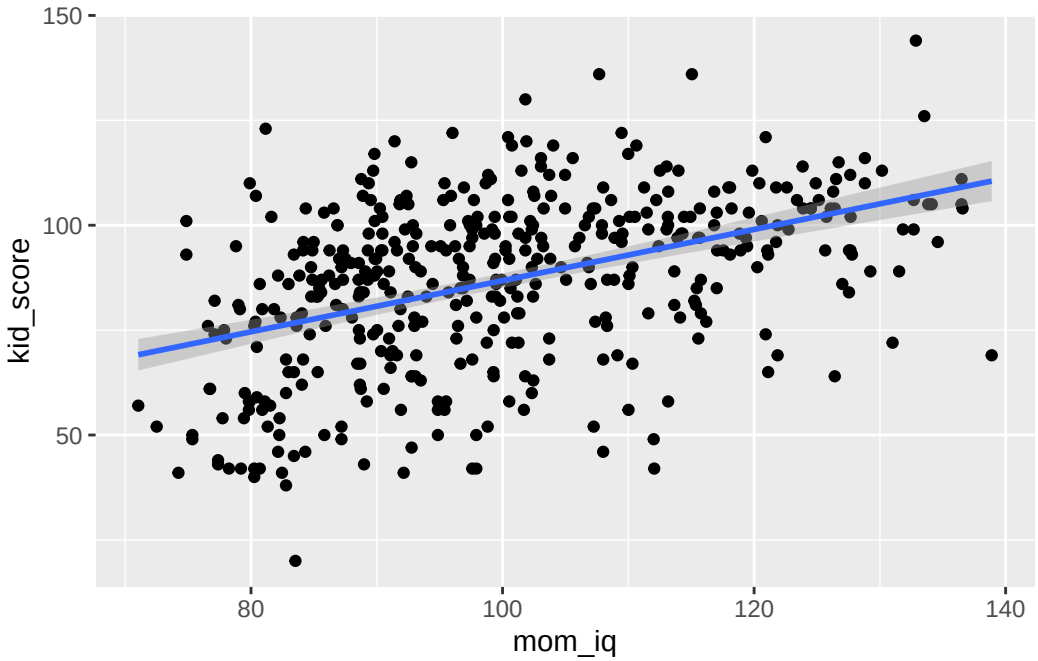
Question 1

```
boxplot(kid_score ~mom_hs , data = kidiq)
```



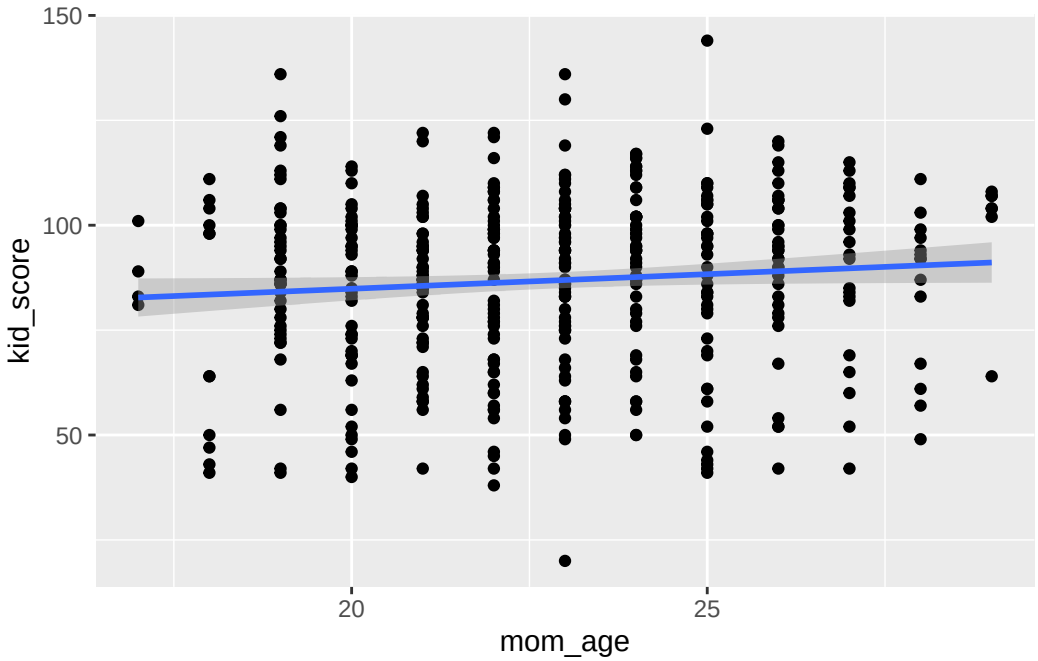
The first interest point is that the status of mother's completion of high school is a binary data 0 and 1. Based on the box plot, the children of mothers who completed high school scored higher than children of mothers who did not.

```
ggplot(kidiq,aes(x=mom_iq,y=kid_score))+geom_point()+geom_smooth(method='lm')
```



The second interest point is that the kid score increases as the their mom's iq increases.

```
ggplot(kidiq,aes(x=mom_age,y=kid_score))+geom_point()+geom_smooth(method='lm')
```



The third interest point is that the kid score do not have strong relationship with the their mom's age.

Question 2

```
y <- kidiq$kid_score
mu0 <- 80
sigma1 <- 0.1
```

```
data1 <- list(y = y,
              N = length(y),
              mu0 = mu0,
              sigma0 = sigma1)
```

```
fit1 <- stan(file = here("code/models/kids2.stan"),
             data = data1,
             chains = 3,
             iter = 500)
```

```
SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 1.9e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.19 seconds.
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:   1 / 500 [  0%]  (Warmup)
Chain 1: Iteration:  50 / 500 [ 10%]  (Warmup)
Chain 1: Iteration: 100 / 500 [ 20%]  (Warmup)
Chain 1: Iteration: 150 / 500 [ 30%]  (Warmup)
Chain 1: Iteration: 200 / 500 [ 40%]  (Warmup)
Chain 1: Iteration: 250 / 500 [ 50%]  (Warmup)
Chain 1: Iteration: 251 / 500 [ 50%]  (Sampling)
Chain 1: Iteration: 300 / 500 [ 60%]  (Sampling)
Chain 1: Iteration: 350 / 500 [ 70%]  (Sampling)
Chain 1: Iteration: 400 / 500 [ 80%]  (Sampling)
Chain 1: Iteration: 450 / 500 [ 90%]  (Sampling)
Chain 1: Iteration: 500 / 500 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.003 seconds (Warm-up)
Chain 1:                0.002 seconds (Sampling)
Chain 1:                0.005 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 3e-06 seconds
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.03 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:   1 / 500 [  0%]  (Warmup)
Chain 2: Iteration:  50 / 500 [ 10%]  (Warmup)
Chain 2: Iteration: 100 / 500 [ 20%]  (Warmup)
Chain 2: Iteration: 150 / 500 [ 30%]  (Warmup)
Chain 2: Iteration: 200 / 500 [ 40%]  (Warmup)
Chain 2: Iteration: 250 / 500 [ 50%]  (Warmup)
Chain 2: Iteration: 251 / 500 [ 50%]  (Sampling)
Chain 2: Iteration: 300 / 500 [ 60%]  (Sampling)
Chain 2: Iteration: 350 / 500 [ 70%]  (Sampling)
Chain 2: Iteration: 400 / 500 [ 80%]  (Sampling)
Chain 2: Iteration: 450 / 500 [ 90%]  (Sampling)
```

```
Chain 2: Iteration: 500 / 500 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.003 seconds (Warm-up)
Chain 2:                0.002 seconds (Sampling)
Chain 2:                0.005 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
Chain 3:
Chain 3: Gradient evaluation took 4e-06 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.04 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:   1 / 500 [  0%]  (Warmup)
Chain 3: Iteration:  50 / 500 [ 10%]  (Warmup)
Chain 3: Iteration: 100 / 500 [ 20%]  (Warmup)
Chain 3: Iteration: 150 / 500 [ 30%]  (Warmup)
Chain 3: Iteration: 200 / 500 [ 40%]  (Warmup)
Chain 3: Iteration: 250 / 500 [ 50%]  (Warmup)
Chain 3: Iteration: 251 / 500 [ 50%]  (Sampling)
Chain 3: Iteration: 300 / 500 [ 60%]  (Sampling)
Chain 3: Iteration: 350 / 500 [ 70%]  (Sampling)
Chain 3: Iteration: 400 / 500 [ 80%]  (Sampling)
Chain 3: Iteration: 450 / 500 [ 90%]  (Sampling)
Chain 3: Iteration: 500 / 500 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.003 seconds (Warm-up)
Chain 3:                0.003 seconds (Sampling)
Chain 3:                0.006 seconds (Total)
Chain 3:
```

```
fit1
```

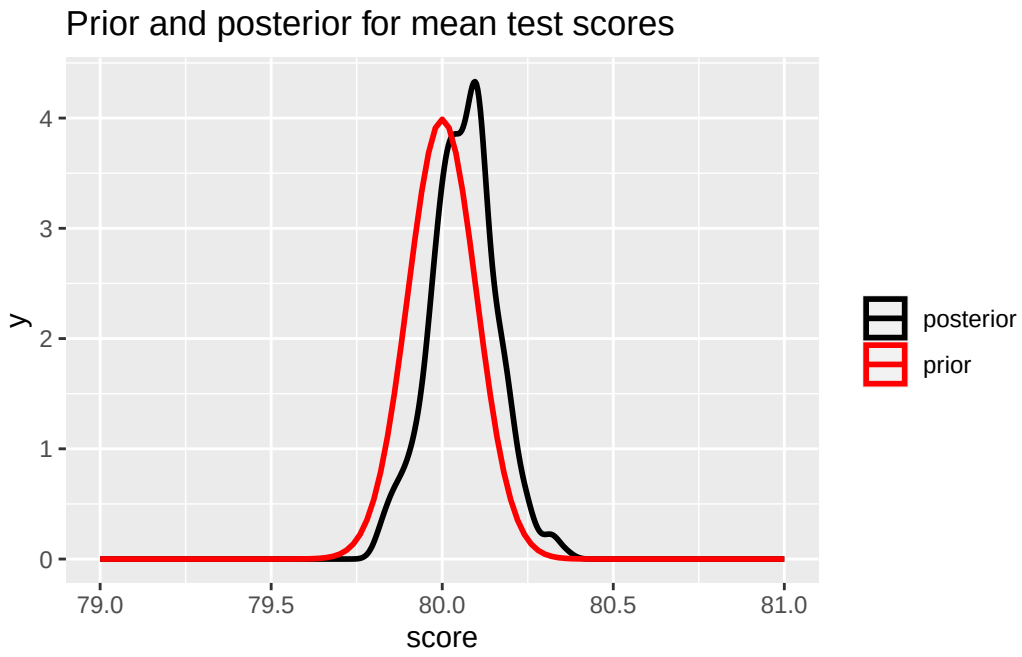
```
Inference for Stan model: anon_model.
3 chains, each with iter=500; warmup=250; thin=1;
post-warmup draws per chain=250, total post-warmup draws=750.

          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    80.12    80.25   515
sigma    21.42    0.03 0.72    20.02    20.95    21.39    21.82    22.90   701
lp__  -1548.33    0.06 1.03 -1551.03 -1548.78 -1547.98 -1547.61 -1547.39   321
```

```
      Rhat
mu       1
sigma    1
lp__     1

Samples were drawn using NUTS(diag_e) at Mon Feb 13 12:25:01 2023.
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 estimated coefficients of fit1 model is mu 80.06 and sigma 21.44. Comparing to fit model(mu 86.74 sigma 20.40), the mu value decreased and the sigma value slightly increased.

```r
dsamples <- fit1  |>
  gather_draws(mu, sigma) # gather = long format
```

```r
dsamples |>
  filter(.variable == "mu") |>
  ggplot(aes(.value, color = "posterior")) + geom_density(size = 1) +
  xlim(c(79, 81)) +
  stat_function(fun = dnorm,
        args = list(mean = mu0,
                    sd = sigma1),
        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 3

```
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("code/models/kids3.stan"),
             data = data,
             iter = 1000)
```

```
SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 4.6e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.46 seconds.
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 1: Iteration: 100 / 1000 [ 10%]  (Warmup)
```

```
Chain 1: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 1: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 1: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 1: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 1: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 1: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 1: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 1: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 1: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 1: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.075 seconds (Warm-up)
Chain 1:                0.033 seconds (Sampling)
Chain 1:                0.108 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 9e-06 seconds
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 2: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 2: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 2: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 2: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 2: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 2: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 2: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 2: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 2: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 2: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 2: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.059 seconds (Warm-up)
Chain 2:                0.042 seconds (Sampling)
Chain 2:                0.101 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
Chain 3:
```

```
Chain 3: Gradient evaluation took 8e-06 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.08 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 3: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 3: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 3: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 3: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 3: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 3: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 3: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 3: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 3: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 3: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 3: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.063 seconds (Warm-up)
Chain 3:                0.047 seconds (Sampling)
Chain 3:                0.11 seconds (Total)
Chain 3:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
Chain 4:
Chain 4: Gradient evaluation took 9e-06 seconds
Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
Chain 4: Adjust your expectations accordingly!
Chain 4:
Chain 4:
Chain 4: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 4: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 4: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 4: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 4: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 4: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 4: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 4: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 4: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 4: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 4: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 4: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 4:
```

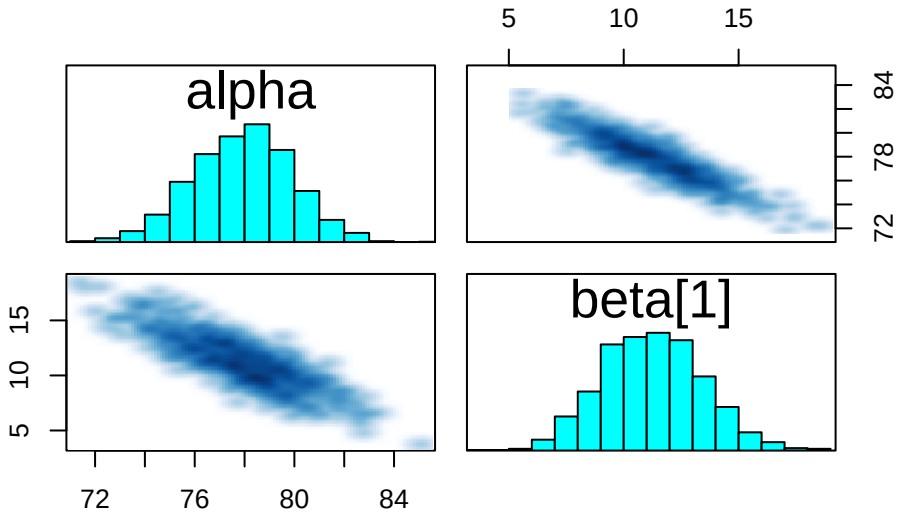
```
Chain 4:  Elapsed Time: 0.076 seconds (Warm-up)
Chain 4:                0.045 seconds (Sampling)
Chain 4:                0.121 seconds (Total)
Chain 4:
```

```r
summary(lm(kid_score~mom_hs,data = kidiq))
```

```
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
```

(a) The coefficients of fit2 is alpha 77.773 beta[1] 11.47.
The coefficients of lm() model is (Intercept) 77.548 and mom_hs 11.771. Their value are very close to each other.

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



(b) There is a large variation in alpha and beta[1], it would make the sampling harder.

Question 4

```
center_scale <- function(x) {
    scale(x, scale = FALSE)
}
```

```
X4 <- cbind(X,center_scale(kidiq$mom_iq)) # force this to be a matrix
K4 <- 2
data4 <- list(y = y, N = length(y),
              X =X4, K = K4)
fit4 <- stan(file = here("code/models/kids3.stan"),
             data = data4,
             iter = 1000)
```

```
SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 1.5e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.15 seconds.
```

```
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 1: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 1: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 1: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 1: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 1: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 1: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 1: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 1: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 1: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 1: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 1: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.077 seconds (Warm-up)
Chain 1:                0.044 seconds (Sampling)
Chain 1:                0.121 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 9e-06 seconds
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 2: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 2: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 2: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 2: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 2: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 2: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 2: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 2: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 2: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 2: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 2: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.068 seconds (Warm-up)
Chain 2:                0.05 seconds (Sampling)
```

```
Chain 2:                0.118 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 3).
Chain 3:
Chain 3: Gradient evaluation took 9e-06 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 3: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 3: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 3: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 3: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 3: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 3: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 3: Iteration: 600 / 1000 [ 60%]  (Sampling)
Chain 3: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 3: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 3: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 3: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.078 seconds (Warm-up)
Chain 3:                0.045 seconds (Sampling)
Chain 3:                0.123 seconds (Total)
Chain 3:

SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
Chain 4:
Chain 4: Gradient evaluation took 9e-06 seconds
Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.09 seconds.
Chain 4: Adjust your expectations accordingly!
Chain 4:
Chain 4:
Chain 4: Iteration:   1 / 1000 [  0%]  (Warmup)
Chain 4: Iteration: 100 / 1000 [ 10%]  (Warmup)
Chain 4: Iteration: 200 / 1000 [ 20%]  (Warmup)
Chain 4: Iteration: 300 / 1000 [ 30%]  (Warmup)
Chain 4: Iteration: 400 / 1000 [ 40%]  (Warmup)
Chain 4: Iteration: 500 / 1000 [ 50%]  (Warmup)
Chain 4: Iteration: 501 / 1000 [ 50%]  (Sampling)
Chain 4: Iteration: 600 / 1000 [ 60%]  (Sampling)
```

```
Chain 4: Iteration: 700 / 1000 [ 70%]  (Sampling)
Chain 4: Iteration: 800 / 1000 [ 80%]  (Sampling)
Chain 4: Iteration: 900 / 1000 [ 90%]  (Sampling)
Chain 4: Iteration: 1000 / 1000 [100%]  (Sampling)
Chain 4:
Chain 4:  Elapsed Time: 0.081 seconds (Warm-up)
Chain 4:                0.05 seconds (Sampling)
Chain 4:                0.131 seconds (Total)
Chain 4:
```

```
fit4
```

```
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     82.28    0.06 1.96    78.40    80.95    82.25    83.57    86.20
beta[1]    5.74    0.07 2.26     1.37     4.20     5.80     7.20    10.17
beta[2]    0.57    0.00 0.06     0.45     0.52     0.56     0.61     0.69
sigma     18.13    0.02 0.62    16.92    17.71    18.12    18.54    19.39
lp__    -1474.45    0.05 1.44 -1478.03 -1475.15 -1474.13 -1473.37 -1472.65
        n_eff Rhat
alpha    1043 1.00
beta[1]   985 1.00
beta[2]  1064 1.00
sigma    1241 1.00
lp__      817 1.01

Samples were drawn using NUTS(diag_e) at Mon Feb 13 12:25:39 2023.
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).
```

Fixed the mother completing high school status, For every 1 increase in IQ, the child's score increases by 0.57; Fixed the mother's IQ, Children of mothers who completed high school scored 5.63 higher than children of mothers who did not.

Question 5

```
summary(lm(kid_score~mom_hs+center_scale(mom_iq),data=kidiq))
```

```
Call:
lm(formula = kid_score ~ mom_hs + center_scale(mom_iq), 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 **
center_scale(mom_iq)  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
```

The coefficients of fit4 is alpha 82.36 beta[1] 5.63 beta[2] 0.57.
The coefficients of lm() model is (Intercept) 82.122 mom_hs 5.95 and center_scale(mom_iq) 0.564. Their value are very close to each other.

Question 6

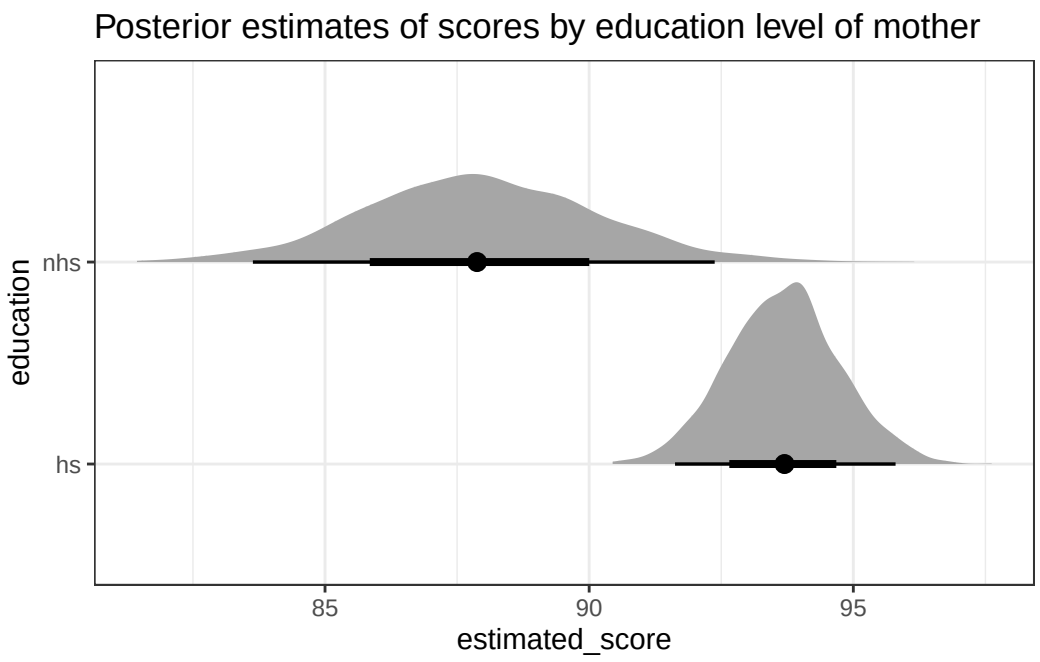
```
IQ6 = 110- mean(kidiq$mom_iq)
```

```
pivot_wider(spread_draws(fit4,alpha, beta[k], sigma),names_from = k,names_prefix = "beta",
```

```
# A tibble: 2,000 x 7
   .chain .iteration .draw alpha sigma beta1 beta2
    <int>      <int> <int> <dbl> <dbl> <dbl> <dbl>
 1      1          1     1  81.0  18.4  5.56 0.564
 2      1          2     2  82.6  17.8  4.12 0.709
```

```
 3      1          3     3  84.0  18.2  2.82 0.505
 4      1          4     4  84.7  17.9  3.39 0.506
 5      1          5     5  83.0  18.1  4.60 0.659
 6      1          6     6  81.3  17.8  6.40 0.547
 7      1          7     7  85.3  18.3  1.73 0.612
 8      1          8     8  83.3  17.9  4.48 0.503
 9      1          9     9  81.7  17.6  6.30 0.523
10      1         10    10  82.4  17.3  6.36 0.505
# ... with 1,990 more rows
```

```
fit4 |>
  spread_draws(alpha, beta[k], sigma) |>
  pivot_wider(names_from = k,names_prefix = "beta",values_from = beta)|>
  mutate(nhs = alpha+0*beta1+IQ6*beta2, # no high school is just the intercept
         hs = alpha + 1*beta1 + IQ6*beta2) |>
  select(nhs, hs) |>
  pivot_longer(nhs:hs, names_to = "education", values_to = "estimated_score") |>
  ggplot(aes(y = education, x = estimated_score)) +
  stat_halfeye() +
  theme_bw() +
  ggtitle("Posterior estimates of scores by education level of mother")
```



Question 7

```
IQ7 = 95-mean(kidiq$mom_iq)
post_samples4 <- extract(fit4)
prediction <- post_samples4[["alpha"]]+post_samples4[["beta"]][,1]*1+post_samples4[["beta"
y4 <- rnorm(mean= prediction, sd = post_samples4[['sigma']],n=2000)
hist(y4)
```