Priyanka_Verma: Bayesian linear regression and introduction to Stan

13/02/23

Introduction

Looking at the kid's test score data set (available in resources for the Gelman Hill textbook).

The data look like this:

```
kidiq <- read_rds(here("/Users/vermap/Desktop/STA-2201/applied-stats-23/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
```

As well as the kid's test scores, we have a binary variable indicating whether or not the mother completed high school, the mother's IQ and age.

Descriptives

Question 1

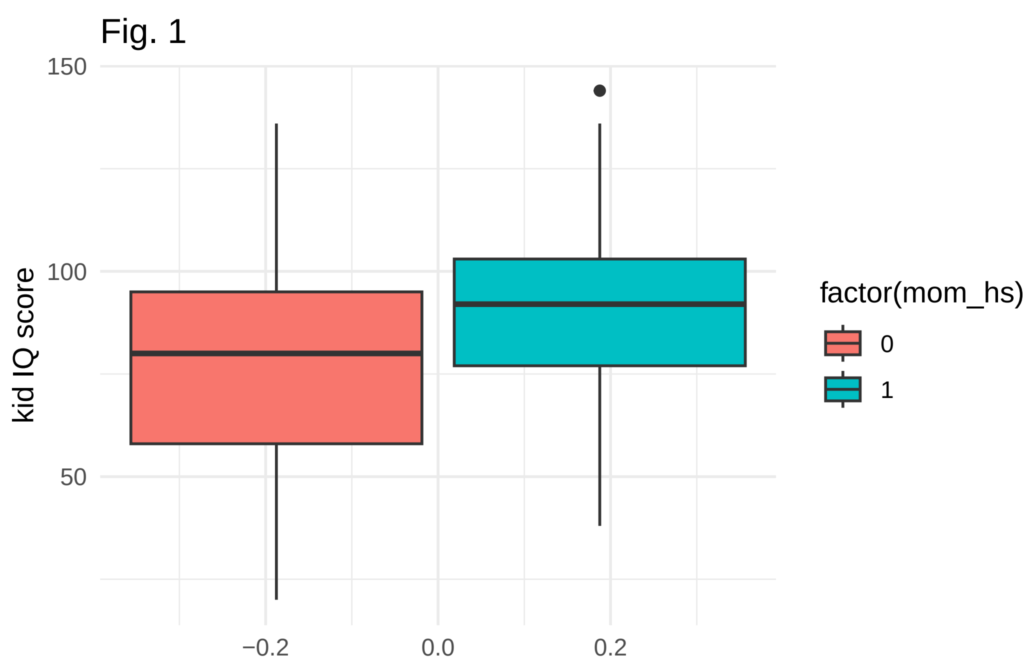
Use plots or tables to show three interesting observations about the data.

- We can see the summary of all the variables. It gives us the idea about the range of values and average values of different variables.

```
summary(kidiq)
```

```
   kid_score         mom_hs           mom_iq          mom_age
 Min.   : 20.0   Min.   :0.0000   Min.   : 71.04   Min.   :17.00
 1st Qu.: 74.0   1st Qu.:1.0000   1st Qu.: 88.66   1st Qu.:21.00
 Median : 90.0   Median :1.0000   Median : 97.92   Median :23.00
 Mean   : 86.8   Mean   :0.7857   Mean   :100.00   Mean   :22.79
 3rd Qu.:102.0   3rd Qu.:1.0000   3rd Qu.:110.27   3rd Qu.:25.00
 Max.   :144.0   Max.   :1.0000   Max.   :138.89   Max.   :29.00
```

- The box plot (Fig.1) shows the range of values of kids iq score with variation in mother's high school education.

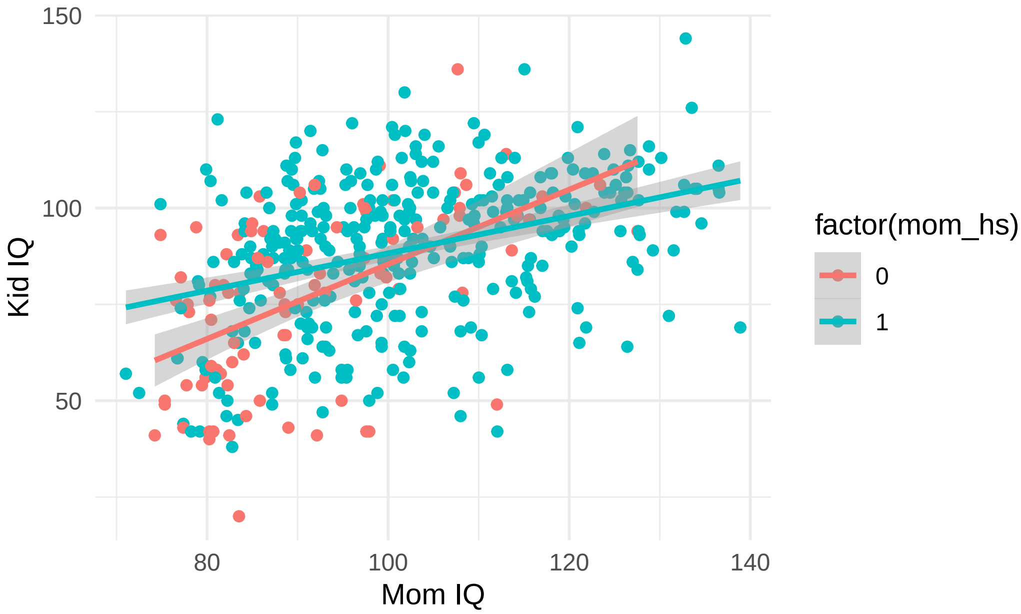


- The average of kids iq score is higher when mom has high school education.

- There is a possible outlier value of kid's iq score (144), however on further looking at the data it does not look unexpected as it is associated with a high mom iq (132). It is interesting to note that the kid has a higher iq score than their mother.

The plot (Fig. 2) shows kids' iq with mothers' iq and its variation by mom high school education.

Fig. 2



- The slopes of the regression of child's test score on mother's IQ differs substantially across subgroups defined by mother's high school completion.

Estimating mean, no covariates

```
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)
```

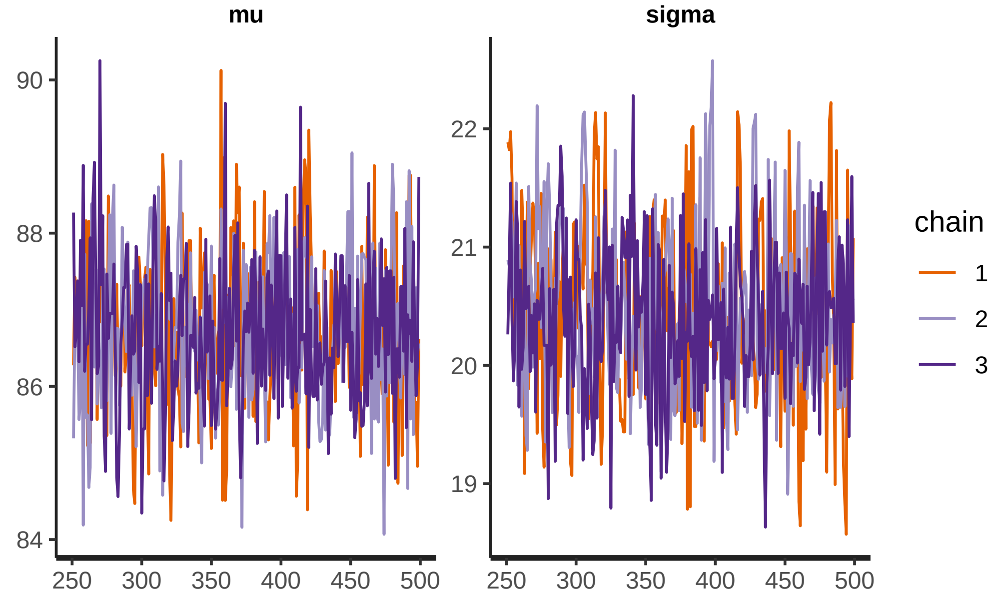
Look at the summary

```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        86.72    0.04 0.98    84.81    86.05    86.67    87.40    88.74   698
sigma     20.45    0.03 0.71    19.10    19.97    20.42    20.91    21.99   592
lp__   -1525.79    0.05 1.03 -1528.57 -1526.10 -1525.49 -1525.04 -1524.79   371
       Rhat
mu     1.00
sigma  1.00
lp__   1.01

Samples were drawn using NUTS(diag_e) at Mon Feb 13 22:12:44 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).
```

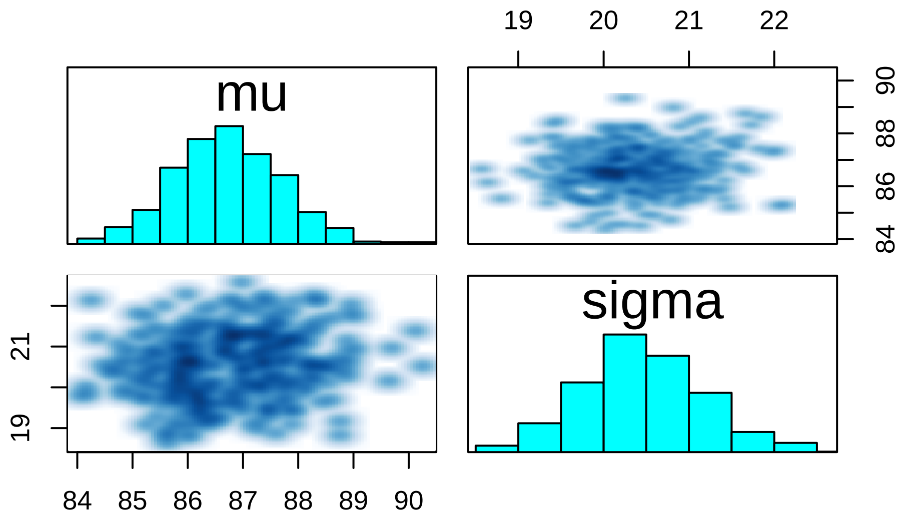
Traceplot

```r
traceplot(fit)
```

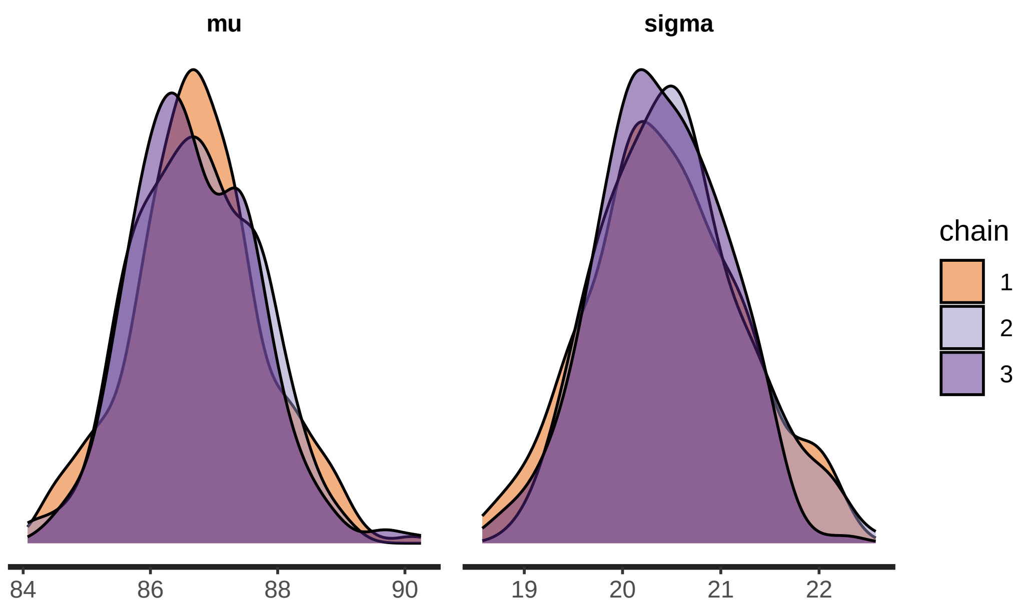


All looks fine.

```
pairs(fit, pars = c("mu", "sigma"))
```

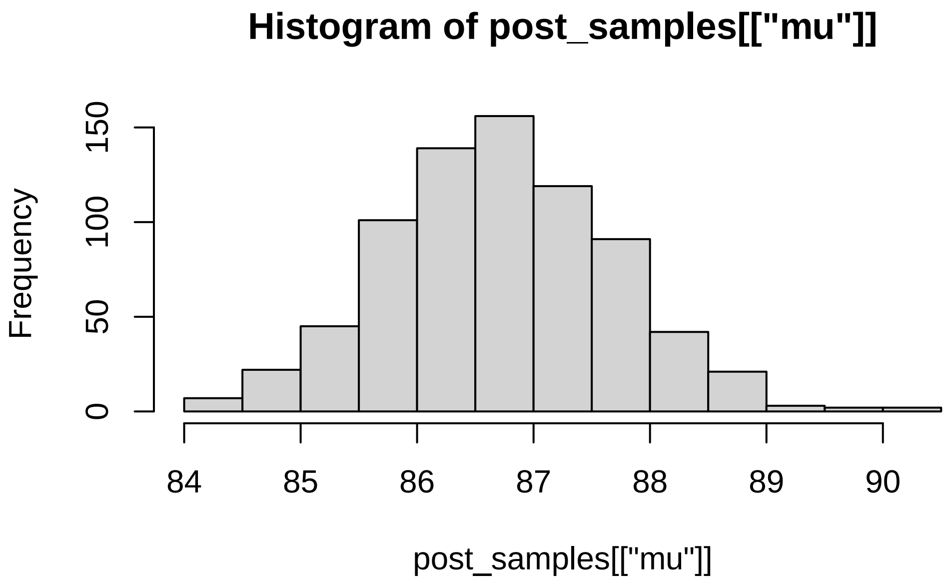



```
stan_dens(fit, separate_chains = TRUE)
```



Understanding output

```
[1] 87.75228 87.32639 86.83278 88.90055 87.29317 87.47591
```




```
[1] 86.67308
```

```
    2.5%
84.80918
```

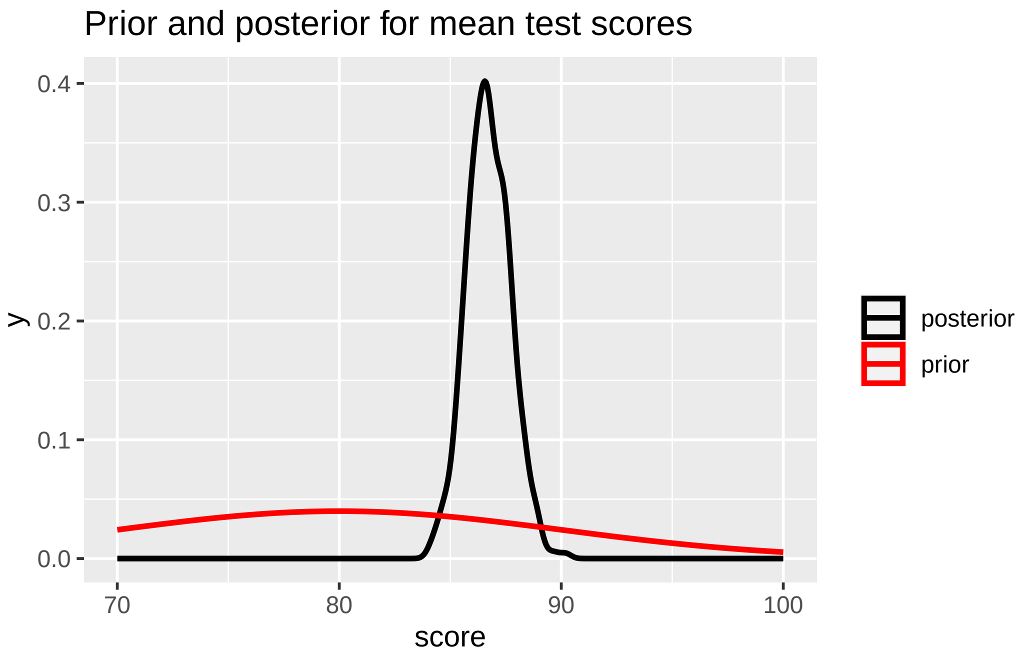
```
   97.5%
88.74026
```

Plot estimates

```
# A tibble: 1,500 x 5
# Groups:   .variable [2]
   .chain .iteration .draw .variable .value
    <int>      <int> <int> <chr>      <dbl>
 1      1          1     1 mu          86.3
 2      1          2     2 mu          87.4
 3      1          3     3 mu          87.3
```

```
 4      1          4     4 mu           86.8
 5      1          5     5 mu           86.7
 6      1          6     6 mu           87.3
 7      1          7     7 mu           86.3
 8      1          8     8 mu           87.0
 9      1          9     9 mu           87.7
10      1         10    10 mu           88.2
# ... with 1,490 more rows

# A tibble: 750 x 5
   .chain .iteration .draw    mu sigma
    <int>      <int> <int> <dbl> <dbl>
 1      1          1     1  86.3  21.9
 2      1          2     2  87.4  21.8
 3      1          3     3  87.3  22.0
 4      1          4     4  86.8  21.6
 5      1          5     5  86.7  19.9
 6      1          6     6  87.3  20.6
 7      1          7     7  86.3  20.6
 8      1          8     8  87.0  20.3
 9      1          9     9  87.7  20.1
10      1         10    10  88.2  20.0
# ... with 740 more rows

# A tibble: 2 x 7
  .variable .value .lower .upper .width .point .interval
  <chr>      <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
1 mu          86.7   85.5   87.9    0.8 median qi
2 sigma       20.4   19.6   21.4    0.8 median qi
```

Let's plot the density of the posterior samples for mu and add in the prior distribution



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.

```
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)
```

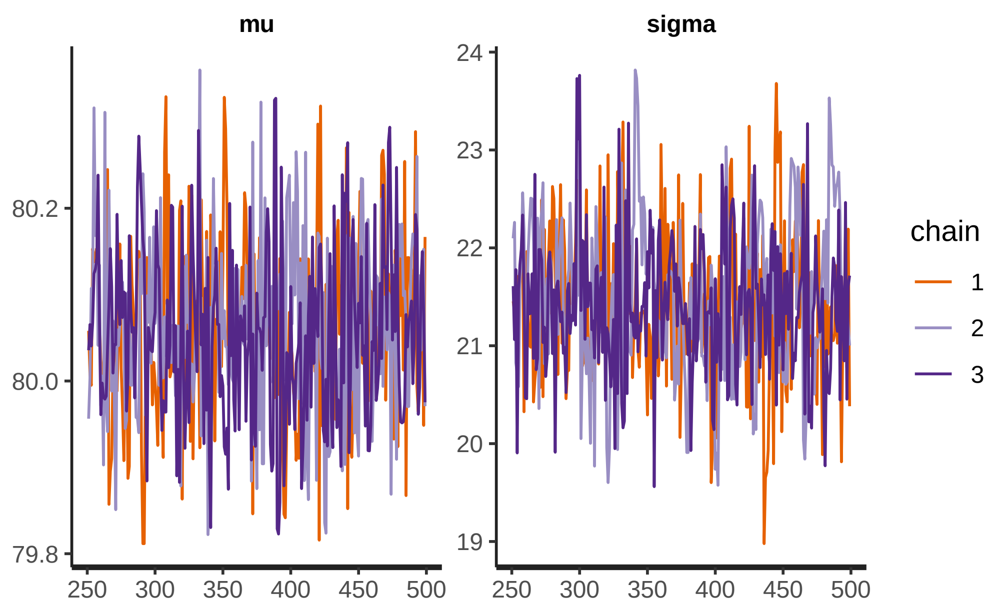
```
fit
```

```
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.
```

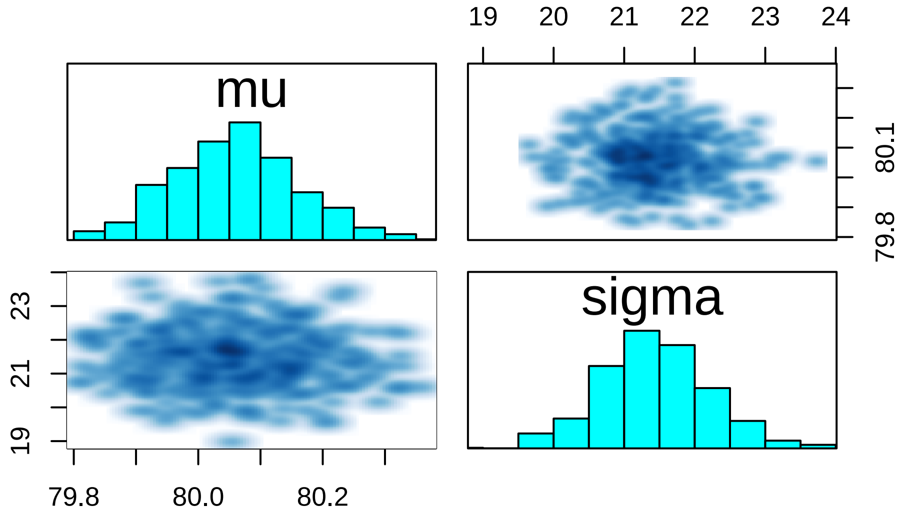
```
          mean se_mean   sd     2.5%      25%      50%      75%    97.5% n_eff
mu       80.06    0.00 0.10    79.87    79.99    80.06    80.13    80.27   609
sigma    21.44    0.04 0.76    19.92    20.92    21.39    21.92    23.00   358
lp__  -1548.44    0.06 1.05 -1551.18 -1548.95 -1548.09 -1547.63 -1547.39   314
      Rhat
mu       1
sigma    1
lp__     1

Samples were drawn using NUTS(diag_e) at Mon Feb 13 22:12:45 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).
```

yes, the output estimates have changed from mu= 86.76 to mu= 80.06 and sigma from 20.39 to 21.44. This is intuitive because our estimates are influenced primarily by the priors, rather than the data, because we have set sigma0 as 0.1- meaning that we are confident in our priors more than the data.

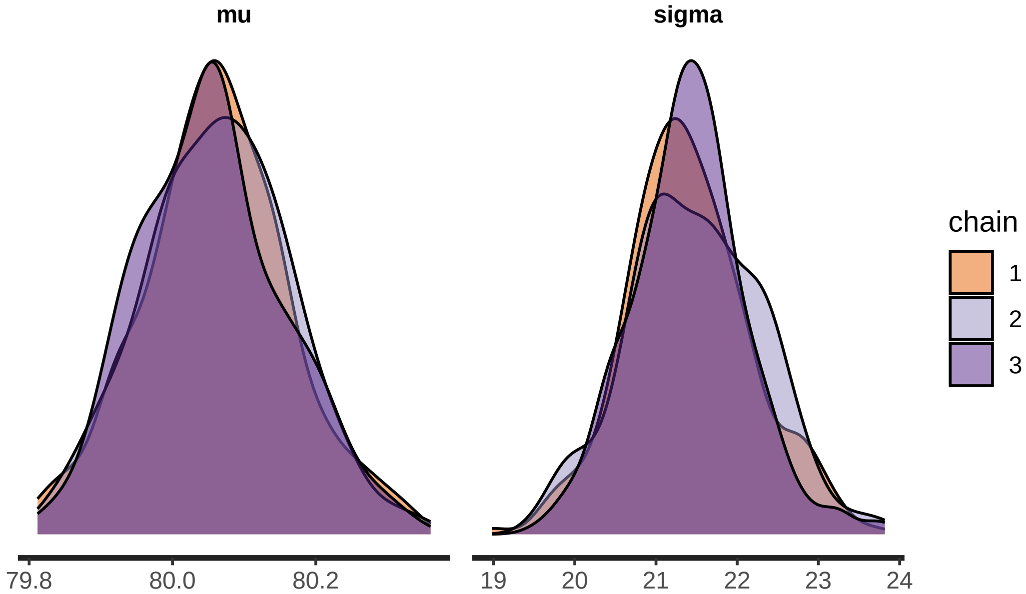
```
traceplot(fit)
```



```
pairs(fit, pars = c("mu", "sigma"))
```




```
stan_dens(fit, separate_chains = TRUE)
```



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).

```
post_samples <- extract(fit2)
names(post_samples)
```

```
[1] "alpha" "beta"  "sigma" "lp__"
```

```
dsamples <- fit2  |>
  gather_draws(alpha, beta[], sigma) # gather = long format
dsamples
```

```
# A tibble: 6,000 x 5
# Groups:   .variable [3]
   .chain .iteration .draw .variable .value
    <int>      <int> <int> <chr>      <dbl>
 1      1          1     1 alpha       78.1
 2      1          2     2 alpha       80.1
 3      1          3     3 alpha       75.7
 4      1          4     4 alpha       77.7
 5      1          5     5 alpha       74.6
 6      1          6     6 alpha       76.0
 7      1          7     7 alpha       78.0
 8      1          8     8 alpha       79.9
 9      1          9     9 alpha       79.6
10      1         10    10 alpha       78.9
# ... with 5,990 more rows
```

```
# wide format
#fit  |>  spread_draws(mu, sigma)

# quickly calculate the quantiles using

dsamples |>
  median_qi(.width = 0.8)
```

```
# A tibble: 3 x 7
  .variable .value .lower .upper .width .point .interval
  <chr>      <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
1 alpha       78.1  75.5    80.8    0.8 median qi
2 beta        11.0   8.12   14.0    0.8 median qi
3 sigma       19.8  19.0    20.7    0.8 median qi
```

Question 3

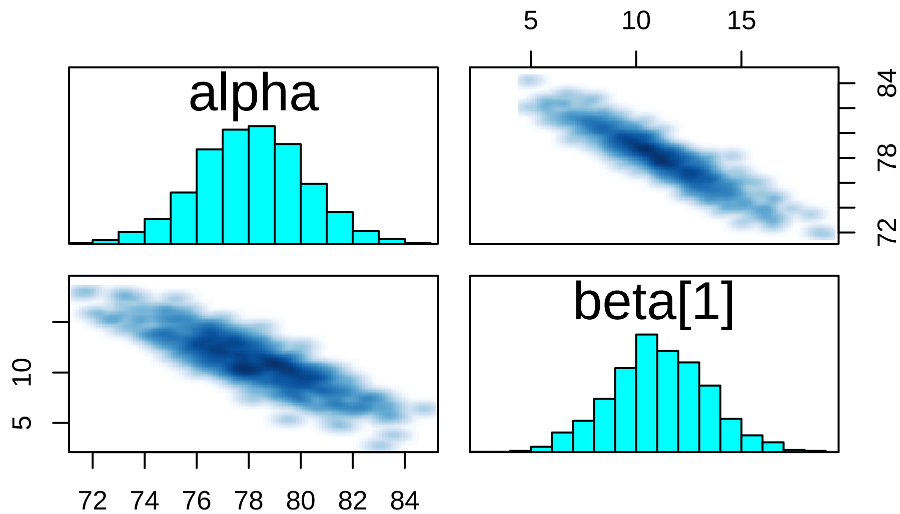
a) Confirm that the estimates of the intercept and slope are comparable to results from `lm()`

```
            mean    se_mean       sd      2.5%       25%      50%      75%
alpha   78.11366 0.07871710 2.047737 73.970439 76.700403 78.10278 79.50046
beta[1] 11.10123 0.08513424 2.296724  6.593139  9.602299 11.03840 12.68638
           97.5%    n_eff     Rhat
```

```
alpha   82.18714 676.7221 1.004634
beta[1] 15.74798 727.7952 1.005055


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
```

- Yes, the coefficients are comparable.

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?

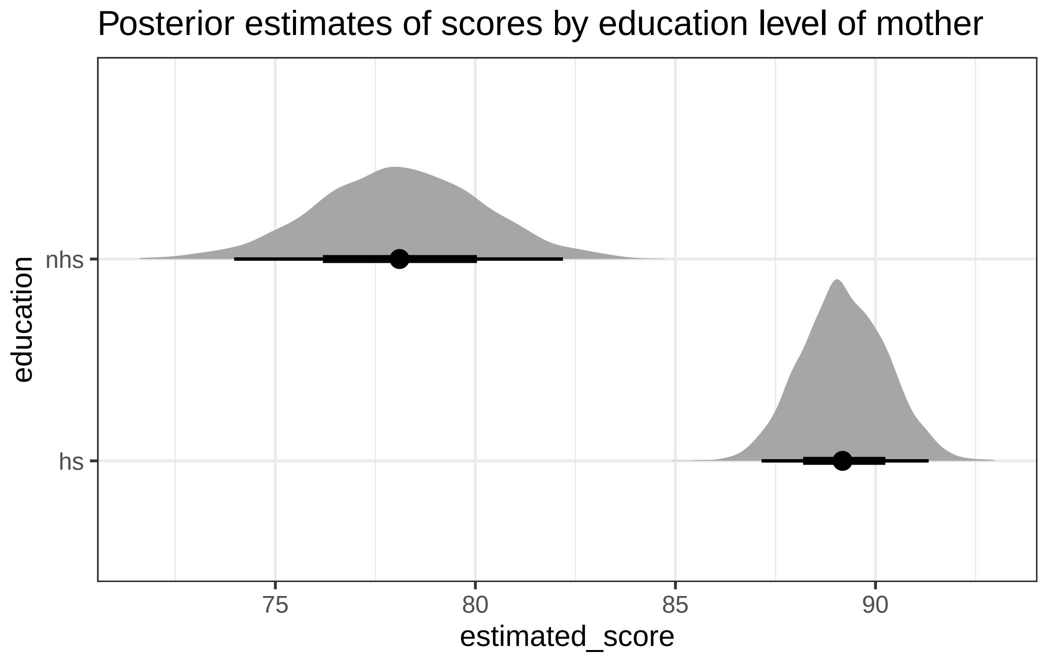


Yes, it is a problem as the joint distribution is narrow. It causes issues in our generating sample of alpha, beta values as it does not efficiently cover the sample space properly, because of the constraints of narrow line.

Plotting results

It might be nice to plot the posterior samples of the estimates for the non-high-school and high-school mothered kids. Here's some code that does this: notice the `beta[condition]` syntax. Also notice I'm using `spread_draws`, because it's easier to calculate the estimated effects in wide format

```
fit2 |>
  spread_draws(alpha, beta[k], sigma) |>
     mutate(nhs = alpha, # no high school is just the intercept
          hs = alpha + beta) |>
  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")
```



– posterior distribution is much higher.

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.

```
y <- kidiq$kid_score
dfx <- kidiq[,2:3]
dfx['mom_iq_cent'] <- dfx$mom_iq- mean(dfx$mom_iq)
dfx <- dfx |> select(-c('mom_iq'))

# named list to input for stan function
dataQ4 <- list(y = y,
               N = length(y),
               K = 2,
               X = as.matrix(dfx,ncol = 2),
               sigma0 = sigma0)
```

```
              mean     se_mean         sd       2.5%        25%        50%
alpha   82.3177673 0.096193125 1.85075019 78.8775043 81.0822780 82.1928188
```

```
beta[1]  5.7003934 0.106270632 2.09765551  1.3492342  4.3913888  5.7904947
beta[2]  0.5689272 0.002580186 0.05904928  0.4349935  0.5311117  0.5700492
               75%      97.5%    n_eff      Rhat
alpha   83.4420767 86.2087575 370.1754 1.0042938
beta[1]  7.1675566  9.5761776 389.6205 1.0041593
beta[2]  0.6103839  0.6732689 523.7538 0.9978399
```

- Intercept value is 82.29. It means that if a mother has mean iq level and does not have high school degree then the iq score of kid would be 82.29.
- beta[1] is the coefficient of mom's high school education. The coefficient estimate of 5.74 means that the kids' iq score changes by 5.74 for mothers who differed in high school degree completion for the same iq level of mother.
- beta[2] is the coefficient of mom's iq. The coefficient estimate of 0.57 means that on comparing children with the same value of mom's high school education, but whose mothers differ by 1 point in IQ, we would expect to see a difference of 0.57 points in the child's test score.

Question 5

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

```
Call:
lm(formula = kid_score ~ mom_hs + 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) 25.73154    5.87521   4.380 1.49e-05 ***
mom_hs       5.95012    2.21181   2.690  0.00742 **
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
```

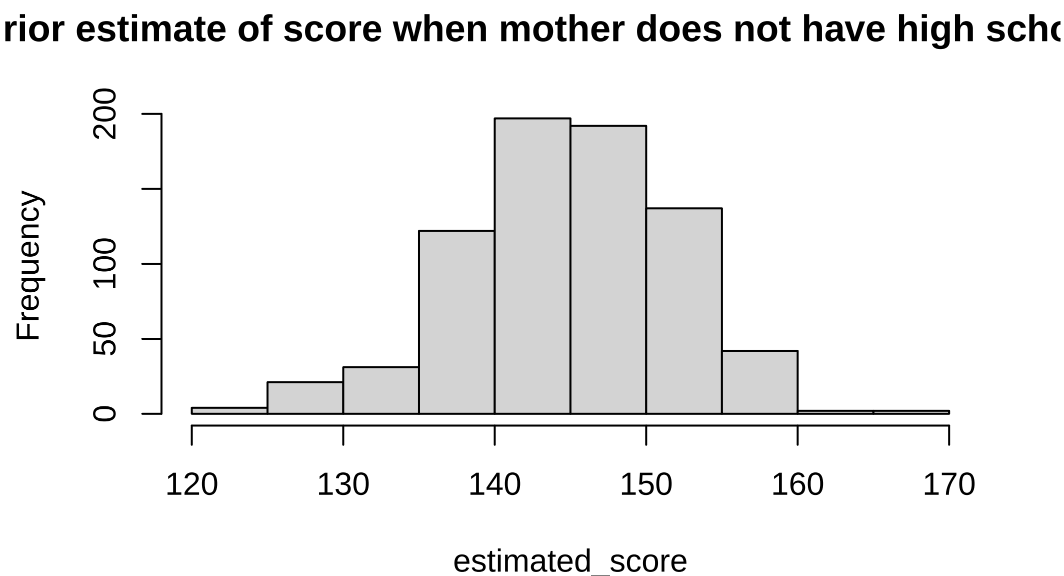
- the estimates of beta[1] and beta[2] are comparable with lm().
- the estimate of the intercept differs by beta[2]*mean(kidiq$mom_iq), which is expected as the values of mom's iq are not centered around the average in the linear model.

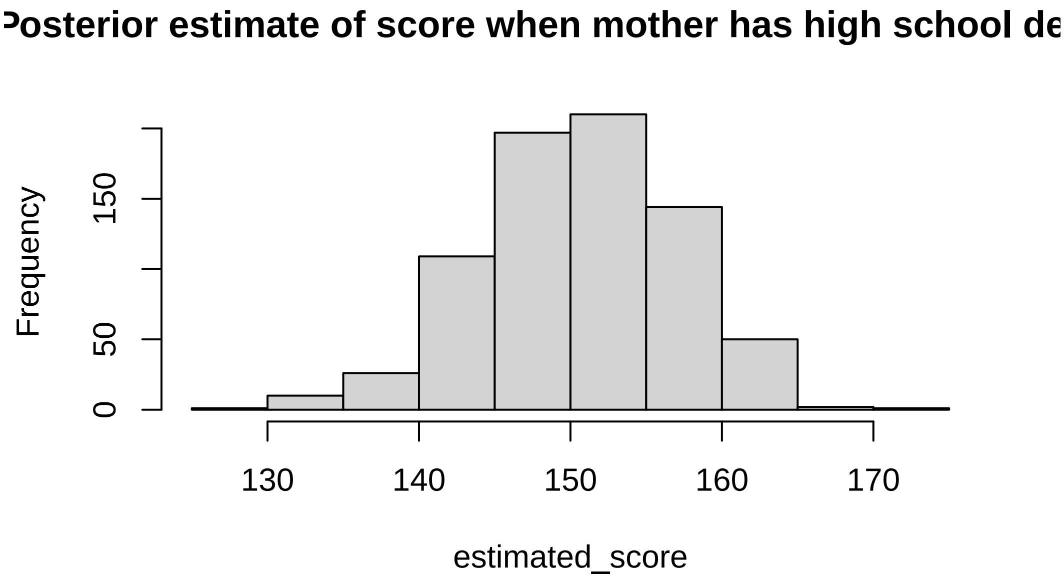
```
[1] 56.391
```

Question 6

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

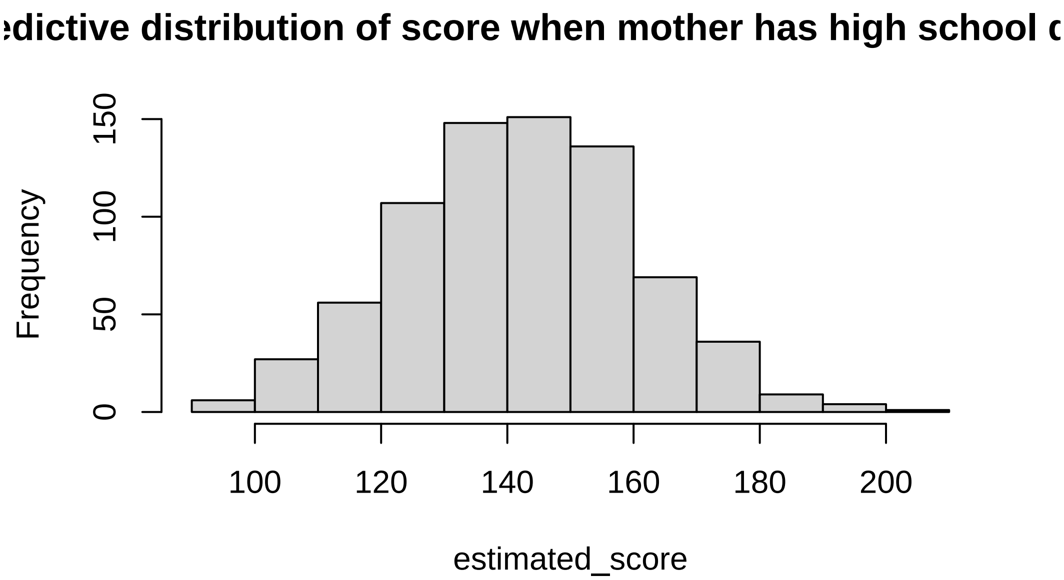
```
[1] "alpha" "beta"  "sigma" "lp__"
```





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.



edictive distribution of score when mother has high school c
Frequency
0
50
100
150
100
120
140
160
180
200
estimated_score