p8105_hw6_yl4606

Yubei Liang

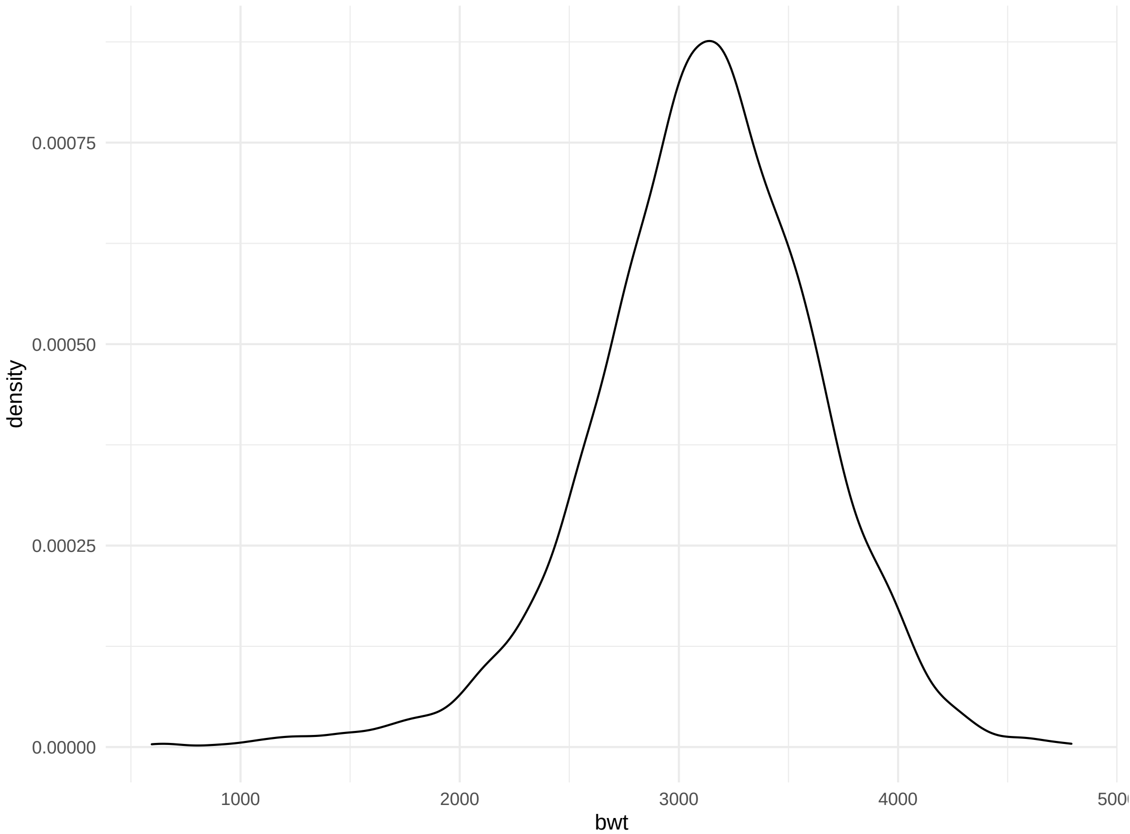
12/5/2020

Problem 2

tidy the data

```
birthweight_df =
  read_csv("data/birthweight.csv", na = c("", "NA", "Unknown")) %>%
  mutate(babysex = factor(babysex),
         frace = factor(frace),
         malform = factor(malform),
         mrace = factor(mrace))
```

```
birthweight_df %>%
  ggplot(aes(x = bwt)) +
  geom_density()
```



The density graph of birthweight has a bell shape, so no need for log transformation.

Backward Elimination

```
mult.fit <- lm(bwt ~ ., data=birthweight_df)
step(mult.fit, direction='backward')
```

```
## Start:  AIC=48717.83
## bwt ~ babysex + bhead + blength + delwt + fincome + frace + gaweeks +
##     malform + menarche + mheight + momage + mrace + parity +
##     pnumlbw + pnumsga + ppbmi + ppwt + smoken + wtgain
##
##
## Step:  AIC=48717.83
## bwt ~ babysex + bhead + blength + delwt + fincome + frace + gaweeks +
##     malform + menarche + mheight + momage + mrace + parity +
##     pnumlbw + pnumsga + ppbmi + ppwt + smoken
##
##
## Step:  AIC=48717.83
## bwt ~ babysex + bhead + blength + delwt + fincome + frace + gaweeks +
##     malform + menarche + mheight + momage + mrace + parity +
##     pnumlbw + ppbmi + ppwt + smoken
##
##
## Step:  AIC=48717.83
## bwt ~ babysex + bhead + blength + delwt + fincome + frace + gaweeks +
##     malform + menarche + mheight + momage + mrace + parity +
##     ppbmi + ppwt + smoken
##
##            Df Sum of Sq       RSS   AIC
## - frace     4    124365 320848704 48712
## - malform   1      1419 320725757 48716
## - ppbmi     1      6346 320730684 48716
## - momage    1     28661 320752999 48716
## - mheight   1     66886 320791224 48717
## - menarche  1    111679 320836018 48717
## - ppwt      1    131132 320855470 48718
## <none>                  320724338 48718
## - fincome   1    193454 320917792 48718
## - parity    1    413584 321137922 48721
## - mrace     3    868321 321592659 48724
## - babysex   1    853796 321578134 48727
## - gaweeks   1   4611823 325336161 48778
## - smoken    1   5076393 325800732 48784
## - delwt     1   8008891 328733230 48823
## - blength   1 102050296 422774634 49915
## - bhead     1 106535716 427260054 49961
##
## Step:  AIC=48711.51
## bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks +
##     malform + menarche + mheight + momage + mrace + parity +
##     ppbmi + ppwt + smoken
##
##            Df Sum of Sq       RSS   AIC
## - malform   1      1447 320850151 48710
```

```
## - ppbmi     1      6975 320855679 48710
## - momage    1     28379 320877083 48710
## - mheight   1     69502 320918206 48710
## - menarche  1    115708 320964411 48711
## - ppwt      1    133961 320982665 48711
## <none>                 320848704 48712
## - fincome   1    194405 321043108 48712
## - parity    1    414687 321263390 48715
## - babysex   1    852133 321700837 48721
## - gaweeks   1   4625208 325473911 48772
## - smoken    1   5036389 325885093 48777
## - delwt     1   8013099 328861802 48817
## - mrace     3  13540415 334389119 48885
## - blength   1 101995688 422844392 49908
## - bhead     1 106662962 427511666 49956
##
## Step:  AIC=48709.53
## bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks +
##     menarche + mheight + momage + mrace + parity + ppbmi + ppwt +
##     smoken
##
##            Df Sum of Sq       RSS   AIC
## - ppbmi     1      6928 320857079 48708
## - momage    1     28660 320878811 48708
## - mheight   1     69320 320919470 48708
## - menarche  1    116027 320966177 48709
## - ppwt      1    133894 320984044 48709
## <none>                 320850151 48710
## - fincome   1    193784 321043934 48710
## - parity    1    414482 321264633 48713
## - babysex   1    851279 321701430 48719
## - gaweeks   1   4624003 325474154 48770
## - smoken    1   5035195 325885346 48775
## - delwt     1   8029079 328879230 48815
## - mrace     3  13553320 334403471 48883
## - blength   1 102009225 422859375 49906
## - bhead     1 106675331 427525481 49954
##
## Step:  AIC=48707.63
## bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks +
##     menarche + mheight + momage + mrace + parity + ppwt + smoken
##
##            Df Sum of Sq       RSS   AIC
## - momage    1     29211 320886290 48706
## - menarche  1    117635 320974714 48707
## <none>                 320857079 48708
## - fincome   1    195199 321052278 48708
## - parity    1    412984 321270064 48711
## - babysex   1    850020 321707099 48717
## - mheight   1   1078673 321935752 48720
## - ppwt      1   2934023 323791103 48745
## - gaweeks   1   4621504 325478583 48768
## - smoken    1   5039368 325896447 48773
## - delwt     1   8024939 328882018 48813
```

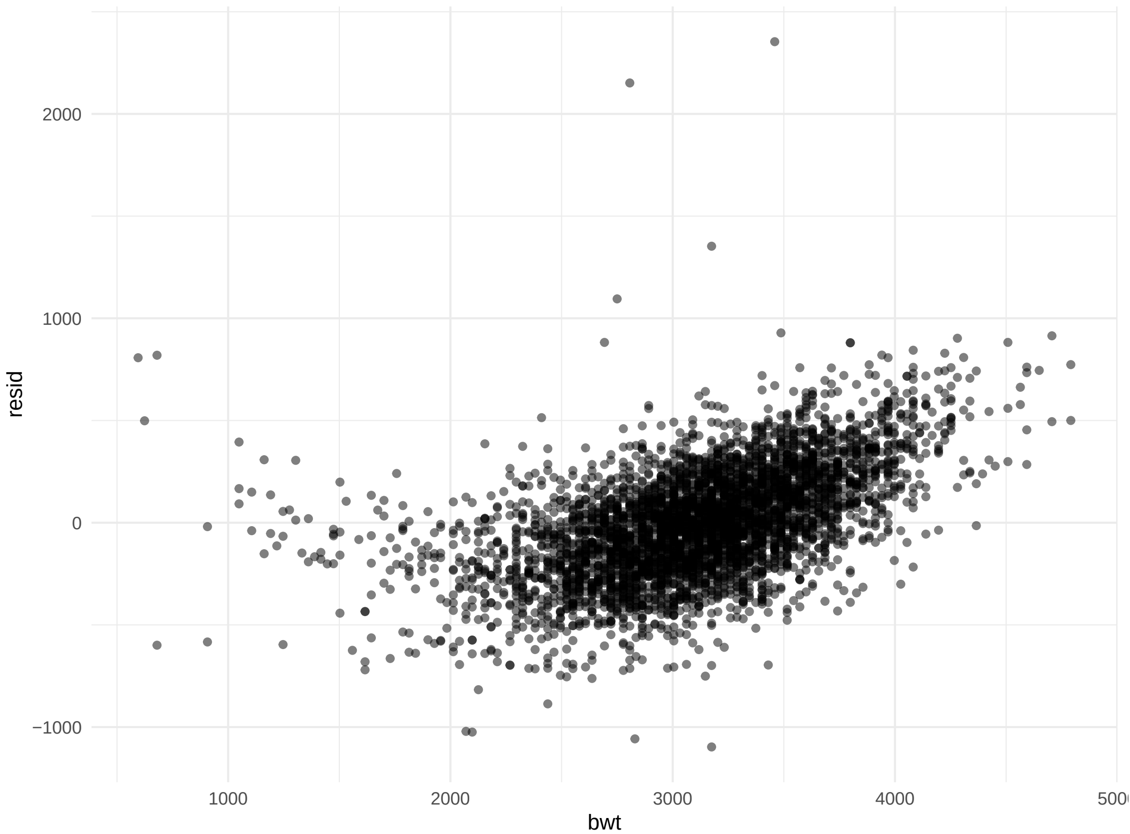
```
## - mrace     3  13551444 334408523 48881
## - blength   1 102018559 422875638 49904
## - bhead     1 106821342 427678421 49953
##
## Step:  AIC=48706.02
## bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks +
##     menarche + mheight + mrace + parity + ppwt + smoken
##
##            Df Sum of Sq       RSS   AIC
## - menarche  1    100121 320986412 48705
## <none>                  320886290 48706
## - fincome   1    240800 321127090 48707
## - parity    1    431433 321317724 48710
## - babysex   1    841278 321727568 48715
## - mheight   1   1076739 321963029 48719
## - ppwt      1   2913653 323799943 48743
## - gaweeks   1   4676469 325562760 48767
## - smoken    1   5045104 325931394 48772
## - delwt     1   8000672 328886962 48811
## - mrace     3  14667730 335554021 48894
## - blength   1 101990556 422876847 49902
## - bhead     1 106864308 427750598 49952
##
## Step:  AIC=48705.38
## bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks +
##     mheight + mrace + parity + ppwt + smoken
##
##           Df Sum of Sq       RSS   AIC
## <none>                 320986412 48705
## - fincome  1    245637 321232048 48707
## - parity   1    422770 321409181 48709
## - babysex  1    846134 321832545 48715
## - mheight  1   1012240 321998651 48717
## - ppwt     1   2907049 323893461 48743
## - gaweeks  1   4662501 325648912 48766
## - smoken   1   5073849 326060260 48771
## - delwt    1   8137459 329123871 48812
## - mrace    3  14683609 335670021 48894
## - blength  1 102191779 423178191 49903
## - bhead    1 106779754 427766166 49950

##
## Call:
## lm(formula = bwt ~ babysex + bhead + blength + delwt + fincome +
##     gaweeks + mheight + mrace + parity + ppwt + smoken, data = birthweight_df)
##
## Coefficients:
## (Intercept)     babysex2        bhead      blength        delwt      fincome
##   -6098.822       28.558      130.777       74.947        4.107        0.318
##     gaweeks      mheight       mrace2       mrace3       mrace4       parity
##      11.592        6.594     -138.792      -74.887     -100.678       96.305
##        ppwt       smoken
##      -2.676       -4.843
```

I use backward elimination method, which removes variables that have large p-value one by one from the original full model and refit. Until all variables are significant, the process completes. Therefore, I derive the model: **bwt ~ babysex + bhead + blength + delwt + fincome + gaweeks + mheight + mrace + parity + ppwt + smoken** with coefficients listed below:

```
##
## Call:
## lm(formula = bwt ~ babysex + bhead + blength + delwt + fincome +
##     gaweeks + mheight + mrace + parity + ppwt + smoken, data = birthweight_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1097.18  -185.52    -3.39   174.14  2353.44
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6098.8219   137.5463 -44.340  < 2e-16 ***
## babysex2       28.5580     8.4549   3.378 0.000737 ***
## bhead         130.7770     3.4466  37.944  < 2e-16 ***
## blength        74.9471     2.0190  37.120  < 2e-16 ***
## delwt           4.1067     0.3921  10.475  < 2e-16 ***
## fincome         0.3180     0.1747   1.820 0.068844 .
## gaweeks        11.5925     1.4621   7.929 2.79e-15 ***
## mheight         6.5940     1.7849   3.694 0.000223 ***
## mrace2       -138.7925     9.9071 -14.009  < 2e-16 ***
## mrace3        -74.8868    42.3146  -1.770 0.076837 .
## mrace4       -100.6781    19.3247  -5.210 1.98e-07 ***
## parity         96.3047    40.3362   2.388 0.017004 *
## ppwt           -2.6756     0.4274  -6.261 4.20e-10 ***
## smoken         -4.8434     0.5856  -8.271  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 272.3 on 4328 degrees of freedom
## Multiple R-squared:  0.7181, Adjusted R-squared:  0.7173
## F-statistic: 848.1 on 13 and 4328 DF,  p-value: < 2.2e-16
```

Residual Plot

```r
birthweight_df %>%
  modelr::add_residuals(my_reg) %>%
  ggplot(aes(x = bwt, y = resid)) +
  geom_point(alpha = 0.5)
```



The residuals have a positive linear pattern.

Compare Models

```r
reg_1 <- lm(bwt ~ blength + gaweeks, data = birthweight_df)
reg_2 <- lm(bwt ~ bhead * blength * babysex, data = birthweight_df)
```

```r
summary(reg_1)
```

```
##
## Call:
## lm(formula = bwt ~ blength + gaweeks, data = birthweight_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1709.6  -215.4   -11.4   208.2  4188.8
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4347.667     97.958  -44.38   <2e-16 ***
## blength       128.556      1.990   64.60   <2e-16 ***
## gaweeks        27.047      1.718   15.74   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 333.2 on 4339 degrees of freedom
## Multiple R-squared:  0.5769, Adjusted R-squared:  0.5767
## F-statistic:  2958 on 2 and 4339 DF,  p-value: < 2.2e-16
```

```r
summary(reg_2)
```

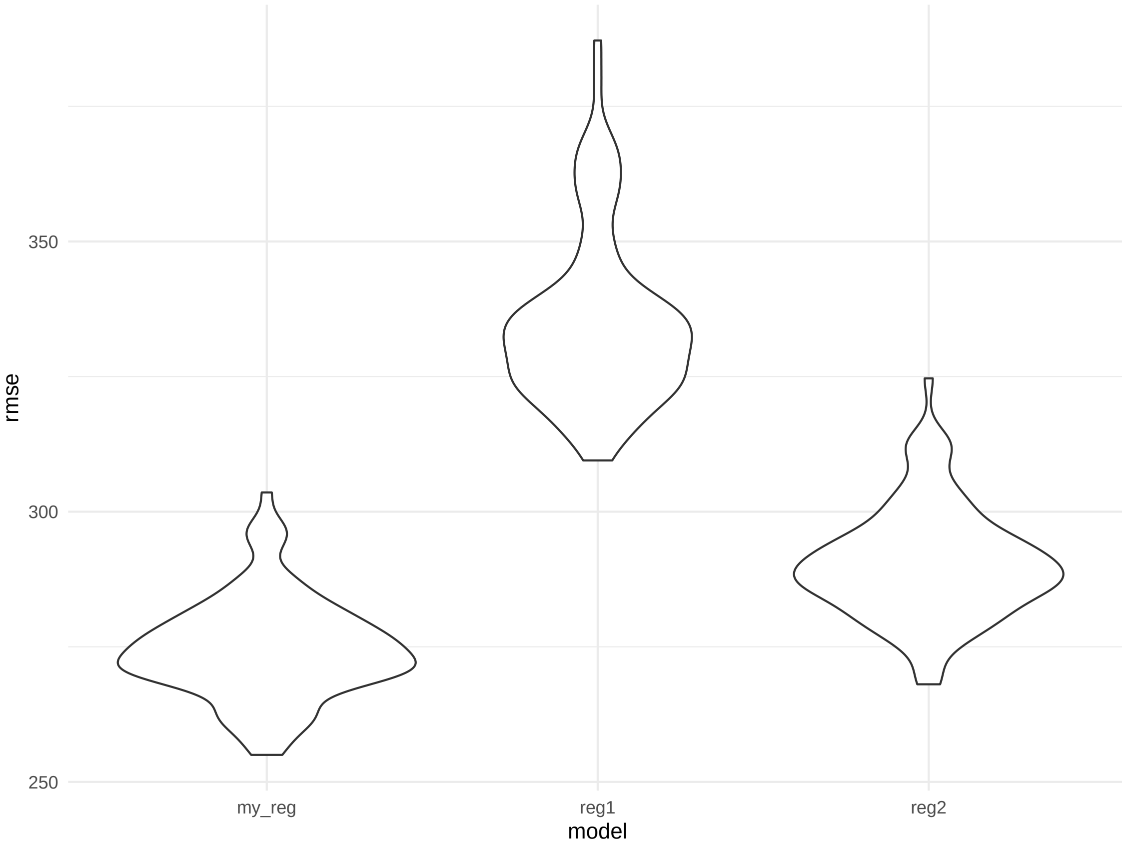
```
##
## Call:
## lm(formula = bwt ~ bhead * blength * babysex, data = birthweight_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1132.99  -190.42   -10.33   178.63  2617.96
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -7176.8170  1264.8397  -5.674 1.49e-08 ***
## bhead                   181.7956    38.0542   4.777 1.84e-06 ***
## blength                 102.1269    26.2118   3.896 9.92e-05 ***
## babysex2               6374.8684  1677.7669   3.800 0.000147 ***
## bhead:blength            -0.5536     0.7802  -0.710 0.478012
## bhead:babysex2         -198.3932    51.0917  -3.883 0.000105 ***
## blength:babysex2       -123.7729    35.1185  -3.524 0.000429 ***
## bhead:blength:babysex2    3.8781     1.0566   3.670 0.000245 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 287.7 on 4334 degrees of freedom
## Multiple R-squared:  0.6849, Adjusted R-squared:  0.6844
## F-statistic:  1346 on 7 and 4334 DF,  p-value: < 2.2e-16
```

As we can see from the summary of all three models, R^2 of model I generated using BIC is 0.7173, compared with 0.5767 and 0.6844 of models given. So the model generated using BIC could explain more about birthweight by variables selected than other two models.

Cross Validation

```r
cv_df =
  crossv_mc(birthweight_df, 100)

cv_df =
  cv_df %>%
  mutate(
    train = map(train, as_tibble),
    test = map(test, as_tibble))

cv_df =
  cv_df %>%
  mutate(
    my_reg_mod  = map(train, ~lm(bwt ~ babysex + bhead + blength + delwt + fincome +
    gaweeks + mheight + mrace + parity + ppwt + smoken, data = .x)),
    reg_1_mod  = map(train, ~lm(bwt ~ blength + gaweeks, data = .x)),
    reg_2_mod  = map(train, ~lm(bwt ~ bhead * blength * babysex, data = .x))) %>%
  mutate(
    rmse_my_reg = map2_dbl(my_reg_mod, test, ~rmse(model = .x, data = .y)),
    rmse_reg1 = map2_dbl(reg_1_mod, test, ~rmse(model = .x, data = .y)),
    rmse_reg2 = map2_dbl(reg_2_mod, test, ~rmse(model = .x, data = .y)))
```

```r
cv_df %>%
  select(starts_with("rmse")) %>%
  pivot_longer(
    everything(),
    names_to = "model",
    values_to = "rmse",
    names_prefix = "rmse_") %>%
  mutate(model = fct_inorder(model)) %>%
  ggplot(aes(x = model, y = rmse)) + geom_violin()
```



The plot shows the distribution of RMSE values for each candidate model. The RMSE distribution of my regression model(BIC method) is the smallest among three models, which suggests that residuals are less spread out in this model. Thus, more accurate predictions would be made.

Problem 3

Bootstrapping

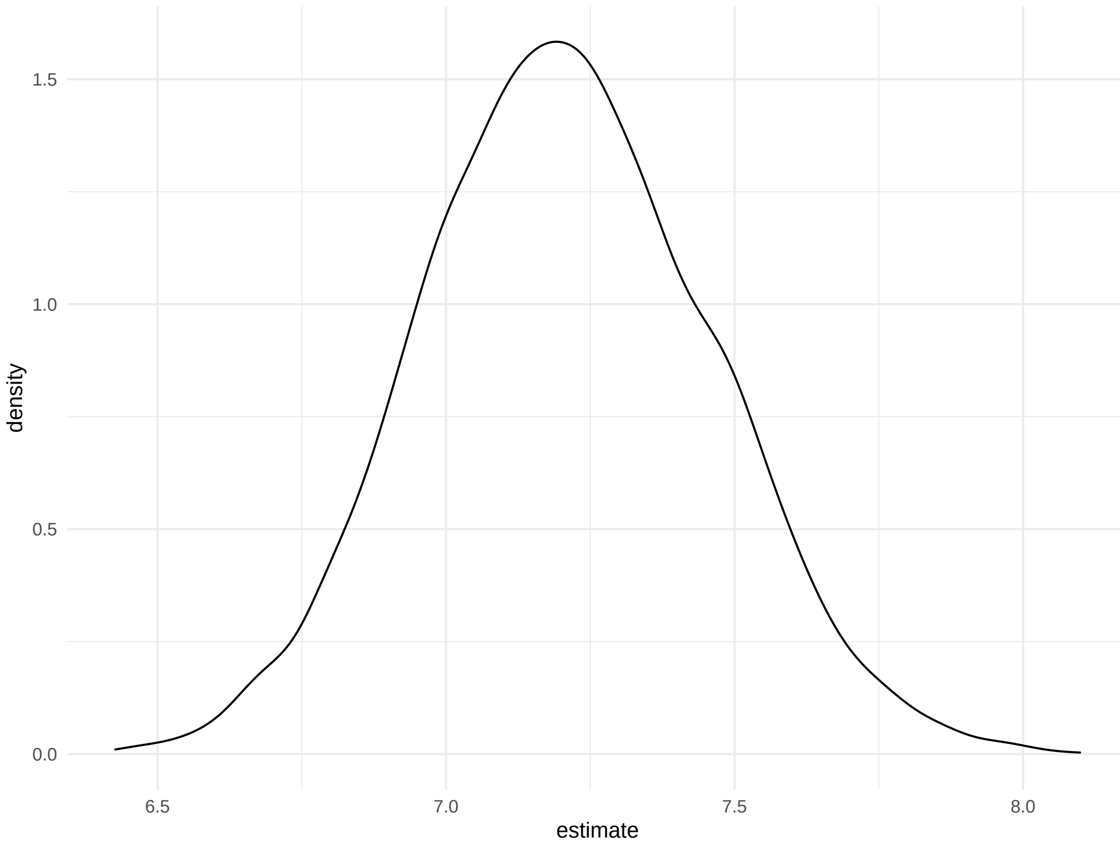
```r
bootstrap_df =
  weather_df %>%
  modelr::bootstrap(5000, id = "strap_number")
```

Unnest results of 5000 models:

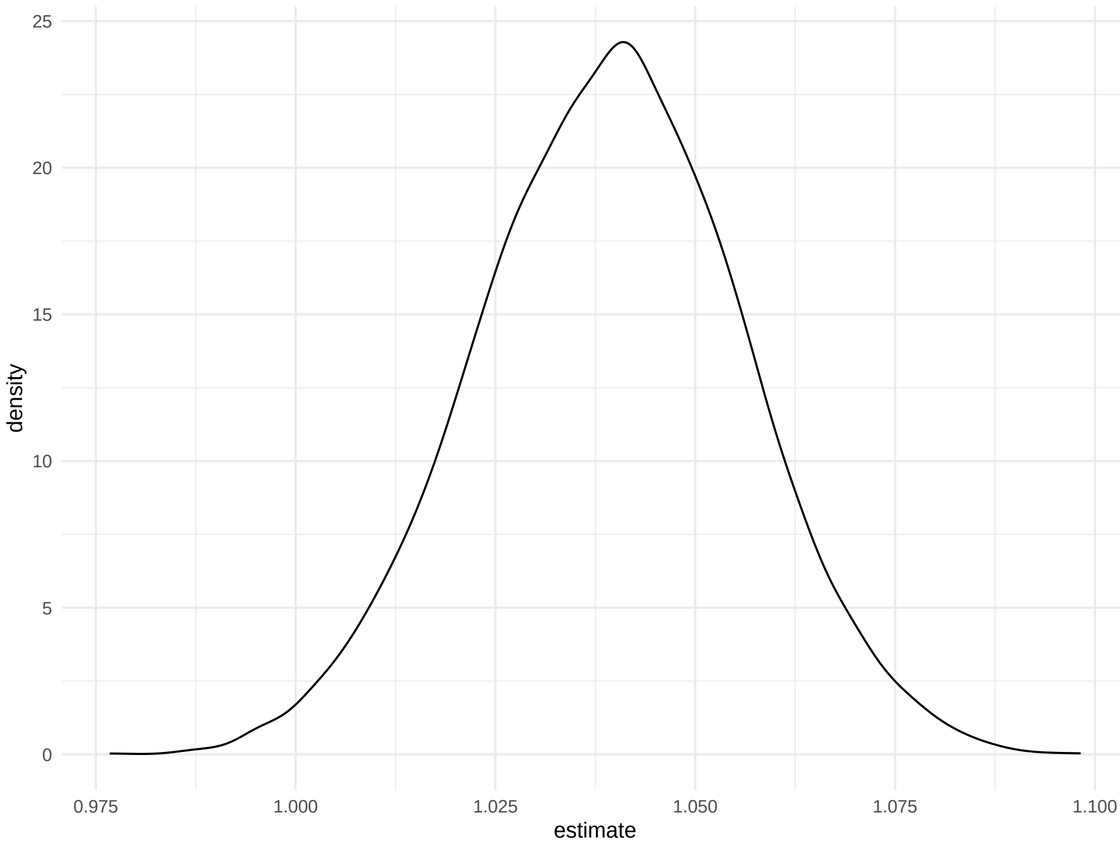
```r
unnest_df = bootstrap_df %>%
  mutate(
    models = map(.x = strap, ~lm(tmax ~ tmin, data = .x)),
```

```r
    results = map(models, tidy)) %>%
  select(strap_number, results) %>%
  unnest(results)
```

```r
unnest_df %>%
  filter(term == '(Intercept)') %>%
  ggplot(aes(x = estimate)) + geom_density()
```




```r
unnest_df %>%
  filter(term == 'tmin') %>%
  ggplot(aes(x = estimate)) + geom_density()
```



The distribution of both estimates β_0 and β_1 follows approximately normal distribution. The distribution of intercept estimate is a little skewed to the left, which may be related to the frequency with which large outliers are included in the bootstrap sample.

Cleaned dataframe with r^2 and $log(\beta_0 * \beta_1)$ extracted:

```
bootstrap_clean_df =
  bootstrap_df %>%
  mutate(
    models = map(.x = strap, ~lm(tmax ~ tmin, data = .x)),
    results_r = map(models, glance),
    results_beta = map(models, tidy)
    ) %>%
  select(strap_number, results_r, results_beta) %>%
  unnest(results_r, results_beta) %>%
  janitor::clean_names() %>%
  select(strap_number, adj_r_squared, term, estimate) %>%
  mutate(
    term = replace(term, term == '(Intercept)', 'intercept')) %>%
  pivot_wider(
    names_from = 'term',
    values_from = 'estimate'
  ) %>%
  mutate(
    beta_1 = intercept,
    beta_2 = tmin,
    log_b1_b2 = log10(beta_1*beta_2)) %>%
```

```r
  select(-intercept, -tmin)

head(bootstrap_clean_df)
```

```
## # A tibble: 6 x 5
##   strap_number adj_r_squared beta_1 beta_2 log_b1_b2
##   <chr>                <dbl>  <dbl>  <dbl>     <dbl>
## 1 0001                 0.899   7.53   1.03     0.888
## 2 0002                 0.909   7.36   1.05     0.887
## 3 0003                 0.912   6.83   1.05     0.855
## 4 0004                 0.924   6.91   1.07     0.868
## 5 0005                 0.919   6.79   1.06     0.856
## 6 0006                 0.917   7.11   1.03     0.865
```

Get 2.5% and 97.5% Quantiles

```r
r_sq = bootstrap_clean_df$adj_r_squared
quantile(r_sq, c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.8941662 0.9267956
```

```r
log = bootstrap_clean_df$log_b1_b2
quantile(log, c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.8535355 0.8943138
```

The 95% confidence interval for r^2 is (0.8937, 0.9273).

The 95% confidence interval for $log(\beta_0 * \beta_1)$ is (0.8530, 0.8945).