Problem set 7

Dr. McNamara

Exercises to hand in: 4.3 (modified d), 4.8, 4.14, 4.18, 4.20 parts a-c, e, modified g (no d or f).

4.3 Major League Baseball winning percentage

```
data(MLBStandings2016)
MLBStandings2016 <- MLBStandings2016 %>%
  select(-Wins, -Losses, -Team) # to help you get started
```

a. Forward selection

```
forward <- regsubsets(WinPct ~ ., data = MLBStandings2016, nbest = 1, nvmax = 4, method =
with(summary(forward), data.frame(cp, outmat))
```

```
              cp LeagueNL BattingAverage Runs Hits HR Doubles Triples RBI SB
1  ( 1 ) 73.42217
2  ( 1 ) 27.74522                              *
3  ( 1 ) 14.51801                              *
4  ( 1 ) 11.10941                              *
         OBP SLG ERA HitsAllowed Walks StrikeOuts Saves WHIP
1  ( 1 )           *
2  ( 1 )           *
3  ( 1 )           *                                    *
4  ( 1 )           *                                    *    *
```

```
m1 <- lm(WinPct~Runs+ERA+Saves+WHIP, data = MLBStandings2016)
summary(m1)
```

```
Call:
lm(formula = WinPct ~ Runs + ERA + Saves + WHIP, data = MLBStandings2016)

Residuals:
      Min        1Q    Median        3Q       Max
-0.051472 -0.017986 -0.001991  0.017048  0.047963

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.160e-01  1.186e-01   4.351 0.000201 ***
Runs         5.187e-04  7.764e-05   6.681 5.31e-07 ***
ERA         -3.636e-02  2.626e-02  -1.385 0.178402
Saves        2.643e-03  6.788e-04   3.893 0.000652 ***
WHIP        -2.658e-01  1.275e-01  -2.085 0.047457 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02404 on 25 degrees of freedom
Multiple R-squared:  0.8863,    Adjusted R-squared:  0.8681
F-statistic:  48.7 on 4 and 25 DF,  p-value: 1.91e-11
```

The predictors in the model are Runs, ERA, Saves, and WHIP and the response variable is WinPct.

The R^2 value of the model is 0.8863 i.e. 88.63% of the variability is explained by the model.

b. Backward elimination

```
backward <- regsubsets(WinPct ~ ., data = MLBStandings2016, nbest = 1, nvmax = 4, method =
with(summary(backward), data.frame(cp, outmat))
```

```
              cp LeagueNL BattingAverage Runs Hits HR Doubles Triples RBI SB
1  ( 1 ) 81.87250
2  ( 1 ) 40.18614                          *
3  ( 1 ) 11.49510                          *
4  ( 1 ) 11.83185                      *   *
         OBP SLG ERA HitsAllowed Walks StrikeOuts Saves WHIP
1  ( 1 )                                             *
2  ( 1 )                                             *
3  ( 1 )                                         *   *
```

```
4  ( 1 )                                             *    *
```

```
m2 <- lm(WinPct~BattingAverage+Runs+Saves+WHIP, data = MLBStandings2016)
summary(m2)
```

```
Call:
lm(formula = WinPct ~ BattingAverage + Runs + Saves + WHIP, data = MLBStandings2016)

Residuals:
     Min       1Q   Median       3Q      Max
-0.04771 -0.01210 -0.00105  0.01774  0.04478

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     0.4641489  0.1401941   3.311  0.00283 **
BattingAverage  0.7818438  0.6840724   1.143  0.26390
Runs            0.0004269  0.0001162   3.674  0.00114 **
Saves           0.0028606  0.0006411   4.462  0.00015 ***
WHIP           -0.4489248  0.0603730  -7.436 8.68e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02432 on 25 degrees of freedom
Multiple R-squared:  0.8836,    Adjusted R-squared:  0.865
F-statistic: 47.45 on 4 and 25 DF,  p-value: 2.538e-11
```

The predictors in the model are BattingAverage, Runs, Saves, and WHIP and the response variable is WinPct.

The R^2 value of the model is 0.8836 i.e. 88.36% of the variability is explained by the model.

c. Best subsets

```
best <- regsubsets(WinPct ~ ., data = MLBStandings2016, nbest = 2, method = "exhaustive")
with(summary(best), data.frame(rsq, adjr2, cp, rss, outmat))
```

```
              rsq     adjr2        cp         rss LeagueNL BattingAverage Runs
1  ( 1 ) 0.6364939 0.6235115 73.422166 0.046165627
1  ( 2 ) 0.6055979 0.5915121 81.872500 0.050089450
```

```
2  ( 1 ) 0.8108098 0.7967957 27.745222 0.024027344                          *
2  ( 2 ) 0.8104729 0.7964339 27.837347 0.024070121
3  ( 1 ) 0.8775356 0.8634051 11.495103 0.015553095                          *
3  ( 2 ) 0.8713605 0.8565175 13.184056 0.016337343
4  ( 1 ) 0.8885387 0.8707049 10.485662 0.014155693                          *
4  ( 2 ) 0.8875897 0.8696041 10.745210 0.014276212                          *
5  ( 1 ) 0.9018830 0.8814419  8.835882 0.012460957                      *   *
5  ( 2 ) 0.9004753 0.8797410  9.220891 0.012639731                          *
6  ( 1 ) 0.9118990 0.8889161  8.096417 0.011188915                      *   *
6  ( 2 ) 0.9117563 0.8887362  8.135434 0.011207032                      *   *
7  ( 1 ) 0.9182833 0.8922825  8.350260 0.010378106                      *
7  ( 2 ) 0.9180642 0.8919937  8.410167 0.010405923                      *   *
8  ( 2 ) 0.9353134 0.9106709  5.692353 0.008215256                      *   *
8  ( 2 ) 0.9297972 0.9030532  7.201104 0.008915829                      *
         Hits HR Doubles Triples RBI SB OBP SLG ERA HitsAllowed Walks
1  ( 1 )                                          *
1  ( 2 )
2  ( 1 )                                          *
2  ( 2 )                          *               *
3  ( 1 )
3  ( 2 )                          *
4  ( 1 )              *
4  ( 2 )                                                        *
5  ( 1 )              *
5  ( 2 )              *                                         *
6  ( 1 )              *                                         *
6  ( 2 )              *                           *
7  ( 1 )    *  *                                            *     *
7  ( 2 )    *          *                                          *
8  ( 1 )    *                            *                  *     *
8  ( 2 )    *                          *       *            *     *
         StrikeOuts Saves WHIP
1  ( 1 )
1  ( 2 )                    *
2  ( 1 )
2  ( 2 )
3  ( 1 )                *    *
3  ( 2 )                *    *
4  ( 1 )                *    *
4  ( 2 )                *    *
5  ( 1 )                *    *
5  ( 2 )                *    *
6  ( 1 )                *    *
```

```
6  ( 2 )               *    *
7  ( 1 )               *    *
7  ( 2 )               *    *
8  ( 1 )               *    *
8  ( 2 )               *    *
```

The first model of size 4 would explain the most variability with a R^2 of 88.85%

The four variables included in this model are Runs, Doubles, Saves, and WHIP.

```
m3 <- lm(WinPct~Runs+Doubles+Saves+WHIP, data = MLBStandings2016)
summary(m3)
```

```
Call:
lm(formula = WinPct ~ Runs + Doubles + Saves + WHIP, data = MLBStandings2016)

Residuals:
      Min         1Q     Median         3Q        Max
-0.041565 -0.012093 -0.002165  0.014349  0.042894

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.6216851  0.1226586   5.068 3.12e-05 ***
Runs         0.0006352  0.0001040   6.110 2.19e-06 ***
Doubles     -0.0004463  0.0002841  -1.571  0.12876
Saves        0.0025272  0.0006910   3.658  0.00119 **
WHIP        -0.4277023  0.0552965  -7.735 4.32e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0238 on 25 degrees of freedom
Multiple R-squared:  0.8885,    Adjusted R-squared:  0.8707
F-statistic: 49.82 on 4 and 25 DF,  p-value: 1.486e-11
```

d. [Modified from book] Find the value of AIC for each of the models produced in (a–c).

```
AIC(m1)
```

```
[1] -132.0211
```

```
AIC(m2)
```

```
[1] -131.3324
```

```
AIC(m3)
```

```
[1] -132.6288
```

e. Which do you prefer?

The best model is model 3 i.e. the one produced by best subsets procedure since it has the lowest AIC value.

4.8 County health: cross validation

```
data("CountyHealth")
CountyHealth <- CountyHealth %>%
  mutate(TsqrtMDs = sqrt(MDs))
```

a. Train

```
part1 <- CountyHealth %>%
  slice(1:35)
m4 <- lm(TsqrtMDs~Hospitals, data = part1)
summary(m4)
```

```
Call:
lm(formula = TsqrtMDs ~ Hospitals, data = part1)

Residuals:
    Min      1Q  Median      3Q     Max
-18.582  -6.362  -2.918   8.277  23.170

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.1695     2.6915  -1.178    0.247
Hospitals     6.7853     0.5284  12.841 2.19e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.627 on 33 degrees of freedom
Multiple R-squared:  0.8332,    Adjusted R-squared:  0.8282
F-statistic: 164.9 on 1 and 33 DF,  p-value: 2.194e-14
```

If the number of community hospitals in a county increases by 1, we predict that the square root of number of medical doctors would increase by approximately 6.8.

b. Predictions

```
part2 <- CountyHealth %>%
  slice(36:53)

part2 <- part2 %>%
  mutate(yhats = predict(m4, newdata = part2))
part2 %>%
summarize(cor = cor(TsqrtMDs, yhats))
```

```
       cor
1 0.9531439
```

The cross-validation correlation is 95.31%

c. Shrinkage

```
part2 %>%
  summarize(cor = cor(TsqrtMDs, yhats)) %>%
  mutate(R2 = cor^2, shrinkage = summary(m4)$r.squared - R2)
```

```
       cor        R2  shrinkage
1 0.9531439 0.9084832 -0.0752451
```

Since the shrinkage is less than 10%, the model is effective.

In class addition: We actually did better than the testing data than on training data. This is weird! Usually, models are better for the data they were fit on.

4.14 More North Carolina births

(No data)

a. t-tests

For a baby whose mother's race is White, we would predict the weight of the baby to be 117.872 ounces. Since the p-vale is less than 0.05, we have enough evidence to suggest that this predictor is statistically significant.

We would expect the weight of the babies born to the mothers of Black race to be 7.3 ounces less than the babies born to mothers who belong to the White race. Since the p-value is less than 0.05, this difference is statistically significant.

We would expect the weight of the babies born to the mothers of Hispanic race to be 0.64 ounces more than the babies born to mothers who belong to the white race. Since the p-value is more than 0.05, this difference is not statistically significant.

We would expect the weight of the babies born to the mothers of Other races to be 0.72 ounces less than the babies born to mothers who belong to the white race. Since the p-value is more than 0.05, this difference is not statistically significant.

b. R squared

The value of R squared is 1.9%. This means that 1.9% of the variability in birth weight is explained by the model using mother's race as a predictor.

c. F-test

Even though two out of four prediction variables are not statistically significant, the overall p-value for the model is less than 0.05. This means that we can reject the null hypothesis. This means that the race of the mother is a significant predictor in determining the weight of the babies and there is a linear relationship between them.

4.18 GPA by Verbal SAT slope

```
data("SATGPA")
l1 <- lm(GPA~VerbalSAT, data = SATGPA)
summary(l1)
```

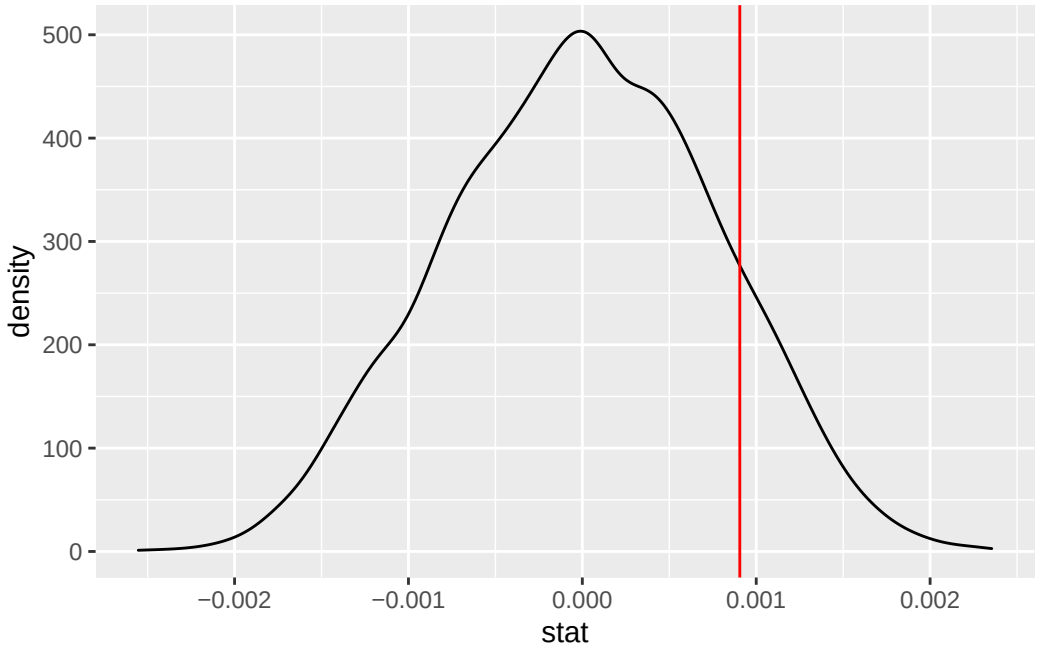
```
Call:
lm(formula = GPA ~ VerbalSAT, data = SATGPA)

Residuals:
     Min       1Q   Median       3Q      Max
-0.62002 -0.25932  0.03885  0.20502  0.51621

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.6042036  0.4377919   5.948  5.5e-06 ***
VerbalSAT   0.0009056  0.0007659   1.182     0.25
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3154 on 22 degrees of freedom
Multiple R-squared:  0.05976,   Adjusted R-squared:  0.01702
F-statistic: 1.398 on 1 and 22 DF,  p-value: 0.2496
```

```
confint(l1)
```

```
                   2.5 %      97.5 %
(Intercept)  1.6962788335 3.512128409
VerbalSAT   -0.0006826956 0.002493913
```

```
slopetest <- SATGPA %>%
  specify(response = GPA, explanatory = VerbalSAT) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 5000, type = "permute") %>%
  calculate(stat = "slope")

ggplot(data = slopetest, aes(x = stat)) +
  geom_density() +
```

```
  geom_vline(xintercept = 0.0009056, color = "red")
```




```
slopetest %>%
  get_p_value(obs_stat = 0.0009056, direction = "both")
```

```
# A tibble: 1 x 1
  p_value
    <dbl>
1   0.252
```

```
get_ci(slopetest)
```

```
# A tibble: 1 x 2
  lower_ci upper_ci
     <dbl>    <dbl>
1 -0.00150  0.00144
```

We get a p value of 0.2496 from the traditional t test and 0.2524 the randomization test. Since the p valueo btained by both the t-test and randomization is greater than 0.05, we fail to

reject the null hypothesis. Therefore, we do not have enough evidence to suggest that there is a linear relationship between GPA and VerbalSAT. For randomization, we get a confidence interval of (-0.0015, 0.0014) i.e. the reasonable values for slope if null hypothesis were true. The slope line falls within the confidence interval, this also provides evidence that the null hypothesis is true.

4.20 Bootstrapping Adirondack hikes

```
data("HighPeaks")
```

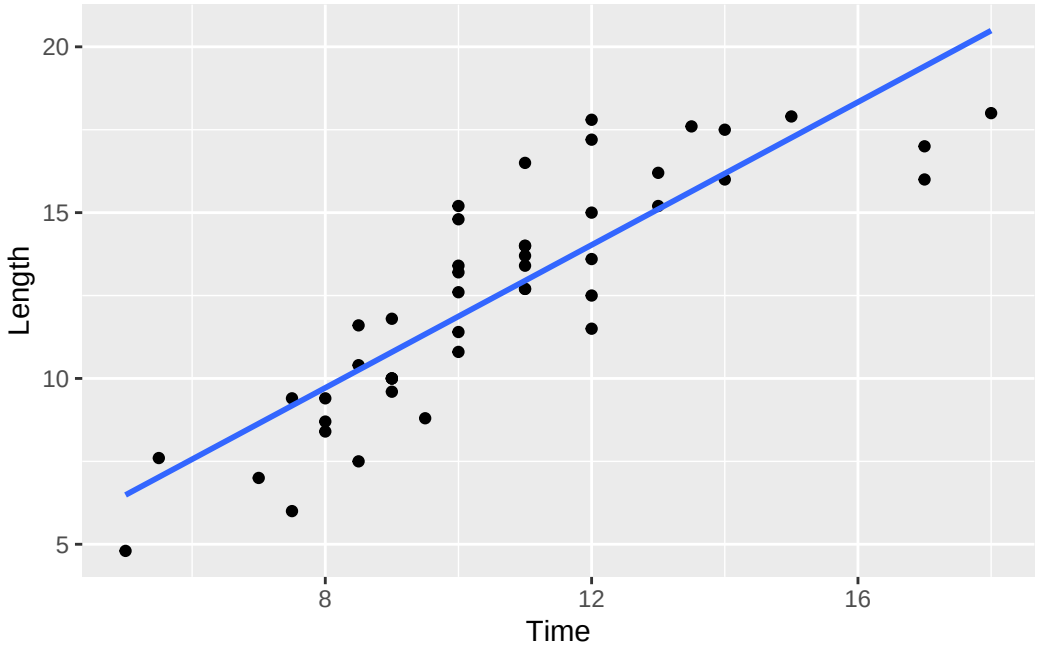
a. slr

```
k1 <- lm(Length~Time, data = HighPeaks)
summary(k1)
```

```
Call:
lm(formula = Length ~ Time, data = HighPeaks)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4112 -1.1636 -0.0413  1.0514  3.7743

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.10039    1.06739   1.031    0.308
Time         1.07711    0.09699  11.105 2.39e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.818 on 44 degrees of freedom
Multiple R-squared:  0.737, Adjusted R-squared:  0.7311
F-statistic: 123.3 on 1 and 44 DF,  p-value: 2.39e-14
```

```
ggplot(data = HighPeaks, aes(x = Time, y = Length)) + geom_point() + geom_smooth(method =
```

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



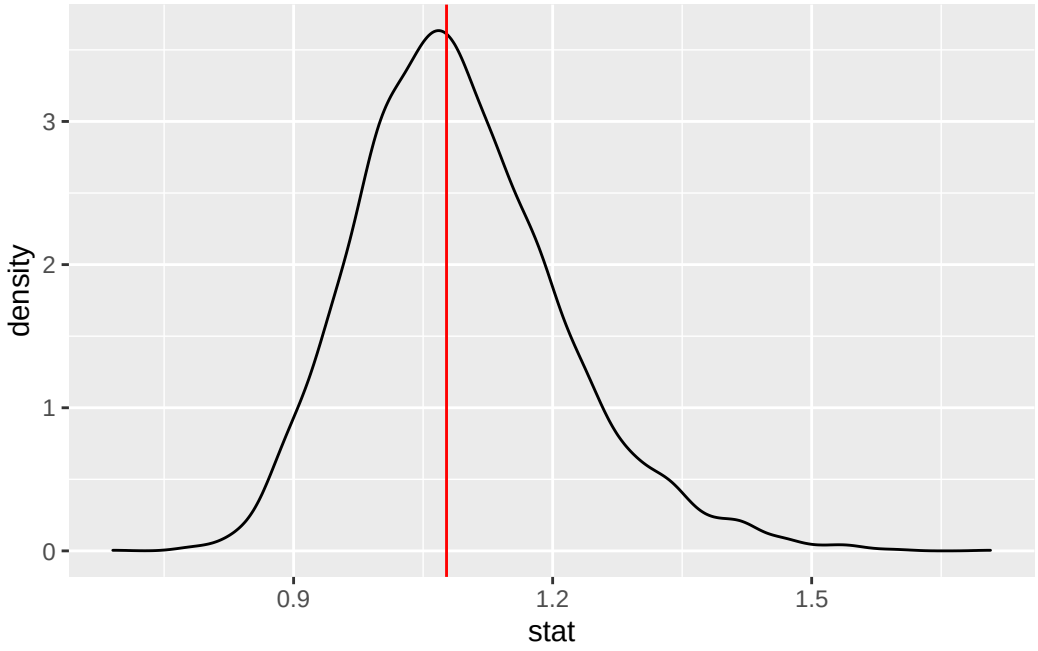

```
confint(k1, level=0.9)
```

```
                 5 %     95 %
(Intercept) -0.6930744 2.893854
Time         0.9141373 1.240075
```

We are 90% confident that for 1 hour increase in time, the average hiking speed increases between 0.91 and 1.24 miles per hour.

b. Bootstrap

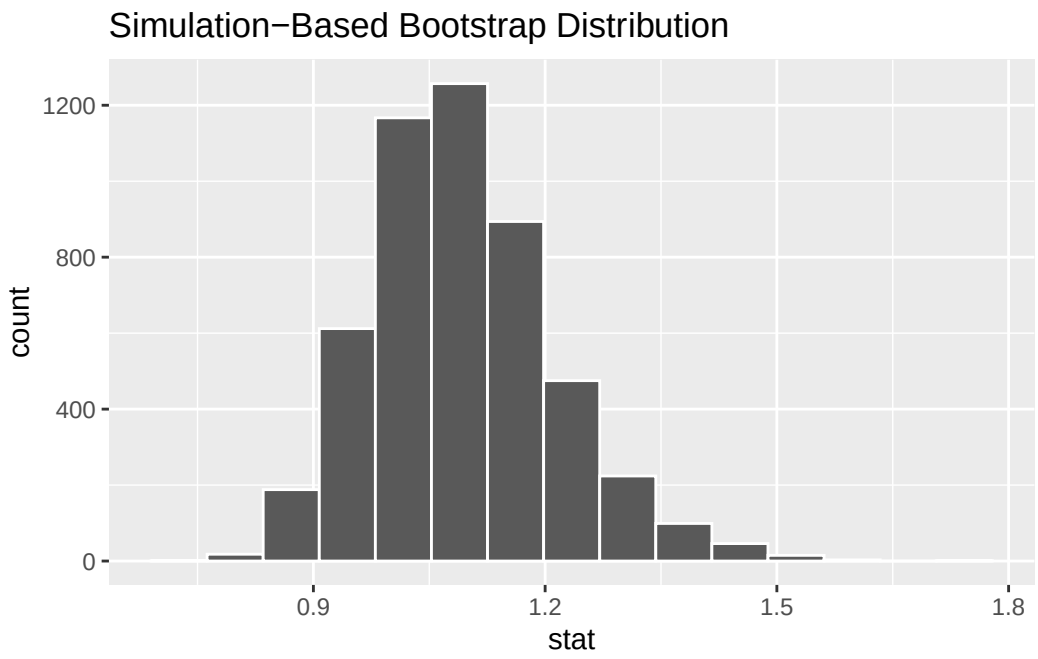
```
slopeboot <- HighPeaks %>%
  specify(response = Length, explanatory = Time) %>%
  generate(reps = 5000, type = "bootstrap") %>%
  calculate(stat = "slope")

ggplot(data = slopeboot, aes(x = stat)) +
  geom_density() +
  geom_vline(xintercept = 1.07711, color = "red")
```




```
get_ci(slopeboot)
```

```
# A tibble: 1 x 2
  lower_ci upper_ci
     <dbl>    <dbl>
1    0.890     1.36
```

```
visualize(slopeboot)
```



The distribution looks evenly distributed around the intercept line but we may say it's a bit right skewed.

Centered around the sample slope.

c. Mean and sd

```
slopeboot %>%
  summarize(SE = sd(stat), mean = mean(stat))
```

```
# A tibble: 1 x 2
     SE  mean
  <dbl> <dbl>
1 0.119  1.09
```

In the original model, estimated coefficient is 1.07 and the standard error is 0.09

In the bootstrap mode, the mean is 1.09 and the standard error is 0.12

We can see that these values are relatively close to each other.

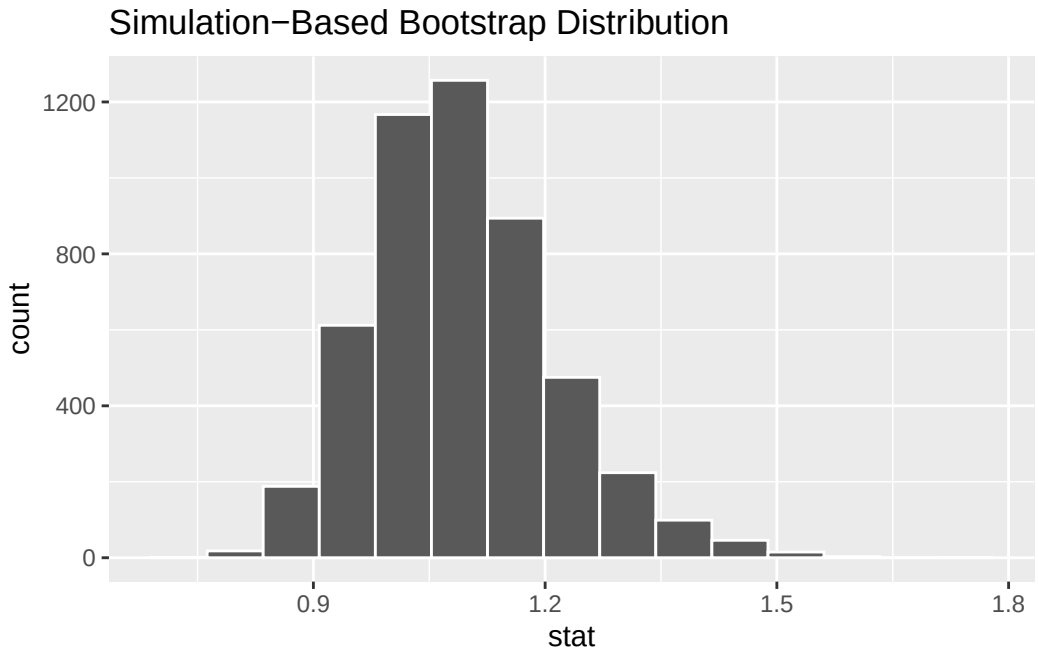
d. [skip]

e. percentile interval

```
get_ci(slopeboot, level = 0.9)
```

```
# A tibble: 1 x 2
  lower_ci upper_ci
     <dbl>    <dbl>
1    0.916     1.31
```

```
visualize(slopeboot)
```



The 5th and 95th quantiles from the bootstrap distribution is (0.9191689, 1.313717)

We are 90% confident that for 1 hour increase in time, the average hiking speed increases between 0.91 and 1.31 miles per hour.

f. [skip]

g. [Modified from book] Do you see much difference between the intervals of parts (a) and (e)?

No, both the confidence intervals obtained are relatively the same.