PS2

Audrey Glaser

1/30/2020

Question 1

```
##
## Call:
## lm(formula = biden ~ female + age + educ + dem + rep, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -75.546 -11.295   1.018  12.776  53.977
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.81126    3.12444  18.823  < 2e-16 ***
## female        4.10323    0.94823   4.327 1.59e-05 ***
## age           0.04826    0.02825   1.708   0.0877 .
## educ         -0.34533    0.19478  -1.773   0.0764 .
## dem          15.42426    1.06803  14.442  < 2e-16 ***
## rep         -15.84951    1.31136 -12.086  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.91 on 1801 degrees of freedom
## Multiple R-squared:  0.2815, Adjusted R-squared:  0.2795
## F-statistic: 141.1 on 5 and 1801 DF,  p-value: < 2.2e-16
```

```
## [1] 395.2702
```

Looking at the P-values of the coefficients (using a one-tailed t-test), the 'female' variable and both party variables are significant, but the age and education variables are not, using an alpha value of 0.01. The inclusion of two extraneous variables inflates the standard errors of the correctly specified variable's coefficients.

According to our model, a female is more likely to score Biden 4.1 points high on average on the feeling thermometer. A Democrat would score Biden 15.4 points higher than independent, on average. And a Republican would score Biden 15.9 points lower than an indendent, on average.

The estimated MSE for the model (or the mean of the squared distances between the observed Biden thermometer values and predicted thermometer values) is 395.27. If the model was a perfect fit to the observed data, the MSE would be zero.

Question 2

Split the sample set into a training set (50%) and a holdout set (50%).

```
#Set seed
set.seed(242)
```

```r
#Basic 50/50 split
nes2008_split <- initial_split(data = nes2008,
                               prop = 0.5)

#Set each half as training and test
nes2008_train <- training(nes2008_split)
nes2008_test <- testing(nes2008_split)
```

Fit the linear regression model using only the training observations.

```
## # A tibble: 6 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  61.1       4.34      14.1   8.97e-41
## 2 female        5.15      1.31       3.92  9.38e- 5
## 3 age           0.0144    0.0389     0.370 7.12e- 1
## 4 educ         -0.485     0.268     -1.81  7.05e- 2
## 5 dem          15.2       1.50      10.2   5.36e-23
## 6 rep         -14.8       1.82      -8.10  1.85e-15
```

Calculate the MSE using only the test set observations.

```
## [1] 19.62857
```

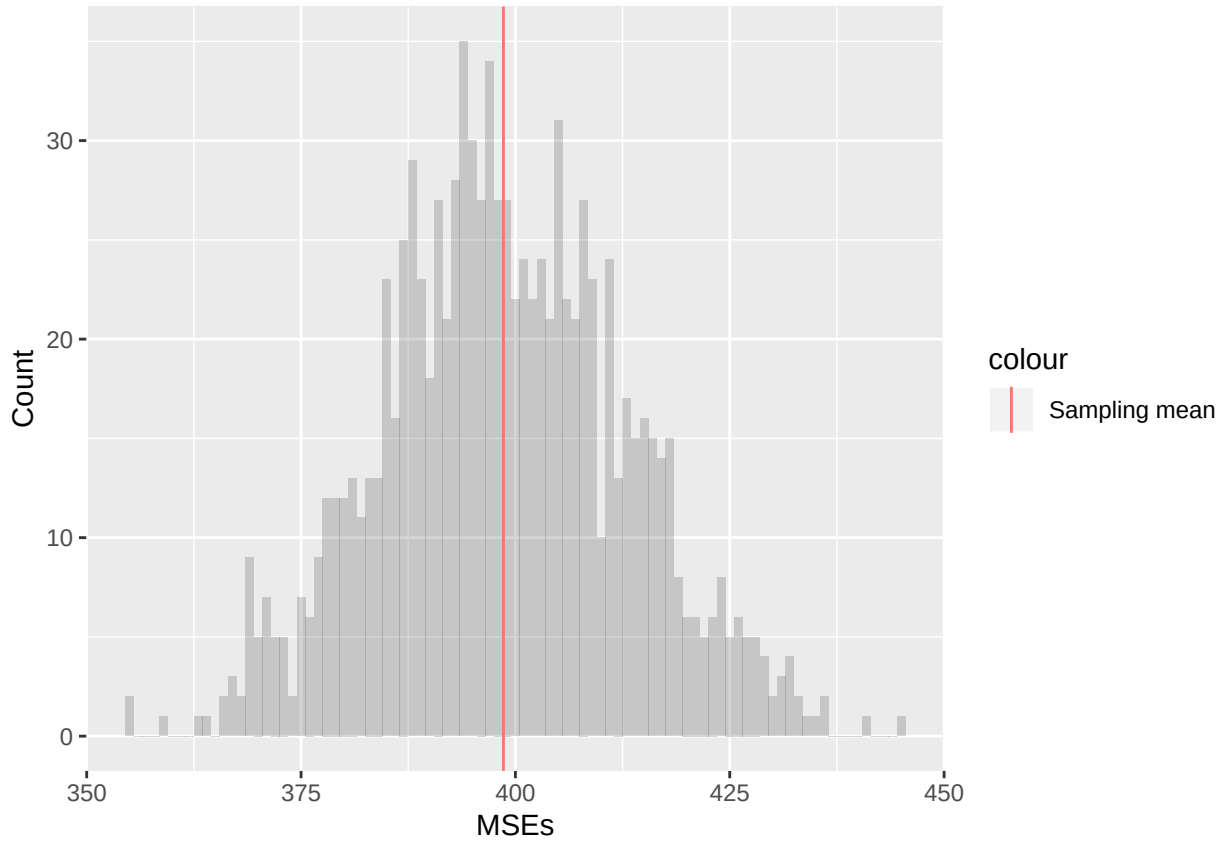
How does this value compare to the training MSE from question 1?

The MSE for model trained on just the test data is equal to 414.899, which is 19.623 units larger than the MSE of the model trained on the entire dataset.

This difference is expected, in part because the second model is trained on a smaller quantity of data than the first model, and in part because the second model is tested on out-of-sample data (data it hasn't "seen" before, that is.) We generally expect models to be overfitted to the data they're trained on, which is why holdout methods are useful to minimizing a model's reducible error.

Question 3

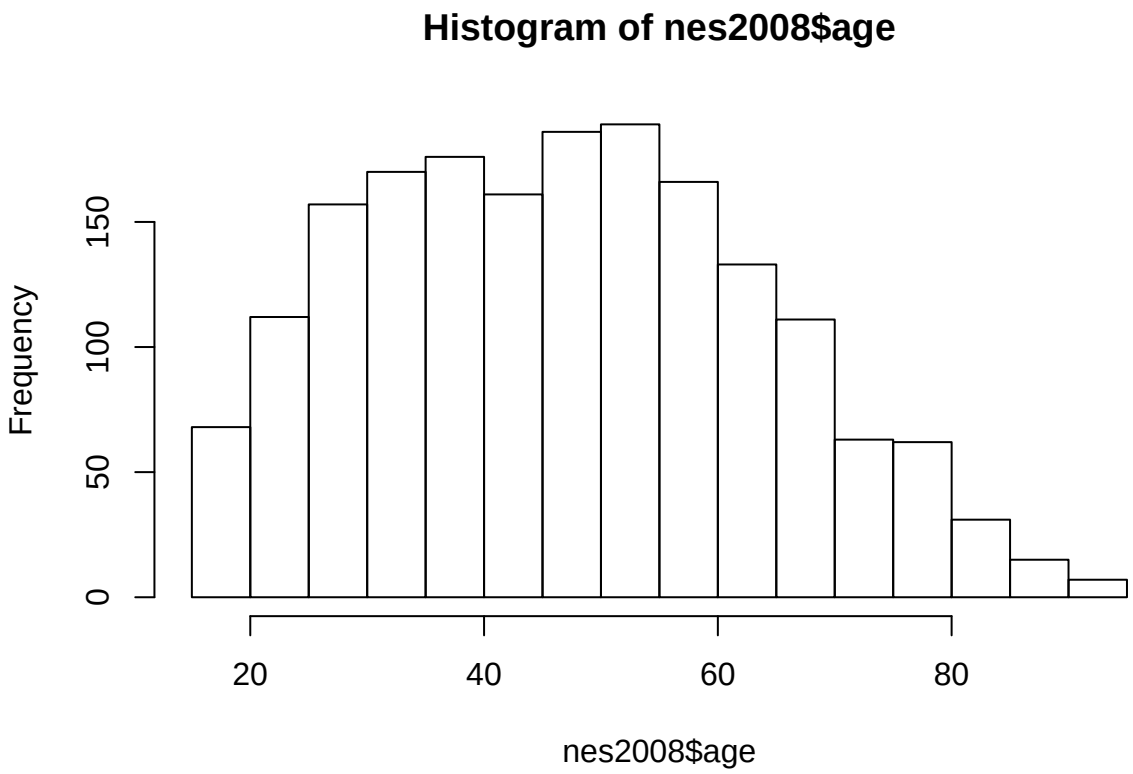
```
## [1] 398.6048
```



The test set MSEs generated from the 1,000 models centers around 398.6048, which is close to the original model's MSE of 395.2702.

Question 4

```
## # A tibble: 6 x 5
##   term        boot.estimate estimate boot.se std.error
##   <chr>               <dbl>    <dbl>   <dbl>     <dbl>
## 1 (Intercept)       58.8     58.8     2.92      3.12
## 2 female             4.11     4.10    0.963     0.948
## 3 age                0.0480   0.0483  0.0289    0.0282
## 4 educ              -0.342   -0.345   0.191     0.195
## 5 dem               15.4     15.4     1.10      1.07
## 6 rep              -15.8    -15.8     1.37      1.31
```



The beta coefficients generated by the linear model and the bootstrap model are nearly identical to each other. With regards to standard errors, the bootstrap approach generates slightly higher standard errors than the linear model, except for the intercept and the education variable. The similar estimates of both approaches makes sense because the distributional assumptions of the linear model (that the residuals are normally distributed) are valid in this case, meaning bootstrapping is unlikely to produce significantly more accurate estimates.