Problem Set 2

Grace Wright

1/30/2020

1. (10 points) Estimate the MSE of the model using the traditional approach. That is, fit the linear regression model using the entire dataset and calculate the mean squared error for the entire dataset. Present and discuss your results at a simple, high level.

```
tradlm <- lm(biden ~ female + age + educ + dem + rep, bidendata)
summary(tradlm)
```

```
##
## Call:
## lm(formula = biden ~ female + age + educ + dem + rep, data = bidendata)
##
## 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
```

```
tradmse <- modelr::mse(tradlm, bidendata)
summary(tradmse)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   395.3   395.3   395.3   395.3   395.3   395.3
```

Ideally, our mean squared error should be zero, but this mean squared error (mse = 395.27) is not. However, the model passes an F-test, which tells us that the model is picking up on some relations between some variables. Additionally, we can see that the p-values for female, dem, and rep are less than .05, thus we can say there is a statistically significant realtion for these variables as they relate to Biden, but this is not the case for the variables of age and education. The "Estimates" or coefficients tell us that for every one unit change female respondents, there should be a positive 4.103 increase in the Biden feeling thermometer. Likewise, with a one point increase in age, there is a postive .0483 increase in the Biden feeling thermometer, and

with every one point increase in "dem" there is a positive 15.424 increase in the Biden feeling thermometer. In other words, the data tells us that when someone identifies themselves as democratic there is likely to be a positive 15.424 effect on the Biden feeling thermometer. On the contrary, in the categories of "educ" and "rep" there is a negative decrease in the Biden feeling thermometer (educ = -.345, rep = -15.849)> Lastly, as can be seen in the numbers mentioned above, there is a greater effect (negative and positive) in the categories of "rep" and "dem" than in "female", "age", and "educ". This tells us that the party with which one aligns has the greatest affect on the individuals overall feelings toward Biden.

2. (30 points) Calculate the test MSE of the model using the simple holdout validation approach.

a. (5 points) Split the sample set into a training set (50%) and a holdout set (50%). Be sure to set your seed prior to this part of your code to guarantee reproducibility of results.

```
set.seed(1234)

trad_split <- initial_split(data = bidendata,
                            prop = 0.5)
trad_train <- training(trad_split)
trad_test <- testing(trad_split)
```

b. (5 points) Fit the linear regression model using only the training observations.

```
trad_trainlm <- lm(biden ~ female + age + educ + dem + rep, trad_train)
summary(trad_trainlm)
```

```
##
## Call:
## lm(formula = biden ~ female + age + educ + dem + rep, data = trad_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -75.880 -11.950   1.929  11.899  46.124
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.68937    4.30323  13.638  < 2e-16 ***
## female        4.41344    1.28889   3.424 0.000644 ***
## age           0.04460    0.03858   1.156 0.247980
## educ         -0.18263    0.26831  -0.681 0.496251
## dem          13.63872    1.45353   9.383  < 2e-16 ***
## rep         -18.76842    1.78349 -10.523  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.11 on 898 degrees of freedom
## Multiple R-squared:  0.3085, Adjusted R-squared:  0.3046
## F-statistic: 80.12 on 5 and 898 DF,  p-value: < 2.2e-16
```

c. (10 points) Calculate the MSE using only the test set observations.

```
(test_mse <- augment(tradlm, newdata = trad_test) %>%
    rcfss::mse(truth = biden, estimate = .fitted))
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 mse     standard        426.
```

d. (10 points) How does this value compare to the training MSE from question 1? Present numeric comparison and discuss a bit.

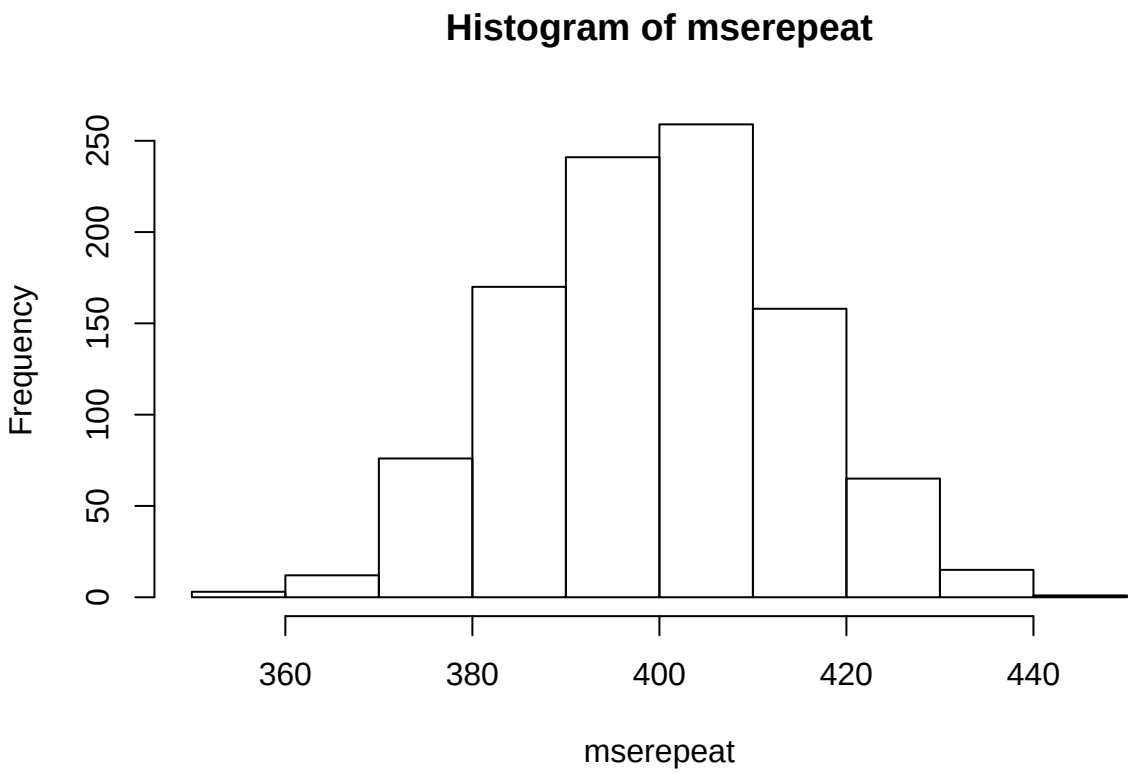
The test MSE calculated in question 2 is larger (test_mse = 426.12), as expected, than the MSE calculated in question 1 using the traditonal approach (tradmse = 395.27). The mean squared error (MSE) tells us the average error of our model's prediction compared to what that model should have predicted. One explanation for the higher MSE in our test set than in the traditional set is that the model in question 1 uses the entire data set, rather than a portion, and therefore has more information or data to be able to make a more accurate model and prediction. Additionally, another way to examine this is to say that the test model in question 2 is trying to form a general prediction without "seeing" all the data. Therefore, the model utilizing the entire data set (question 1) is able to make a more accurate and generalizable prediction because it "sees" more of the data. An important additional note, is that the data used for the training and test set in question 2 were split evenly (50/50). This tells us that the training set was not significantly disadvantaged from making an accurate prediction by having too small a sample (compared to the test set).

3. (30 points) Repeat the simple validation set approach from the previous question 1000 times, using 1000 different splits of the observations into a training set and a test/validation set. Visualize your results as a sampling distribution (hint: think histogram or density plots). Comment on the results obtained.

```
set.seed(1)
mserepeat <- vector("double", 1000)

for(i in 1:1000){
  train = sample(1:nrow(bidendata), 0.5*nrow(bidendata))
  test = setdiff(1:nrow(bidendata), train)
  mod <- lm(biden ~ female + age + educ + dem + rep, data = bidendata[train,])
  pred <- predict(mod, bidendata[test,])
  x <- bidendata$biden[test] - pred
  mserepeat[i] <- mean(x*x)
  }

hist(mserepeat)
```



4. (30 points) Compare the estimated parameters and standard errors from the original model in question 1 (the model estimated using all of the available data) to parameters and standard errors estimated using the bootstrap ($B = 1000$). Comparison should include, at a minimum, both numeric output as well as discussion on differences, similarities, etc. Talk also about the conceptual use and impact of bootstrapping.

```
mean_sample <- mean(bidendata$biden)
stemean_sample <- sqrt(mean_sample / nrow(bidendata))

lm_coefs <- function(splits, ...) {
  ## use `analysis` or `as.data.frame` to get the analysis data
mod <- lm(..., data = analysis(splits))
  tidy(mod)
}

trad_boot <- bidendata %>%
  bootstraps(1000) %>%
  mutate(coef = map(splits, lm_coefs, as.formula(biden ~ female + age + educ + rep + dem)))

trad_boot_lm_dataframe <- trad_boot %>%
  unnest(coef) %>%
  group_by(term) %>%
```

```
  summarize(boot.estimate = mean(estimate),
  boot.se = sd(estimate, na.rm = TRUE))

biden_lm_dataframe <- tidy(tradlm)

biden_lm_dataframe <- biden_lm_dataframe %>%
  left_join(trad_boot_lm_dataframe, by = "term") %>%
  select(c("term", "boot.estimate", "estimate", "boot.se", "std.error"))

biden_lm_dataframe
```

```
## # A tibble: 6 x 5
##   term        boot.estimate estimate boot.se std.error
##   <chr>               <dbl>    <dbl>   <dbl>     <dbl>
## 1 (Intercept)       58.8     58.8     2.94      3.12
## 2 female             4.05     4.10    0.965     0.948
## 3 age                0.0485   0.0483  0.0289    0.0282
## 4 educ              -0.343   -0.345   0.188     0.195
## 5 dem               15.4     15.4     1.04      1.07
## 6 rep              -15.9    -15.8     1.37      1.31
```

As can be seen in the table above, the original model and the bootstrap model have very comparable numbers. Both the standard error and the Beta estimate coefficient are nearly identical. However, the standard error for the original model is *slightly* smaller in the categories of "age", "dem", and "rep". This shows us that in those categories, the original model is more accurate than the bootstrap model. Likewise, the standard error for the bootstrap model is *slightly* smaller in the categories of "female" and "educ". Similarly, this shows that in those categories the bootstrap model is more accurate than the original model.

These findings are in-line with what we might expect using the parametric (original model) and non-parametric (bootstrap model) approaches. A parametric approach resamples a known distribution function using parameters estimated from your sample, while a non-parametric approach makes no assumptions about the distribution of the observations and resamples from the original sample. Since the parametric approach makes use of more assumptions from the known, observable data we would expect the model to be more accurate (have a lower standard error) than the non-parametric approach. Therefore, our findings align with our expectations with the original model (parametric approach) having a generally lower standard error than the bootstrap model (non-parametric approach). It's also important to note that a bootstrap model can be parametric - it simply is not for our study. Additionally, a bootstrap model is generally better for smaller sample sizes, where the general distribution may be unknown. For this study, the dataset is larger and the distribution known, thus the original model will likely yield more accurate results.