Code ▾

Machine Learning PSET 2

Krish Suchak

1. Estimate MSE based on Linear Regression Model of Dataset

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.

Hide

```
survey <- read.csv('nes2008.csv')
full_fit <- lm(formula = biden ~ ., data = survey)
summary(full_fit)
```

```

Call:
lm(formula = biden ~ ., data = survey)

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
```

Hide

```
mse <- mean(full_fit$residuals^2)
mse
```

```
[1] 395.2702
```

According to our results (specifically the coefficients of the fitted independent variables), female respondents exhibit more warmth towards Biden than do male respondents (average of 4 points greater). Older respondents also exhibit more warmth towards Biden than do younger responders. However, the p value ($0.09 > 0.05$) for this measure indicate that this result is not statistically significant. More educated voters exhibit less warmth towards Biden than do less educated voters. However, the p value ($0.08 > 0.05$) for this measure indicate that this result is not statistically significant. Lastly, Democrats are 15 points warmer to Biden than non-Democrats are on average. Conversely, Republicans are 15 points colder on the Biden sentiment scale than are non-Republicans. Our model has a weak fit ($R^2 = 0.28$), and our mean squared error is 395.

2. Calculate the Test MSE using Simple Holdout Validation

- 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.

Hide

```
    set.seed(45)
    samples <- sample(1 : nrow(survey), nrow(survey) * 0.5, replace = FALSE)
    train <- survey[samples, ]
    test <- survey[-samples, ]
```

- Fit the linear regression model using only the training observations.

Hide

```
    half_fit <- lm(formula = biden ~ ., data = train)
    summary(half_fit)
```

```


Call:
lm(formula = biden ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-75.189 -10.520   1.077  12.118  53.484 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  53.44908    4.34453  12.303  < 2e-16 ***
female        3.83148    1.33186   2.877  0.00411 ** 
age           0.05402    0.03929   1.375  0.16953    
educ         -0.10622    0.27431  -0.387  0.69868    
dem          17.34279    1.51015  11.484  < 2e-16 ***
rep         -13.68729    1.79637  -7.619 6.47e-14 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 19.6 on 897 degrees of freedom
Multiple R-squared:  0.2965,    Adjusted R-squared:  0.2926 
F-statistic: 75.61 on 5 and 897 DF,  p-value: < 2.2e-16
```

So, most of the coefficients from this model do not differ significantly from the last model. For the most part, the coefficients of this new model fall in the standard error ranges of the old model. Also, this model produces a similar fit to the last model ($R^2 = 0.30$).

- Calculate the MSE using only the test set observations.

Hide

```
    preds <- predict(half_fit, test)
    mse <- mean((test$biden - preds)^2)
    mse
```

```
[1] 412.6956
```

- How does this value compare to the training MSE from question 1? Present numeric comparison and discuss a bit.

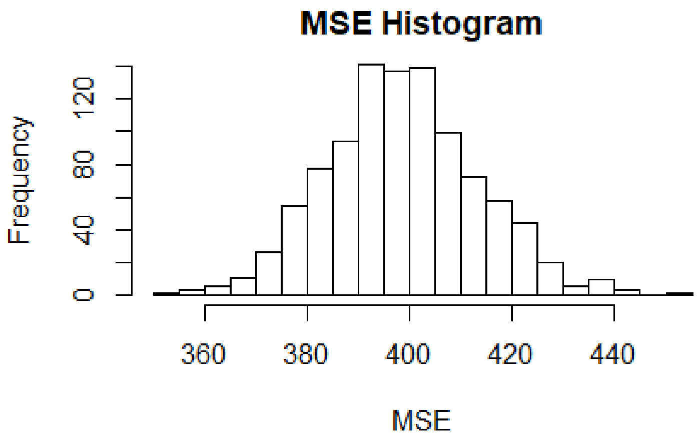
The test mean squared error has increased in this new model (compared to that of the old model) trained on only half the dataset and tested on the other half. Since we have only trained this new model on half the available data, it is reasonable to have a larger MSE than the previous model. This larger MSE suggests that the new model produces less accurate estimates than the old model.

3. MSE Histogram

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.

Hide

```
l <- list()
for (i in 1:1000) {
  samples <- sample(1:nrow(survey), nrow(survey)*0.5, replace = FALSE)
  train <- survey[samples, ]
  test <- survey[-samples, ]
  half_fit <- lm(biden ~ ., train)
  preds <- predict(half_fit, test)
  mse <- mean((test$biden - preds)^2)
  l <- c(l, mse)
}
hist(as.numeric(l), breaks=25, xlab="MSE", main="MSE Histogram")
```



Hide

```
mean(as.numeric(l))
```

```
[1] 398.7472
```

Hide

```
sd(as.numeric(l))
```

```
[1] 14.68601
```

So, the simulations produce a normal distribution of test MSEs with a mean of approximately 399. Considering that the model fit on the entire dataset has an MSE of 395, this model trained on only half the available data does not perform significantly worse than the previous model. Additionally, the benefit of a model only trained on a train partition of a dataset instead of the whole dataset is to ensure we do not overfit the model.

4. Bootstrapping

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.

Hide

```
library('dplyr')
library('rsample')
library(purrr)
library(tidyverse)
library(tidyr)
lm_coefs <- function(splits, ...) {
  mod <- lm(..., data = analysis(splits))
  tidy(mod)
}
boot <- survey %>%
  bootstraps(1000) %>%
  mutate(coef = map(splits, lm_coefs, as.formula(biden ~ .)))
boot %>%
  unnest(coef) %>%
  group_by(term) %>%
  summarize(.estimate = mean(estimate),
  .se = sd(estimate, na.rm = TRUE))
```

term	**.estimate**	**.se**
<chr>	<dbl>	<dbl>
(Intercept)	58.70041981	3.03210658
age	0.04822071	0.02881585
dem	15.43849678	1.06393197
educ	-0.33797363	0.19504117
female	4.09268845	0.94970417
rep	-15.80612514	1.37798510

6 rows

The coefficients in the table by the bootstrap above still fall in the standard error of the full model. This means that the differences in the observed coefficients are not statistically significant. The corresponding standard error values are also similar. Since bootstrapping does not rely on assumptions about the original dataset's distribution, estimates made from this bootstrap are more robust than ones made directly from the full model.

Additionally, the bootstrap contains the true population mean in its 95% confidence intervals.

The greatest benefit of bootstrapping is the ability to gain understanding about a population through the method of resampling rather than using more data. Since finding new data can be expensive or impossible in certain situations, bootstrapping serves as an important tool in a statistician's toolbox in gaining insights for a given dataset.