The Bootstrap

Nate Wells

Math 243: Stat Learning

September 27th, 2021

Outline

In today's class, we will. . .

- Discuss the bootstrap for estimating variance of error
- Implement bootstrapping in R

Section 1

The Bootstrap

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

- Suppose you are interested in the distribution of slopes $\hat{\beta}_3$ of the interaction term in an MLR model under random sampling:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2$$

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

- Suppose you are interested in the distribution of slopes $\hat{\beta}_3$ of the interaction term in an MLR model under random sampling:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2$$

- The classic approach:

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

- Suppose you are interested in the distribution of slopes $\hat{\beta}_3$ of the interaction term in an MLR model under random sampling:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2$$

- The classic approach:
    - Write the statistic $\hat{\beta}_3$ as a function of the random observations $x_1, \cdot, x_n$ and use properties of random variables to derive the theoretical distribution. Make some (sometimes unreasable) simplifying assumptions

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

- Suppose you are interested in the distribution of slopes $\hat{\beta}_3$ of the interaction term in an MLR model under random sampling:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2$$

- The classic approach:
    - Write the statistic $\hat{\beta}_3$ as a function of the random observations $x_1, \cdot, x_n$ and use properties of random variables to derive the theoretical distribution. Make some (sometimes unreasable) simplifying assumptions
    - Look up the theoretical distribution based on someone else's attempt to do part (1).

Why Bootstrap?

So, you want to know how a particular statistic is distributed?

- Suppose you are interested in the distribution of slopes $\hat{\beta}_3$ of the interaction term in an MLR model under random sampling:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2$$

- The classic approach:
  - Write the statistic $\hat{\beta}_3$ as a function of the random observations $x_1, \cdot, x_n$ and use properties of random variables to derive the theoretical distribution. Make some (sometimes unreasable) simplifying assumptions
  - Look up the theoretical distribution based on someone else's attempt to do part (1).
  - Hope that the sample size is large enough to allow the Central Limit Theorem to come into play so that the statistic is approximately Normal

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
    - Generate a large number of samples and compute the statistic of interest on each

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
  - Generate a large number of samples and compute the statistic of interest on each
  - Plot and summarize the distribution of the statistic.

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
    - Generate a large number of samples and compute the statistic of interest on each
    - Plot and summarize the distribution of the statistic.
    - The problem?

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
  - Generate a large number of samples and compute the statistic of interest on each
  - Plot and summarize the distribution of the statistic.
  - The problem?
- The bootstrap approach:

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
  - Generate a large number of samples and compute the statistic of interest on each
  - Plot and summarize the distribution of the statistic.
  - The problem?
- The bootstrap approach:
  - Assume that your sample is large enough to be "representative" of your population.

The Resampling Approach

As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
    - Generate a large number of samples and compute the statistic of interest on each
    - Plot and summarize the distribution of the statistic.
    - The problem?
- The bootstrap approach:
    - Assume that your sample is large enough to be “representative” of your population.
    - Create a new bootstrap sample by sampling **with replacement** from your original sample, a number of times equal to your original sample size.

The Resampling Approach

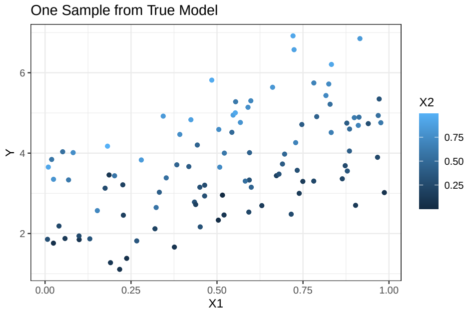
As an alternative to using the theoretical distribution, use simulation to approximate.

- The optimistic approach:
  - Generate a large number of samples and compute the statistic of interest on each
  - Plot and summarize the distribution of the statistic.
  - The problem?
- The bootstrap approach:
  - Assume that your sample is large enough to be “representative” of your population.
  - Create a new bootstrap sample by sampling **with replacement** from your original sample, a number of times equal to your original sample size.
  - Repeat the process to create many bootstrap samples. Compute the statistic of interest on each and plot the results.

Bootstrap Demo

Suppose $Y = 1 + 2 \cdot X_1 + 3 \cdot X_2 + X_1 \cdot X_2 + \epsilon$ with $\epsilon \sim N(0, 0.25)$.

```
set.seed(10101)
n<-100
X1<-runif(n, 0, 1)
X2 <- runif(n, 0, 1)
e<-rnorm(n, 0 ,.5)
Y<-1 + 2*X1 + 3*X2 + X1*X2+ e
d<-data.frame(X1, X2, Y)
```



Bootstrap Demo

```
my_mod<-lm(Y ~ X1*X2, data = d)
summary(my_mod)$coefficients
```

```
##              Estimate Std. Error  t value     Pr(>|t|)
## (Intercept) 1.447174  0.2100171 6.890742 5.807042e-10
## X1          1.317290  0.3803365 3.463485 7.982768e-04
## X2          2.405724  0.4102938 5.863417 6.404175e-08
## X1:X2       2.044325  0.7415455 2.756844 6.985948e-03
```

```
b3 <-  my_mod$coefficients[4]
```

The Simulation Approach

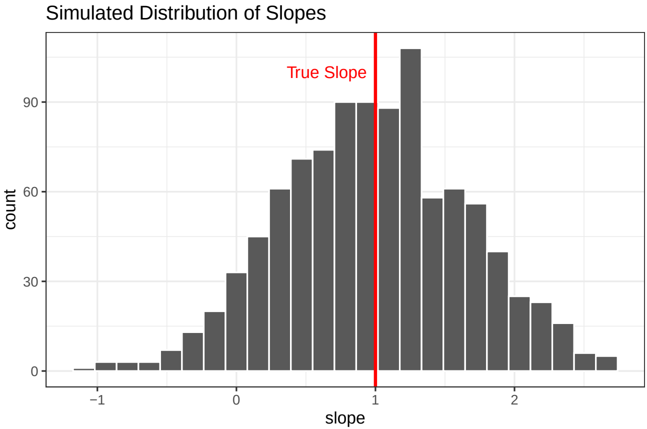
```
set.seed(234)
trials<-1000 #Number of simulations
n<-100 #Number points in each simulation
X1<-runif(n, 0, 1) # Generate random X1; same for all sims
X2 <- runif(n, 0, 1) # Generate random X1; same for all sims
slopes<-data.frame() #Create empty dataframe for the slopes

for (i in 1:trials){
sim_e<-rnorm(n, 0 ,.5)
sim_Y<-1 + 2*X1 + 3*X2 + X1*X2+ sim_e
sim_d<-data.frame(X1, X2, sim_Y)
sim_mod<-lm(sim_Y ~ X1*X2, data = sim_d)
slopes<-rbind( slopes,
               data.frame(slope = summary(sim_mod)$coefficients[4,1]))
}
```

```
head(slopes)
```

```
##        slope
## 1  0.5494089
## 2  1.4382129
## 3  0.9934332
## 4  0.7086642
## 5 -0.9140541
## 6  1.8136110
```

Simulation Distribution




```r
slopes %>% summarize(mean_slope = mean(slope), sd_slope = sd(slope))
```

```
##   mean_slope  sd_slope
## 1  0.9895467 0.6620953
```

The Bootstrap Approach

We have 1 sample:

```
head(d)
```

```
##          X1        X2        Y
## 1 0.1903066 0.1056760 1.275277
## 2 0.9108393 0.6749109 4.690218
## 3 0.2277161 0.1748862 2.455955
## 4 0.8249905 0.7360649 5.719890
## 5 0.9155760 0.8434911 6.849461
## 6 0.5052083 0.7491072 4.589090
```

But we can create a bootstrap sample:

```
set.seed(135)
a_bootstrap_sample<-slice_sample(d, n = n, replace = T)
```

The Bootstrap Approach

We have 1 sample:

```
head(d)
```

```
##          X1        X2        Y
## 1 0.1903066 0.1056760 1.275277
## 2 0.9108393 0.6749109 4.690218
## 3 0.2277161 0.1748862 2.455955
## 4 0.8249905 0.7360649 5.719890
## 5 0.9155760 0.8434911 6.849461
## 6 0.5052083 0.7491072 4.589090
```

But we can create a bootstrap sample:

```
set.seed(135)
a_bootstrap_sample<-slice_sample(d, n = n, replace = T)
```

Duplicates?

```
common<-intersect(a_bootstrap_sample, d)
length(common$X1)
```

```
## [1] 66
```

The Bootstrap Approach, cont'd

Now, we create 1000 bootstraps and calculate the slope of each

```
# Create a function to compute statistic from bootstrap sample
set.seed(929)
interaction_slope <- function(split){
  x <- analysis(split)
  boot_mod <-lm(Y ~ X1*X2 , data = x)
  slope <- boot_mod$coefficients[4]
  slope
}
```

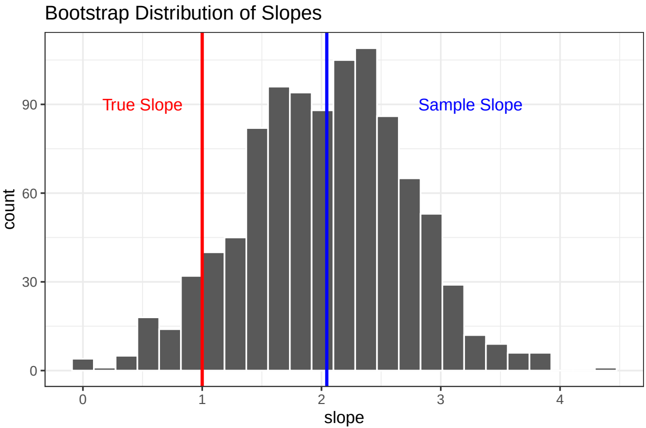
The Bootstrap Approach, cont'd

Now, we create 1000 bootstraps and calculate the slope of each

```
# Create a function to compute statistic from bootstrap sample
set.seed(929)
interaction_slope <- function(split){
  x <- analysis(split)
  boot_mod <-lm(Y ~ X1*X2 , data = x)
  slope <- boot_mod$coefficients[4]
  slope
}
```

```
# Use rsample to create bootstrap samples and apply function
library(rsample)
bt_resamples <- bootstraps(d, times = 1000)
bt_resamples$slope <- map_dbl(bt_resamples$splits, interaction_slope)
```

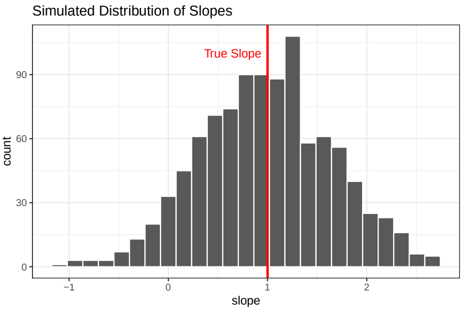
Bootstrap Distribution

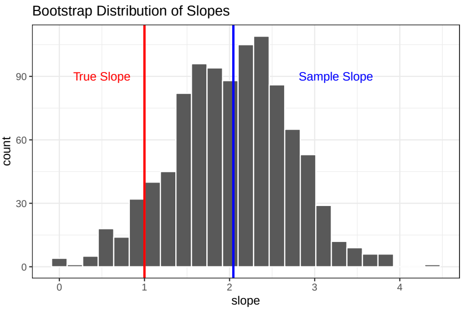



```
bt_resamples %>% summarize(mean_slope = mean(slope), sd_slope = sd(slope))
```

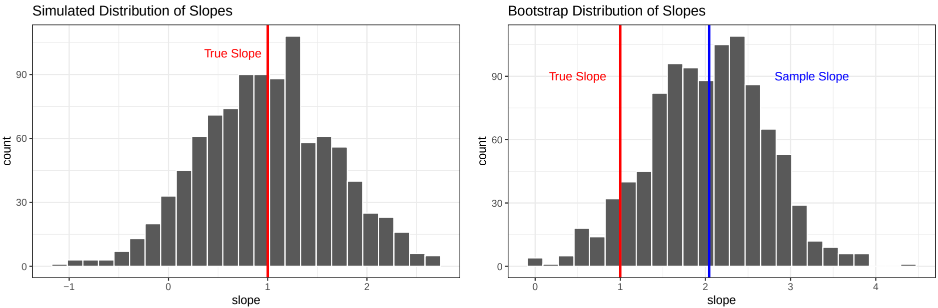
```
##   mean_slope  sd_slope
## 1   2.026826 0.6849343
```

Side-by-Side Comparison





Side-by-Side Comparison




```
rbind(slopes, bt_resamples %>% select(slope)) %>%
  cbind(method = rep(c("sim", "boot"), each = 1000)) %>%
  group_by(method) %>% summarize(mean_slope = mean(slope), sd_slope = sd(slope),
                                 q.025 = quantile(slope,.025), q.975 = quantile(slope, .975))
```

```
## # A tibble: 2 x 5
##   method mean_slope sd_slope  q.025 q.975
##   <fct>       <dbl>    <dbl>  <dbl> <dbl>
## 1 boot        2.03     0.685  0.620  3.34
## 2 sim         0.990    0.662 -0.286  2.27
```

CV verus Bootstrapping

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

CV verus Bootstrapping

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

Cross-validation: Often used for *model assessment* and *model selection*.

CV verus Bootstrapping

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

Cross-validation: Often used for *model assessment* and *model selection*.

- Partition data into test and train
- Fit model to train, predict on test
- Iterate though all possible *folds*
- Compute aggregate measure of predictive ability

CV verus Bootstrapping

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

Cross-validation: Often used for *model assessment* and *model selection*.

- Partition data into test and train
- Fit model to train, predict on test
- Iterate though all possible *folds*
- Compute aggregate measure of predictive ability

Bootstrapping: Often used for *quantifying uncertainty*.

CV verus Bootstrapping

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

Cross-validation: Often used for *model assessment* and *model selection*.

- Partition data into test and train
- Fit model to train, predict on test
- Iterate though all possible *folds*
- Compute aggregate measure of predictive ability

Bootstrapping: Often used for *quantifying uncertainty*.

- Draw a bootstrap sample of size n from your data *with replacement*.
- Compute estimate of interest
- Consider distribution of bootstrap estimates over many samples