Bootstrap Resampling

Alireza Sheikh-Zadeh, PhD

Bootstrapping is a statistical method for estimating the sampling distribution of an estimator by sampling with replacement from the original sample, most often with the purpose of deriving robust estimates of standard errors and confidence intervals of a population parameter like a mean, median, proportion, odds ratio, correlation coefficient or regression coefficient.

A great advantage of bootstrap is its simplicity. It is a straightforward way to derive estimates of standard errors and confidence intervals for complex estimators of complex parameters of the distribution, such as percentile points, proportions, odds ratio, and correlation coefficients. Bootstrap is also an appropriate way to control and check the stability of the results. Although for most problems it is impossible to know the true confidence interval, bootstrap is asymptotically more accurate than the standard intervals obtained using sample variance and assumptions of normality.

Why the bootstrap works?

The underlying sample that you have collected is the best information you can have about what the population actually looks like. And you surely agree with me that most samples will, if randomly chosen, look quite like the population they came from. Consequently, it is likely that your sample does too. Let us sample a large number of data sets from our underlying sample and compute the statistic of interest on each of these datasets. Thus we receive a distribution of our statistic. This distribution expresses the variability of our estimate. Just let the computer do the work for us.

Example Find a 95% confidence interval by bootstrapping for the mean of "income" variable in "Prestige" dataset. Compare your answer with the theoretical confidence interval using the central limit theorem.

```
library("car")

## Loading required package: carData

## Warning: package 'carData' was built under R version 3.5.2

data("Prestige")
income <- Prestige$income
mean(income)

## [1] 6797.902

boot_mean <- function(x){
  mean(sample(x, replace = TRUE))
}
```

```r
num_sim = 1000

boot_income_rep <- replicate(num_sim, boot_mean(income))

# Bootstrapping answer
quantile(boot_income_rep, c(0.025, 0.975))
```

```
##     2.5%    97.5%
## 5980.553 7601.151
```

```r
# hist(boot_income_rep)

# CLT answer
xbar = mean(income)

#Xbar (+-) 1.96 * sigma/sqrt(n)

se <- qnorm(0.975)*sd(income)/sqrt(length(income))

# low bound
xbar - se
```

```
## [1] 5973.916
```

```r
# upper bound
xbar + se
```

```
## [1] 7621.888
```

Example Find a 95% confidence interval by bootstraping for the standard-deviation of "income" variable in "Prestige" dataset.

```r
income <- Prestige$income
sd(income)
```

```
## [1] 4245.922
```

```r
boot_sd <- function(x){
  sd(sample(x, replace = TRUE))
}

num_sim = 1000

boot_income_rep <- replicate(num_sim, boot_sd(income))

# Bootstrapping answer
quantile(boot_income_rep, c(0.025, 0.975))
```

```
##     2.5%    97.5%
## 2956.151 5381.451
```

Practice: Find a 95% confidence interval by bootstrapping for the skewness of "income" variable in "Prestige" dataset.

Example Find a 95% confidence interval by bootstrapping for the correlation between "income" and "education" variables in "Prestige" dataset.

```
income <- Prestige$income
edu <- Prestige$education

data <- cbind(income, edu)

cor(data)[1,2]
```

```
## [1] 0.5775802
```

```
boot_cor <- function(x){
  rowIndex = sample(1:nrow(x), replace = TRUE)
  cor(x[rowIndex,])[1,2]
}

num_sim = 1000

boot_income_rep <- replicate(num_sim, boot_cor(data))

# Bootstrapping answer
quantile(boot_income_rep, c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.4617226 0.6870120
```

Example

The New York Times had on January 27, 1987, on its front page an article entitled Heart Attack Risk Found to be Cut by Taking Aspirin. This double-blind trial ultimately led to the following table; see also Efron and Tibshirani (1993):

	Heart Attacks	#Persons
Aspirin	104	11037
Placebo	189	11034

The odds ratio of the two components is the following: (104/11,037)/(189/11,034) = 0.55

A header in the newspaper could possibly be: Those who take aspirin regularly have only 55% as many heart attacks as people who take no aspirin.

As statisticians, we want to estimate the real population parameter θ. Of course, we are not really interested in only since is still only a point estimate of $\hat{\theta}$. If we conducted the study again and collected new data, we would get another result (different from 0.55).

We are interested in the accuracy/variability/uncertainty of = 0.55 (statistical inference).

But how do we calculate the confidence interval (CI) for $\hat{\theta}$?

	Heart Attacks	#Persons
Aspirin	a	c
Placebo	b	d

$log(\hat{\theta}) \pm 1.96 * \sqrt{1/a + 1/b + 1/c + 1/d}$

```
dat <- matrix(c(104,11037,189,11034),2,2, byrow=TRUE)
dat
```

```
##      [,1]  [,2]
## [1,]  104 11037
## [2,]  189 11034
```

```
library(vcd)
```

```
## Warning: package 'vcd' was built under R version 3.5.3
```

```
## Loading required package: grid
```

```
confint(oddsratio(dat, log=FALSE))
```

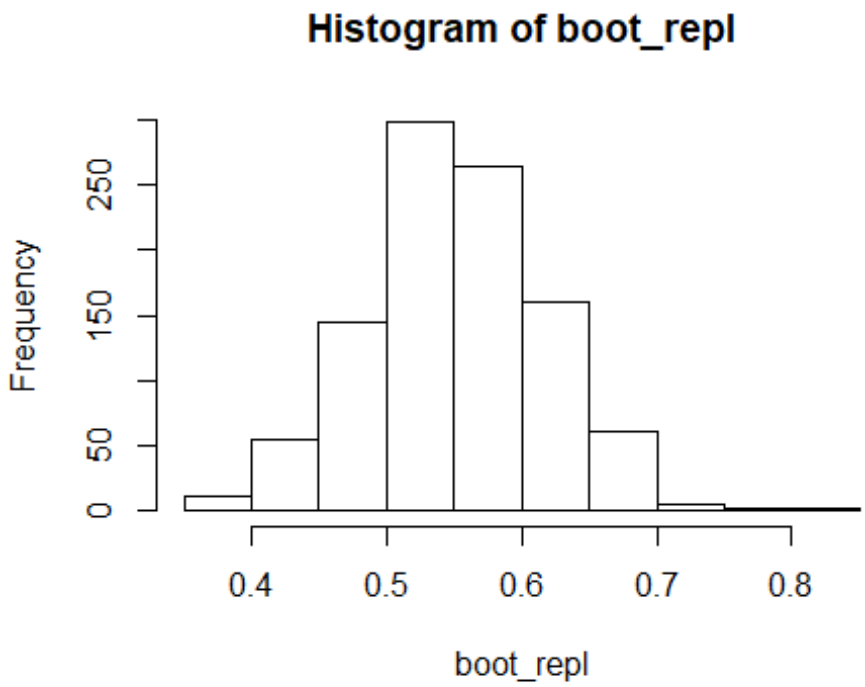
```
##       2.5 %    97.5 %
## / 0.4324132 0.6998549
```

The following questions remain unanswered:

- Are there better analytical estimates of the confidence interval available?

- Do we have simpler methods for the determination of the confidence interval?

```
# Bootstrapping

## original surveyed data
s1 <- rep(c(TRUE, FALSE), times = c(104, 11037-104))
s2 <- rep(c(TRUE, FALSE), times = c(189, 11034-189))

## function for drawing a bootstrap sample
## and estimating the bootstrap replicate

boot_oddRatio <- function(s1, s2){
## odds ratio
# Sampling with replacement
ac <- sum(sample(s1, replace = TRUE))/length(s1)
bd <- sum(sample(s2, replace = TRUE))/length(s2)
oddsRatio <- ac/bd
return(oddsRatio)
}

num_sim = 1000

boot_repl <- replicate(num_sim, boot_oddRatio(s1, s2))

## confidence interval
quantile(boot_repl, c(0.025, 0.975))

##      2.5%     97.5%
## 0.4257167 0.6837375

# Let's look at the distribution of the odds ratio.
hist(boot_repl)
```



For this example, the confidence intervals by bootstrap are very close to the one estimated previously from the analytical method. The estimation of confidence intervals using bootstrap was data-based, without preconditions (except the assumption that a good random number is chosen) and assumptions, and done in a (almost) very intuitive manner without mathematics.

The Application of Bootstrapping in Hypothesis Testing

Assessing whether the difference between average food expense in spring 2014 and fall 2014 is explainable by a null model where all data are produced as i.i.d by the same distribution.

In other words we want to test:

H0: Mu1 - Mu2 = 0

```
food.s14 = read.csv("food-sp-14")
food.f14 = read.csv("food-fa-14")

fs = food.s14$Food
ff = food.f14$Food
fs

##  [1]  7 22 15  9 10  8 10 15 15 10 15 20 25 30 10  5  8  7 15  8

ff
```

```
##  [1] 18 12  7 10  8 18  5 22 10 25  4 10 15 17 10  7 15 17  5 14 11 15 15
## [24] 15 12 10 10  5 10 20 12  6
```

```
# xBar1 - Xbar2
diff = mean(fs) - mean(ff)
diff
```

```
## [1] 1.0125
```

```
## Under the iid model, all data are from the same process. So pool the data
to estimate the process distribution via the bootstrap distribution.

alldata = c(fs,ff)
# hist(alldata)
# qq.obj = qqnorm(alldata)
# agreement = cor(qq.obj$x, qq.obj$y)
# agreement
n1 = length(fs)
n2 = length(ff)
n1
```

```
## [1] 20
```

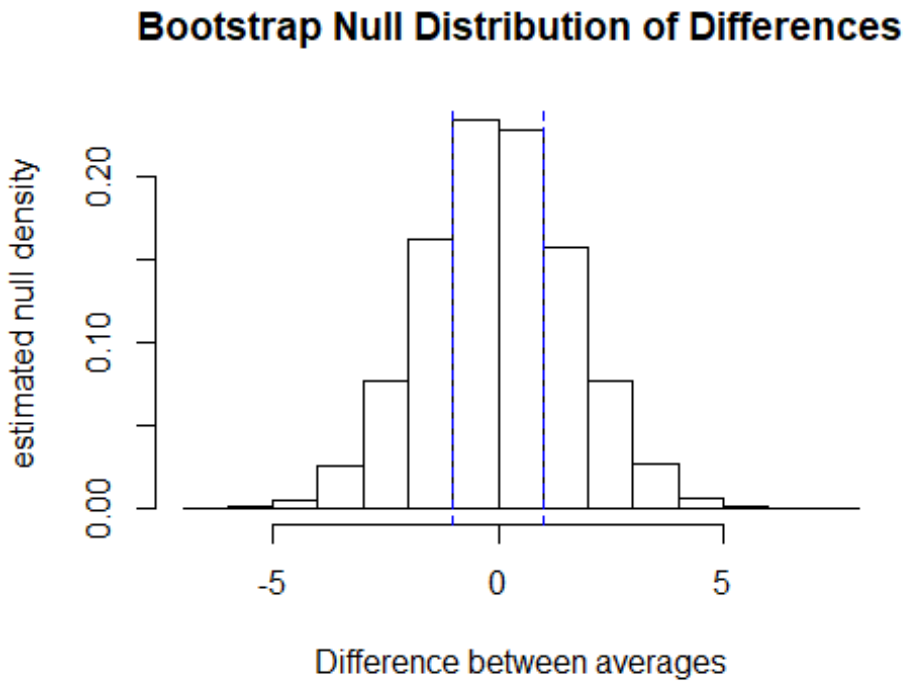
```
n2
```

```
## [1] 32
```

```
## You can approximate the null distribution of the difference by simulating
all data from the same distribution.
# Null Hypothesis: Mu1 = Mu2 (H0: two sample has the same mean and came from
the same distribution)
## A good distribution to use is the bootstrap distribution of the combined d
ata.

boot_mean <- function(x, sampleSize){
  mean(sample(x, sampleSize, replace = TRUE))
}

Nsim = 100000

xbar1.sim = replicate(Nsim, boot_mean(alldata, n1))

xbar2.sim = replicate(Nsim, boot_mean(alldata, n2))

head(cbind(xbar1.sim, xbar2.sim, xbar1.sim - xbar2.sim))
```

```
##      xbar1.sim xbar2.sim
## [1,]     11.10  12.09375 -0.99375
## [2,]     12.05  11.40625  0.64375
## [3,]     11.25  11.53125 -0.28125
## [4,]     12.60  12.78125 -0.18125
```

```
## [5,]     14.65  12.59375  2.05625
## [6,]     11.15  12.84375 -1.69375

## Now, the null distribution of the difference between averages is estimated
using these Nsim differences:

null.diff = xbar1.sim - xbar2.sim
hist(null.diff, main="Bootstrap Null Distribution of Differences", freq=F, xl
ab = "Difference between averages", ylab = "estimated null density")

## The observed difference and its negative are indicated by dashed vertical
lines
abline(v = c(diff,-diff), lty=2, col = 'blue')
```




```
## The two-sided p-value calculation by Monte-Carlo Simulation:

head(cbind(xbar1.sim, xbar2.sim, null.diff, null.diff >= abs(diff), null.diff
<= -abs(diff)))

##      xbar1.sim xbar2.sim null.diff
## [1,]     11.10  12.09375  -0.99375 0 0
## [2,]     12.05  11.40625   0.64375 0 0
## [3,]     11.25  11.53125  -0.28125 0 0
## [4,]     12.60  12.78125  -0.18125 0 0
## [5,]     14.65  12.59375   2.05625 1 0
## [6,]     11.15  12.84375  -1.69375 0 1
```

```r
pval2 = mean(null.diff >= abs(diff)) + mean(null.diff<= -abs(diff))
pval2
```

```
## [1] 0.53389
```

```r
# Compare it with t-test:

t.test(fs, ff)
```

```
##
##  Welch Two Sample t-test
##
## data:  fs and ff
## t = 0.57731, df = 33.341, p-value = 0.5676
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.554298  4.579298
## sample estimates:
## mean of x mean of y
##   13.2000   12.1875
```

Practice Assess whether the variance of food expense in Spring 2014 and Fall 2014 is explainable by a null model where all data are produced as iid by the same distribution.

In other words we want to test:

H0: Var1/Var2 = 1