STAT 641: BOOTSTRAPPING METHODS

Jiyoun Myung

Department of Statistics and Biostatistics
California State University, East Bay

Spring 2021, Day 7

Law data

```
library(bootstrap) # law data
head(law)
```

```
##   LSAT  GPA
## 1  576 3.39
## 2  635 3.30
## 3  558 2.81
## 4  578 3.03
## 5  666 3.44
## 6  580 3.07
```

We want to estimate the correlation between LSAT and GPA scores.

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

```
## [1] 0.7763745
```

boot.ci function in boot package

Let's find an approximate 95% confidence interval for the correlation using the percentile bootstrap approach for the **law** data.

```
library(boot)
theta.hat <- function(d,i){ cor(d[i, 1], d[i, 2]) }
#Perform bootstrapping using the boot function.
set.seed(123)
boot_corr <- boot(data = law, statistic = theta.hat , R = 5000)

boot.ci(boot_corr, conf = .95, type = "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr, conf = 0.95, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.4681,  0.9622 )
## Calculations and Intervals on Original Scale
```

Confidence Intervals

Theoretical CI

Note: $\text{se}(r) = \sqrt{\frac{1-r^2}{n-2}}$.

```
n <- nrow(law)
(r.s <- cor(law[,1], law[,2])) # sample correlation coefficient
```

```
## [1] 0.7763745
```

```
(r.se <- sqrt((1-r.s^2)/(n - 2))) # standard error of sample correlation coefficient
```

```
## [1] 0.174806
```

```
# 95% CI
t.crit <- qt(0.975, df = n-2)
c(r.s - t.crit*r.se, r.s  + t.crit*r.se)
```

```
## [1] 0.3987292 1.1540198
```

Alternatives:

- Use Fisher's transformation. See the Chapter 6.
- Use the bootstrap.

boot.ci function in boot package

Law data

Let's find an approximate 95% confidence intervals for the correlation coefficient between LSAT and GPA.

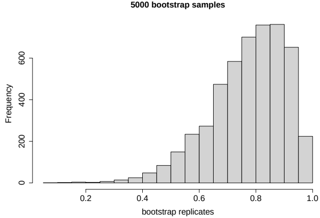
```
boot.ci(boot_corr)  # conf = .95
```

```
## Warning in boot.ci(boot_corr): bootstrap variances needed for studentized
## intervals
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.5192,  1.0414 )   ( 0.5905,  1.0847 )
##
## Level     Percentile            BCa
## 95%   ( 0.4681,  0.9622 )   ( 0.3110,  0.9372 )
## Calculations and Intervals on Original Scale
```

Confidence Intervals

Bootstrap sampling distribution

```r
hist(boot_corr$t)
```



Not normal.

```r
# sample correlation coefficient
boot_corr$t0
```

```
## [1] 0.7763745
```

```r
# bootstrap estimate of r
mean(boot_corr$t)
```

```
## [1] 0.7724472
```

```r
# bootstrap estimate of Bias
(bias.est <- mean(boot_corr$t) - boot_corr$t0)
```

```
## [1] -0.003927276
```

```r
# std error of bootstrap estimate
(boot.se <- sd(boot_corr$t))
```

```
## [1] 0.1332268
```

Bootstrap Confidence Intervals

(Standard) Normal

```
boot.ci(boot_corr, type = "norm")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr, type = "norm")
##
## Intervals :
## Level      Normal
## 95%   ( 0.5192,  1.0414 )
## Calculations and Intervals on Original Scale
```

```
# Biased corrected estimate
(corrected <- r.s - bias.est)
```

```
## [1] 0.7803018
```

```
z.crit <- qnorm(0.975)
lwr <- corrected - z.crit*boot.se
upr <- corrected + z.crit*boot.se
c(lwr, upr)
```

```
## [1] 0.5191821 1.0414214
```

Bootstrap Confidence Intervals

Basic

```r
boot.ci(boot_corr, type = "basic")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr, type = "basic")
##
## Intervals :
## Level      Basic
## 95%   ( 0.5905,  1.0847 )
## Calculations and Intervals on Original Scale
```

```r
boot.q <- quantile(boot_corr$t, c(0.025, 0.975))
lwr2 <- 2*boot_corr$t0 - boot.q[2]
upr2 <- 2*boot_corr$t0 - boot.q[1]
c(lwr2, upr2)
```

```
##     97.5%      2.5%
## 0.5905242 1.0845237
```

Bootstrap Confidence Intervals

Percentile

```r
boot.ci(boot_corr, type = "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.4681,  0.9622 )
## Calculations and Intervals on Original Scale
```

```r
quantile(boot_corr$t, c(.025, .975), type = 6)
```

```
##      2.5%     97.5%
## 0.4680678 0.9622447
```

```r
quantile(boot_corr$t, c(.025, .975))
```

```
##      2.5%     97.5%
## 0.4682253 0.9622248
```

Bootstrap Confidence Intervals

Bias Corrected and Accelerated (BCa) Confidence Intervals

These are percentile-based confidence intervals adjusted for the bias and skewness. That is, the endpoints of the intervals have bias adjustments.
The $100(1-\alpha)\%$ CI is

$$\left(\hat{\theta}^{*(\alpha_1)},\ \hat{\theta}^{*(\alpha_2)}\right)$$

where

$$\alpha_1 = \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z^{(\alpha/2)}}{1-\hat{a}(\hat{z}_0 + z^{(\alpha/2)})}\right),\ \ \alpha_2 = \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z^{(1-\alpha/2)}}{1-\hat{a}\left(\hat{z}_0 + z^{(1-\alpha/2)}\right)}\right),$$

$$\hat{a} = \frac{\sum_{i=1}^{n}\left(\hat{\theta}_{(\cdot)} - \hat{\theta}_{(i)}\right)^3}{6\left\{\sum_{i=1}^{n}\left(\hat{\theta}_{(\cdot)} - \hat{\theta}_{(i)}\right)^2\right\}^{3/2}},\ \text{ and } \hat{z}_0 = \Phi^{-1}\left(\frac{1}{B}\sum_{b=1}^{B} I_{[\hat{\theta}^*(b)<\hat{\theta}]}\right)$$

where $\hat{a}$ is the estimated acceleration term with $\hat{\theta}_{(i)}$ the estimate after deleting the ith case and $\hat{\theta}_{(\cdot)}$ the mean of $\hat{\theta}_{(1)}, \cdots \hat{\theta}_{(n)}$, and $\hat{z}_0$ is the estimated bias correction term (a measure of the discrepancy between the median bias of $\hat{\theta}^*(b)$ and θ.) Also $z^{(\alpha)} = \Phi^{-1}(\alpha)$ is the 100αth percentile point of a standard normal distribution.

Bootstrap Confidence Intervals

BCa

```
boot.ci(boot_corr, type = "bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   ( 0.3110,  0.9372 )
## Calculations and Intervals on Original Scale
```

NOTE: We saw that the percentile and BCa methods were the only ones considered here that were guaranteed to return a confidence interval that respected the statistic's sampling space. It turns out that there are theoretical grounds to prefer BCa in general. **It is "second-order accurate", meaning that it converges faster to the correct coverage.** Unless you have a reason to do otherwise, make sure to perform a sufficient number of bootstrap replicates (a few thousand is usually not too computationally intensive) and go with reporting BCa intervals.

Source: https://blog.methodsconsultants.com/posts/understanding-bootstrap-confidence-interval-output-from-the-r-boot-package/

Bootstrap Confidence Intervals

Bootstrap-t

```
theta.hat2 <- function(d,i){
   corr <- cor(d[i, 1], d[i, 2])
   cor.var <- sqrt((1-corr^2)/(nrow(d)-1))
   return(c(corr, cor.var))
}
#Perform bootstrapping using the boot function.
set.seed(123)
boot_corr_t <- boot(data = law, statistic = theta.hat2 , R = 5000)
boot.ci(boot_corr_t, type = "stud")
```

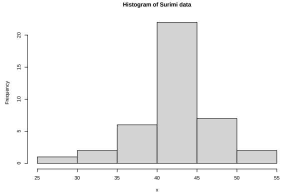
```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_corr_t, type = "stud")
##
## Intervals :
## Level    Studentized
## 95%   ( 0.4935,  1.0367 )
## Calculations and Intervals on Original Scale
```

Example: Surimi Data

Intereste parameter is Mean.

" Surimi " is purified fish protein used as a material to make imitation crab and shrimp food products. The strength of surimi gels is a critical factor in production. Each incoming lot of surimi raw material is sampled and a cooked gel is prepared. From these gels, test portions are selected and tested for strength. Our sample data are the measured ultimate stresses to penetrate the test portions for 40 incoming lots of surimi.

```r
set.seed(1)
# deformation stress required to puncture test
# specimens for 40 lots of surimi
x <- c(41.28, 45.16, 34.75, 40.76, 43.61, 39.05,
       41.20, 41.02, 41.33, 40.61, 40.49, 41.77,
       42.07, 44.83, 29.12, 45.59, 41.95, 45.78,
       42.89, 40.42, 49.31, 44.01, 34.87, 38.60,
       39.63, 38.52, 38.52, 43.95, 49.08, 50.52,
       43.85, 40.64, 45.86, 41.25, 50.35, 45.18,
       39.67, 43.89, 43.89, 42.16)
```



Source: An Introduction to Bootstrap Methods with applications to R, Chernik and LaBudde. (2011)

Example: Surimi Data

```r
mu0 <- mean(x) #mean of original sample
n <-  length(x)
Shat <- sd(x)/sqrt(n)
thetas <- c()
tstar <- c()
for (i in 1:1000) {
x_sample <- sample(x, n, replace = TRUE) #new resample
mu <- mean(x_sample) #estimate
thetas[i] <- mu
tstar[i] <- (mu - mu0)/(sd(x_sample)/sqrt(n))#pivotal quantity
}
```

Example: Surimi Data

```
c(mu0, mean(thetas)) #compare sample mean to mean of bootstrap sampling distribution

## [1] 42.18575 42.16095

c(Shat, sd(thetas)) #compare standard error from sample to standard error estimate from bootstrap distribu

## [1] 0.6576018 0.6427834

quantile(tstar, c(0.025, 0.975)) #quantiles from bootstrap percentile t

##      2.5%     97.5%
## -1.875655  1.994204

qt(c(0.025,0.975), n - 1) #quantiles from student t distribution

## [1] -2.022691  2.022691

mu0 - quantile(tstar, c(0.975, 0.025))*Shat #bootstrap percentile t confidence interval

##    97.5%     2.5%
## 40.87436 43.41918

mu0 - qt(c(0.975, 0.025), n - 1)*Shat #student t confidence interval for comparison

## [1] 40.85562 43.51588
```

Bootstrap-t

For comparison to the script given above, the function " boott " from the package " bootstrap " gives

```
set.seed(1) #reproducibility
library(bootstrap)
sdmean <- function(x, ...) {sqrt(var(x)/length(x))}
bt <- boott(x, theta = mean, sdfun = sdmean, nboott = 1000,
            perc = c(0.025, 0.975)) #bootstrap percentile t
bt$confpoints
```

```
##         0.025    0.975
## [1,] 40.78712 43.47851
```

Simple idea with intuitive procedure and works well for location parameters. But it is particularly not resistant to outliers or crazy sampling distributions of the statistic and doesn't work as well for correlation/association measures.

Bootstrap CIs

CI	Symmetric	Range Resp	Trans Resp	Accuracy	Normal Samp Dist?	Other
BS SE	Yes	No	No	1^{st} order	Yes	param assump $F(\hat{\theta})$
BS-t	No	No	No	2^{nd} order	Yes/No	computer intensive
perc	No	Yes	Yes	1^{st} order	No	small $n \rightarrow$ low accuracy
BCa	No	Yes	Yes	2^{nd} order	No	limited param assump