Bootstrap: introduction (1/2)

Suppose that we observe a sample $x_1, ..., x_n$ which are realizations of i.i.d. r.v. $X_1, ..., X_n$, with distribution L_x and with CDF F_x unknown.. The empirical Cummulative Distribution Function (eCDF) is defined as

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{x_i \leq x}$$

and is an estimator of the true CDF F_x. Moreover, we have that

$$E[F_n(x)] = F_x$$

and

$$var(F_n(x)) = \frac{1}{n}\Big((F_x)(1 - F_x)\Big)$$

Bootstrap: introduction (2/2)

Besides, from the Law of Large Numbers (LLN), we have that

$$F_n(x) \xrightarrow{a.s.} F_x \qquad \text{as } n \text{ goes to } \infty.$$

And from the Central Limit Theorem (CLT), we have that

$$F_n(x) \xrightarrow{L} N\left(F_x, \frac{1}{n}F_x(1 - F_x)\right) \qquad \text{as } n \text{ goes to } \infty.$$

The underlying idea of nonparametric bootstrap is that bootstrap samples can be generated from the $F_n(x)$ when F_x is unknow since $F_n(x)$ is a consistent estimator for F_x .

General algorithm for Bootstrap

1. B independent samples $x^*_{b1}, ..., x^*_{bn}$, $b = 1, ..., B$ are drawn with replacement from the data $x_1, ..., x_n$.

2. On each of the B independent samples, an estimate $\hat{\theta}^*_b$ is computed (the bootstrap replicates of $\hat{\theta}$).

3. The expectation of $\hat{\theta}$ (given $F_n(x)$) is estimated as follows:

$$E_{boot}[\hat{\theta}] = \frac{1}{B} \sum_{b=1}^{B} \hat{\theta}^*_b$$

4. The variance of $\hat{\theta}$ (given $F_n(x)$) is estimated as follows:

$$var_{boot}(\hat{\theta}) = \frac{1}{B-1} \sum_{b=1}^{B} \left(\hat{\theta}^*_b - \frac{1}{B} \sum_{b=1}^{B} \hat{\theta}^*_b \right)^2$$

Example 1

Example: We observe a the following sample:

$$\begin{aligned} x_1, ..., x_n = & 8.26, 6.33, 10.4, 5.27, 5.35, 5.61, 6.12, \\ & 6.19, 5.2, 7.01, 8.74, 7.78, 7.02, 6, 6.5, \\ & 5.8, 5.12, 7.41, 6.52, 6.21, 12.28, 5.6, 5.38, 6.6, 8.74 \end{aligned}$$

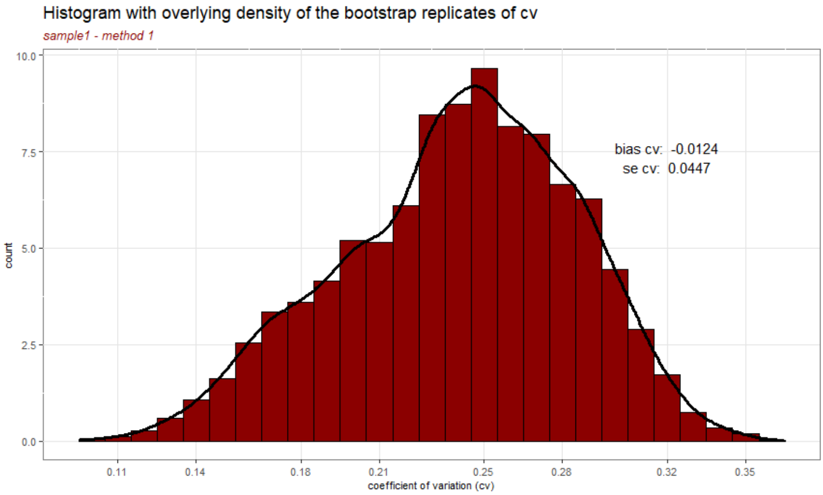
Estimate the bias and the standard error of $\hat{\theta}$, the estimate of the coefficient of variation. The coefficient of variation cv is equal to

$$cv = sd(x)/mean(x)$$

Method 1 the R language

```r
library(bootstrap)

sample1 = c(8.26, 6.33, 10.4, 5.27, 5.35, 5.61, 6.12, 6.19, 5.2, 7.01, 8.74,
 7.78, 7.02, 6, 6.5, 5.8, 5.12, 7.41, 6.52, 6.21, 12.28, 5.6, 5.38, 6.6, 8.74)

#-----------------------------------------------
# Perform bootstrap to estimate the cv - mehthod 1
#-----------------------------------------------

# define the function to estimatee the cv
theta_hat = function(x) {
  sd(x) / mean(x) # coefficient of variation
}

set.seed(2023)
B = 10000
boot = bootstrap(x = sample1,         # x: initial sample
                 n = B,               # n: number of bootstrap replicates,
                 theta_hat)           # the function

# the bootstrap replicates are saved in 'thetastat'
# which is an object returned by the function 'bootstrap'

# estimate of the bias
bias <- mean(boot$thetastar) - theta_hat(sample1)  # the bias estimate
bias # [1] -0.01236049

# estimate of the standard error
sdthetastar <- sd(boot$thetastar)
sdthetastar # [1] 0.04466796
```

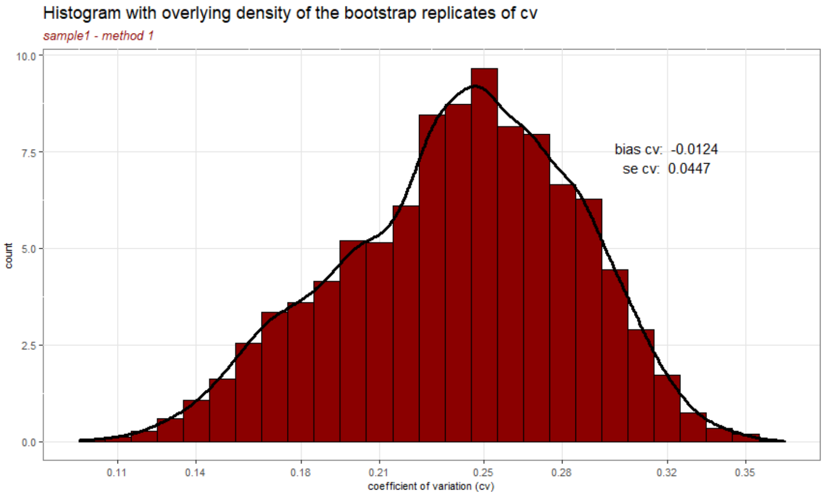
Method 1: histogram of replicates of cv



Method 2 the R language

```r
#---------------------------------------------------
# Perform bootstrap to estimate the cv - mehthod 2
#---------------------------------------------------
set.seed(2023)
B = 10000                                             # B: set the number of bootstrap replicates
bootstrap_object <- matrix(rep(0, B*length(sample1)),
                           nrow = B)
cv <- numeric(B)

# perform the bootstrap using the function sample() with replacement
for(i in 1:B) {

  bootstrap_object[i,] <- sample(sample1, size = length(sample1), replace = TRUE)

  cv[i] <- sd(bootstrap_object[i, ]) / mean(bootstrap_object[i, ])
}

# estimate of the bias
bias <- mean(cv) - (sd(sample1)/mean(sample1))
bias # [1] -0.01255868

# estimate of the standard error
sd <- sd(cv)
sd # [1] 0.0452905
```

Method 2: histogram of replicates of cv



CI for the median in Python

```
import pandas as pd
import numpy as np
from scipy.stats import bootstrap

# import csv file
sample1 = pd.read_csv("C:/Users/julia/OneDrive/Desktop/github/sample1.csv")
sample1

# or write data
sample1 = np.array([8.26, 6.33, 10.4, 5.27, 5.35, 5.61, 6.12, 6.19, 5.2, 7.01,
                   8.74, 7.78, 7.02, 6, 6.5, 5.8, 5.12, 7.41, 6.52, 6.21, 12.28,
                   5.6, 5.38, 6.6, 8.74], dtype = float)

sample1 = (sample1,)
B = 10000   # set the number of boostrap replicates

#calculate 95% bootstrapped confidence interval for median
bootstrap_ci = bootstrap(sample1, np.median, random_state=B, method='percentile'
    )

#view 95% boostrapped confidence interval
bootstrap_ci.confidence_interval
# ConfidenceInterval(low=5.8, high=7.02)
```

References

Rizzo, M.L. (2019). Statistical Computing with R, Second Edition (2nd ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9780429192760

Efron, B. and Tibshirani, R. J. (1993), An introduction to the Bootstrap, Chapman Hall.

The R Project for Statistical Computing: https://www.r-project.org/

Python: https://www.python.org/

course notes