Module 10: Bootstrap

Yuan Tian

07/27/2023

Applying bootstrap

The following code generates (X_i, Y_i) pairs.

```
library(MASS)
generate_pairs <- function(n) {
  # Generate n pairs of financial returns.
  muX <- 2
  muY <- -1
  CovMx <- matrix(c(1, -.25, -.25, 2), nrow = 2)
  data <- mvrnorm(n = 100, mu = c(muX,muY), Sigma = CovMx)
  return(data.frame('X' = data[, 1],
                    'Y' = data[, 2]))
}
```

```
fin_pairs <- generate_pairs( 100 ); # Generate 100 (X,Y) pairs.
head(fin_pairs)
```

```
##          X           Y
## 1 1.142239 -0.28536471
## 2 2.422669 -2.88259996
## 3 2.337702 -0.22304923
## 4 1.412434 -0.09437044
## 5 3.176583  0.44947354
## 6 2.332156 -4.30400271
```

We are interested in

$$\hat{\alpha} = \frac{\hat{\sigma}_Y^2 - \hat{\sigma}_{XY}}{\hat{\sigma}_X^2 + \hat{\sigma}_Y^2 - 2\hat{\sigma}_{XY}}$$

```
Sigmahat <- cov(fin_pairs)
Sigmahat
```

```
##            X          Y
## X  1.0988392 -0.2587532
## Y -0.2587532  2.2063904
```

```
sigma2hatXX <- Sigmahat[1,1]
sigma2hatYY <- Sigmahat[2,2]
sigmahatXY <- Sigmahat[1,2]
```

The $\hat{\alpha}$ is

```
alphahat <- (sigma2hatYY - sigmahatXY)/(sigma2hatXX + sigma2hatYY -2*sigmahatXY)
alphahat
```

```
## [1] 0.6448637
```

While the true value of alpha is

```
sigma2XX <- 1
sigma2YY <- 2
sigmaXY <- -0.25
alpha_true <-(sigma2YY - sigmaXY)/(sigma2XX + sigma2YY -2*sigmaXY)
alpha_true
```

```
## [1] 0.6428571
```

Now, again, we're going to resample with replacement from our data, and compute our statistic $\hat{\alpha}$ on each resample. The hope is that these resampled versions of the statistic will resemble the distribution of the statistic evaluated on the original data.

1. Create a function to compute alphahat from a given data set.
2. Resample the data $B = 200$ times, evaluating $\hat{\alpha}$ on each resample. Then, we'll use those resampled values to estimate the variance.
3. Create the confidence interval at the estimate.