ST 502 HW 3 Chapter 7 Problem 65

Bruce Campbell

January 29, 2017

(a) Suppose that truly our population follows a $N(\mu = 3, \sigma^2 = 4)$ distribution. We want to inspect how well certain CIs for μ will work. Generate a random sample of size n=5 from the population. For that sample, save the MOM (also the MLE here) estimate of μ , $\hat{\mu} = \bar{Y}$, and the unbiased estimate of σ^2

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \bar{Y})^2$$

```r
library(pander)
library(plyr)
library(dplyr)
library(readr)

mu <- 3
sigma <- 2
sampleSize <- 5
sample5 <- rnorm(5, mu, sigma)

muHat <- mean(sample5)

S2 <- var(sample5)

# Make sure R is using unbiased estimator sum(sample5^2 -
# muHat^2)/(sampleSize-1)
```

(b) Create and save a 95% CI for μ using only the data you have (i.e. you must use the t-distribution).

Since our sample is from a normal distribution we have a pivotal quantity from which we can make a probaility statement that will be the basis of our confidence interval.

$$\frac{\sqrt{n}(\bar{(Y)} - \mu}{S} \sim t_{n-1}$$

Letting $t(\frac{\alpha}{2})$ be the $frac\alpha 2$ quantile and noting that $t(\frac{\alpha}{2}) = t(1 - \frac{\alpha}{2})$ by symmetry, we have

$$P(\bar{Y} - \frac{S}{\sqrt{n}}t(\frac{\alpha}{2}) \leq \mu \leq \bar{Y} + \frac{S}{\sqrt{n}}t(\frac{\alpha}{2})) = 1 - \alpha$$

```r
S <- sqrt(S2)
alpha <- 0.05
talpha <- qt(1 - alpha/2, df = sampleSize - 1)

CI <- c(muHat - S/sqrt(sampleSize) * talpha, muHat + S/sqrt(sampleSize) * talpha)

pander(data.frame(CI), caption = "95% CI")
```

Table 1: 95% CI

CI
0.4318
5.481

(c) We will also create a bootstrap percentile interval for μ. For the RS of size five, use your observed MOM estimators to bootstrap $B = 500$ data sets (parametric bootstrap). Save the mean for each bootstrapped data set.

(d) Create a 95% CI for μ using the bootstrap percentile method.

```
sampleSize <- 5
numSamples <- 50
# Define a function to run the simulations
run.CI <- function(sampleSize, numSamples) {
    CIs <- data.frame(matrix(ncol = 2, nrow = numSamples))
    colnames(CIs) <- c("Xbar", "s2")
    for (i in 1:numSamples) {
        srs <- rnorm(sampleSize, mean = muHat, sd = sqrt(S2))
        CIs[i, ]$Xbar <- mean(srs)
        CIs[i, ]$s2 <- var(srs)
    }
    return(CIs)
}

df <- run.CI(sampleSize, numSamples)

rightCI <- quantile(df$Xbar, 0.975)
leftCI <- quantile(df$Xbar, 0.025)

pander(data.frame(left = as.numeric(leftCI), right = as.numeric(rightCI)), caption = " 95% CI from boots
```

Table 2: 95% CI from bootstrap quantiles

left	right
1.453	5.117

(e) Repeat the above steps a total of N=5000 times.

(f) At the bottom of your file, report the proportion of intervals that contained the true value of $\mu = 3$ for each sample size and method.

```
sampleSize <- 5
numCIs <- 500

run.BS <- function(sampleSize, numCIs) {
    BSCIs <- data.frame(matrix(ncol = 4, nrow = numCIs))
    colnames(BSCIs) <- c("left", "right", "containsP", "CILength")

    for (i in 1:numCIs) {
```

```r
        df <- run.CI(sampleSize, numSamples)
        BSCIs[i, ]$right <- quantile(df$Xbar, 0.975)
        BSCIs[i, ]$left <- quantile(df$Xbar, 0.025)
        BSCIs[i, ]$containsP <- mu >= BSCIs[i, ]$left & mu <= BSCIs[i, ]$right
        BSCIs[i, ]$CILength <- BSCIs[i, ]$right - BSCIs[i, ]$left
    }
    return(BSCIs)

}

df <- run.BS(sampleSize = sampleSize, numCIs = numCIs)

proportion_n5_B500_quantile <- sum(df$containsP)/numCIs
meanCILength_n5_B500_quantile <- mean(df$CILength)

print(proportion_n5_B500_quantile)
```

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

```r
print(meanCILength_n5_B500_quantile)
```

```
## [1] 3.280724
```

(f) Repeat all of the above for n = 20 and n = 100.

```r
sampleSize <- 20


df <- run.BS(sampleSize = sampleSize, numCIs = numCIs)

proportion_n20_B500_quantile <- sum(df$containsP)/numCIs
meanCILength_n20_B500_quantile <- mean(df$CILength)

print(proportion_n20_B500_quantile)
```

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

```r
print(meanCILength_n20_B500_quantile)
```

```
## [1] 1.628606
```

n=100

```r
sampleSize <- 100


df <- run.BS(sampleSize = sampleSize, numCIs = numCIs)


proportion_n100_B500_quantile <- sum(df$containsP)/numCIs
meanCILength_n100_B500_quantile <- mean(df$CILength)

print(proportion_n100_B500_quantile)
```

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

```r
print(meanCILength_n100_B500_quantile)
```

```
## [1] 0.7333343
```

(h) Report the mean CI length for each sample size and method.

```r
pander(data.frame(meanCILength_n5_B500_quantile = meanCILength_n5_B500_quantile,
    meanCILength_n20_B500_quantile = meanCILength_n20_B500_quantile, meanCILength_n100_B500_quantile = r
    caption = "length")
```

Table 3: length (continued below)

meanCILength_n5_B500_quantile	meanCILength_n20_B500_quantile
3.314	1.629

meanCILength_n100_B500_quantile
0.7333