Problem Set 9

Ian McGroarty

03APRIL2020

Problem 1:

Consider an atrifical dataset consisting of eight numebrs. let $\hat{\theta}$ be the 25% trimmed mean, computer by deleting the smallest two numbers and the largets two numbers, and then taking the average of the four remaining numbers. ## (a) Calculate $V\hat{ar}_B(\hat{\theta})$ for B = 25,100,00,500,1000,2000. From these results estimate the ideal bootstrap estimate $V\hat{ar}_\infty(\hat{\theta})$

So we need to draw B samples of size n=8 from X. Order the sample from smallest to largest. Take the average of the middle four numbers.

```
X1 <- c(1,2,3.5,4,7.3,8.6,12.4,13.8)

theta <- c()
bootstrap.var <- function(DATA,B){
    for (i in 1:B){
        # Make sure the data is in vector form
          vctrzdData <- as.vector(DATA)
        # Sample of size n with replacement from data
          btstrpSmpl = sample(vctrzdData,length(vctrzdData), replace = TRUE)
        # Sort
          btstrpSmpl.ordered <- sort(btstrpSmpl)
        # Take the mean of the middle 4 numbers
          theta[i] <- mean(btstrpSmpl.ordered[3:6])
      }
  # Take the mean of the B thetas
    theta.mean <- mean(theta)
  # Take the variance of the B thetas (var(theta) also works)
    theta.var <- (1/(B-1))*(sum((theta-theta.mean)^2))
  return(theta.var)
}

## The cool way to do it with a line
  #X <- c()
  #for (j in 1:200){ X[j] <- bootstrap.var(DATA=X1,j)}
  #plot(X,type = 'l', xlab="Bootstrap iterations",ylab= "Variance of theta")

# Run over 10 difference seeds
X1 <- c(1,2,3.5,4,7.3,8.6,12.4,13.8)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X1,j)))
    }
```

```
## [1] "B=25 Var(theta) = 5.39111041666667"
## [1] "B=100 Var(theta) = 4.11019292929293"
## [1] "B=200 Var(theta) = 3.71846543655779"
## [1] "B=500 Var(theta) = 4.9605442760521"
## [1] "B=1000 Var(theta) = 4.58574178616116"
## [1] "B=2000 Var(theta) = 4.68183452663832"
```

```
X2 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X2,j)))
    }
```

```
## [1] "B=25 Var(theta) = 14.1404284419534"
## [1] "B=100 Var(theta) = 8.97547708715917"
## [1] "B=200 Var(theta) = 9.63913477176873"
## [1] "B=500 Var(theta) = 9.30530630485229"
## [1] "B=1000 Var(theta) = 9.3659728221358"
## [1] "B=2000 Var(theta) = 9.1310836971426"
```

```
X3 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X3,j)))
    }
```

```
## [1] "B=25 Var(theta) = 2.51919649803716"
## [1] "B=100 Var(theta) = 1.77744912404241"
## [1] "B=200 Var(theta) = 2.26893334048123"
## [1] "B=500 Var(theta) = 2.17584269699146"
## [1] "B=1000 Var(theta) = 2.43192539064996"
## [1] "B=2000 Var(theta) = 2.23414089914593"
```

```
X4 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X4,j)))
    }
```

```
## [1] "B=25 Var(theta) = 8.90123868112373"
## [1] "B=100 Var(theta) = 7.70646162492471"
## [1] "B=200 Var(theta) = 7.68226006628201"
## [1] "B=500 Var(theta) = 7.11376095677702"
## [1] "B=1000 Var(theta) = 7.4933980170208"
## [1] "B=2000 Var(theta) = 7.67959546394477"
```

```
X5 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X5,j)))
    }
```

```
## [1] "B=25 Var(theta) = 2.1342783131009"
## [1] "B=100 Var(theta) = 1.68924747094572"
## [1] "B=200 Var(theta) = 1.95025420572124"
## [1] "B=500 Var(theta) = 1.74652360660902"
## [1] "B=1000 Var(theta) = 1.71427541909319"
## [1] "B=2000 Var(theta) = 1.58543807675857"
```

```
X6 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X6,j)))
    }
```

```
## [1] "B=25 Var(theta) = 3.36795083572854"
## [1] "B=100 Var(theta) = 2.50939970813516"
## [1] "B=200 Var(theta) = 3.37442102662124"
## [1] "B=500 Var(theta) = 3.57529283137123"
## [1] "B=1000 Var(theta) = 3.477231239425"
## [1] "B=2000 Var(theta) = 3.11477357102632"
```

```
X7 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X7,j)))
    }
```

```
## [1] "B=25 Var(theta) = 1.4927852211157"
## [1] "B=100 Var(theta) = 2.30549728804673"
## [1] "B=200 Var(theta) = 1.87986745777868"
## [1] "B=500 Var(theta) = 1.93050107846122"
## [1] "B=1000 Var(theta) = 1.79408523712013"
## [1] "B=2000 Var(theta) = 1.84956892605942"
```

```
X8 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X8,j)))
    }
```

```
## [1] "B=25 Var(theta) = 4.67179119404421"
## [1] "B=100 Var(theta) = 5.04541116522392"
## [1] "B=200 Var(theta) = 5.63606048535886"
## [1] "B=500 Var(theta) = 4.81942582920986"
## [1] "B=1000 Var(theta) = 5.34362341717113"
## [1] "B=2000 Var(theta) = 5.57616428578002"
```

```
X9 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X9,j)))
    }
```

```
## [1] "B=25 Var(theta) = 8.71394326244225"
## [1] "B=100 Var(theta) = 11.7678631092231"
## [1] "B=200 Var(theta) = 11.1680112064402"
## [1] "B=500 Var(theta) = 11.4899533674793"
## [1] "B=1000 Var(theta) = 11.2954553527634"
## [1] "B=2000 Var(theta) = 10.9522323401775"
```

```
X10 <- runif(n=8,min=1,max=20)
    for (j in c(25,100,200,500,1000,2000)){
      print(paste0("B=",j," Var(theta) = ",bootstrap.var(DATA=X10,j)))
    }
```

```
## [1] "B=25 Var(theta) = 14.9570539546351"
## [1] "B=100 Var(theta) = 8.96474387872754"
## [1] "B=200 Var(theta) = 9.17276100826375"
## [1] "B=500 Var(theta) = 8.3583572387775"
## [1] "B=1000 Var(theta) = 7.88212177818985"
## [1] "B=2000 Var(theta) = 8.78628950481153"
```

Problem 2

Patients with advanced terminal cancer of the breast were treated with ascorbate in an attempt to prolong survival.

(a) Construct a 95% confidence interval

of the the mean breast cancer survival time by apply the simple bootstrap to logged data and exponentiating the resutling interval boundaries.

```
B <- 200

Survival = c(25, 42, 45, 46, 51, 103, 124, 146, 340, 396, 412, 879, 1112)
survival.log <- log(Survival)

## DONT FORGET TO EMPTY THETA
  theta <- c()

    for (i in 1:200){
        # Make sure the data is in vector form
          vctrzdData <- as.vector(survival.log)
        # Sample of size n with replacement from data
          btstrpSmpl = sample(vctrzdData,length(vctrzdData), replace = TRUE)
        # Take the mean
          theta[i] <- mean(btstrpSmpl)
      }
  # Take the mean of the B thetas
    theta.mean <- mean(theta)
  # 95% confidence interval for 200 observations mean the 5th and 195th observations
    theta.ordered <- sort(theta)
    print(paste0(
      "The 95% confidence interval for the logged survival data with 200 bootstrap iterations is
"
      ,exp(theta.ordered[c(5)]),",",exp(theta.ordered[c(195)])))
```

```
## [1] "The 95% confidence interval for the logged survival data with 200 bootstrap iterations i
s 74.9493814679739,273.344323503873"
```

(b) Construct another 95% using original data

```
B <- 200

Survival = c(25, 42, 45, 46, 51, 103, 124, 146, 340, 396, 412, 879, 1112)
#survival.log <- log(Survival)

## DONT FORGET TO EMPTY THETA
  theta <- c()
    for (i in 1:B){
        # Make sure the data is in vector form
          vctrzdData <- as.vector(Survival)
        # Sample of size n with replacement from data
          btstrpSmpl = sample(vctrzdData,length(vctrzdData), replace = TRUE)
        # Take the mean
          theta[i] <- mean(btstrpSmpl)
      }
  # Take the mean of the B thetas
    theta.mean <- mean(theta)
  # 95% confidence interval for 200 observations mean the 5th and 195th observations
    theta.ordered <- sort(theta)
    print(paste0(
      "The 95% confidence interval for the raw survival data with 200 bootstrap iterations is "
      ,theta.ordered[c(5)],",",theta.ordered[c(195)]))
```

```
## [1] "The 95% confidence interval for the raw survival data with 200 bootstrap iterations is 1
24.692307692308,495.846153846154"
```

Problem 3

Assume $X_1 \cdots X_n \ \sim Unif(0, \theta)$. What is the maximum likelihood estimate of $$.

For reference: (Marx & Larsen (2018), p. 281). This is a strange question since the uniform distribution will have an equal probablity $(p_X = \frac{1}{\theta})$, for all numbers in the support of X_n. That means that the maximum likelihood estimate of $\theta = \theta$. Or more formally,

$$L(\theta) = \prod_{i=1}^{n} \frac{1}{\theta} = \theta^{-n}$$
$$\frac{dlnL(\theta)}{d\theta} = \frac{d}{d\theta} - nln(\theta) = \frac{-n}{\theta}$$
$$\theta = \theta_e = x_{max}$$

So the question becomes, how do estimate θ if it is simply the max of the sample?

Problem 4

Suppose $X^* = (x_1^*, \cdots, x_n^*)$ is a bootstrap sample obtained from $x = (x_1, \cdots, x_n)$. Show that the probability that any particular value of x is in x^* exactly k times.

So we have a set $X = (x_1, \cdots, x_n)$ and we draw a sample with replacement from x to obtain $X^* = (x_1^*, \cdots, x_n^*)$. This is equvalent to drawing a uniform random variable from the support of X. Thus, each element of the set x is drawn with probability $\frac{1}{n}$.

Given $x_j \in X$ and $x_i^* \in X^*$, let $x_1^* x_n* $ be n independent trials, each resulting in either success or failure, with success defined as $x_i^* = x_j$. The probablity of k success is a binomial random variable with the probability of success for any trial $p = \frac{1}{n}$. Therefore, by Theorem 3.2.1 (Larsen & Marx (2018), p. 104):

$$\mathrm{P(k\ successes)} = \binom{n}{k}(\frac{1}{n})^k(\frac{n-1}{n})^{n-k}$$

We can see this in R by calculating the theoretical probability and creating a bootstrap sample for

$$\hat{\theta}_i^* = \sum_{i=1}^{n} I_{(x_j^* = x_i)}$$

Where $I(\phi)$ is a Binomial random variable with success probability $$

```
## Define nchoosek (Larsen & Marx (2018), p. 84).
  nchoosek <- function(n,k){
    factorial(n)/((factorial(k))*(factorial(n-k)))
  }

## Define full function (Larsen & Marx (2018), p. 104)
 binom.nk <- function(n,k){
   nchoosek(n,k) * ((1/n)^k) * ((n-1)/n)^(n-k)
 }


## Get a base X of size n
 vctrzdData <- c(1,2,3.5,4,7.3,8.6,12.4,13.8)

## Set parameters
 n <- length(vctrzdData)
 k <- 2
 B <- 1000

## Boot Strap
  count <- c()
    for (i in 1:B){
      ## Get B samples from X
        btstrpSmpl = sample(vctrzdData,length(vctrzdData), replace = TRUE)
      ## Count how many times x_i shows up in the sample
        count[i] <- sum(btstrpSmpl == vctrzdData[3])
    }
## Probability
  (sum(count == k))/B
```

```
## [1] 0.211
```

```
## Compare to theoretical probability
  binom.nk(length(vctrzdData),k)
```

```
## [1] 0.196348
```

Problem 5

We know that $\hat{\theta} = \bar{X}$ is an unbiased estimate of the mean for a particular density. So the true bais, $Bias_{True}(\hat{\theta}) = 0.$. Prove that

Just for the record, the notation here is really throwing me off.

(a) $Bias_{True}(\hat{\theta}) = 0$

If $\hat{\theta} = \bar{X}$ is an unbiased estimate for $E(\theta)$ then by Definition 5.4.1 (Larsen & Marx (2018) p. 310) that $E(\hat{\theta}) = \theta$. Thus,

$$bias(\hat{\theta}) = E(\hat{\theta}) - \theta = \theta - \theta = 0$$

(b) $\hat{Bias}_{Revised}(\hat{theta}) = 0$

$$\begin{aligned} \hat{Bias}_{revised}(\hat{\theta}) &= \hat{\theta}^*(\cdot) - \hat{\theta}_{revised} \\ &= \frac{1}{B}\sum_{j=1}^{B}\hat{\theta}_i^* - g(\bar{P}^* x^T) \\ &= \frac{1}{B}\sum_{j=1}^{B} g(x_j^*) - g(\frac{1}{B}\sum_{j=1}^{B} P_j^* x^T) \\ \text{As } lim_{B\to\infty} P_j^* \to (\frac{1}{n}) &\implies P_j^* x^T = x_j \cdot (\frac{1}{n}) \end{aligned}$$

Since x_j is continuously sampled from the empirical distribution function, these two terms are equvalent.

(c) show that it’s not necessarily the case that $\hat{Bias}(\hat{\theta}) = 0$.

Since we have shown that $\hat{\theta}^*(\cdot) = \hat{\theta}_{revised}$ it suffices to show that $\hat{\theta}_{revised} \neq \hat{\theta}$ Which was shown in problems one and two which showed that $\hat{\theta}$ had non-zero variance and depended heavily on the size of B. So while they may be assymtotically equal, they are not equal by construction.