Problem Set 5 Corrections

October 27, 2020

[20]:
```
library(multtest)
data(golub)
gene1.values = golub[1,]
gene1.values[1:5]
options(warn=-1)
```

1. -1.45769 2. -1.3942 3. -1.42779 4. -1.40715 5. -1.42668

1 Problem 1

I did this problem mostly correct. I used the right formula but yielded the wrong values. I also didn't apply CLT for some of the last questions. Correct approach to CI, but wrong bounds.

2 Problem 2

Part 1 (a) is correct.

Part b is inccorect i accidently put sd in the numerator instead of the MLEs.

Part c i didnt get to on time but understand now.

Part d i did correct!

Part e i didnt get but understand now.

3 Problem 3

I got this correct.

4 Problem 4

I did this 100% correct.

4.a)

$$If E[x] = \mu = \frac{\alpha}{3}, thus\,, by\ random\ sampling\ , E\bar{x} = E[x] = \mu \therefore E[3\bar{x}] = 3\mu = \alpha \tag{1}$$

4.b)

$$E[x^2] - \mu^2 = (\frac{1}{3} - \frac{\alpha^2}{9}), thus\ the\ variance\ of\ [3\bar{x}] = 9Var[x]/n = \frac{(3-\alpha^2)}{n} \tag{2}$$

4.c)

$$\hat{\alpha} \tag{3}$$

will be asymptotically normally distributed with mean of and var of (3- ^2)/n.

$$var = (3-1^2)/20 = 2/20 = 1/10 mean = 1 \tag{4}$$

[14]:
```
#4.c) calculation
pnorm(.5, mean = 1, sd = sqrt(1/10), lower.tail = FALSE) #.5 to infinity
```

0.943076850996671

5 Problem 5

I did mostly every thing right, messed up some little bootstrap stuff towards the end.

5.a)

$$\frac{\partial}{\partial\theta}I(\theta) = \frac{-2n_1+n_2}{1-\theta} + \frac{2n_3+n_2}{\theta} 0 = \frac{-2n_1+n_2}{1-\theta} + \frac{2n_3+n_2}{\theta} \theta_{MLE} = \frac{2n_3+n_2}{2n_1+2n_2+2n_3} = \frac{2*112+68}{2*190} = .76842 \tag{5}$$

5.b)

$$Var(\theta_{MLE}) \xrightarrow{P} \frac{1}{nl(\theta_{MLE})} = \frac{1}{190(.76642)(1-.76642)} = \frac{1}{190(.17795)} = .03 \tag{6}$$

5.c)

$$I(\theta_{MLE}) = \frac{2n}{\theta_{MLE}(1-\theta_{MLE})} = \frac{2*190}{(.76642)(1-.76642)} = 2,135.42 \tag{7}$$

$$CI = (\theta_{MLE} - \frac{Z(\alpha/2)}{\sqrt{l(\theta_{MLE})}}\ ,\ \theta_{MLE} + \frac{Z(\alpha/2)}{\sqrt{l(\theta_{MLE})}}) = (.76842 - \frac{2.576}{\sqrt{2,135.42}}\ ,\ .76842 + \frac{2.576}{\sqrt{2,135.42}}) = (.71267\ ,\ .82416) \tag{8}$$

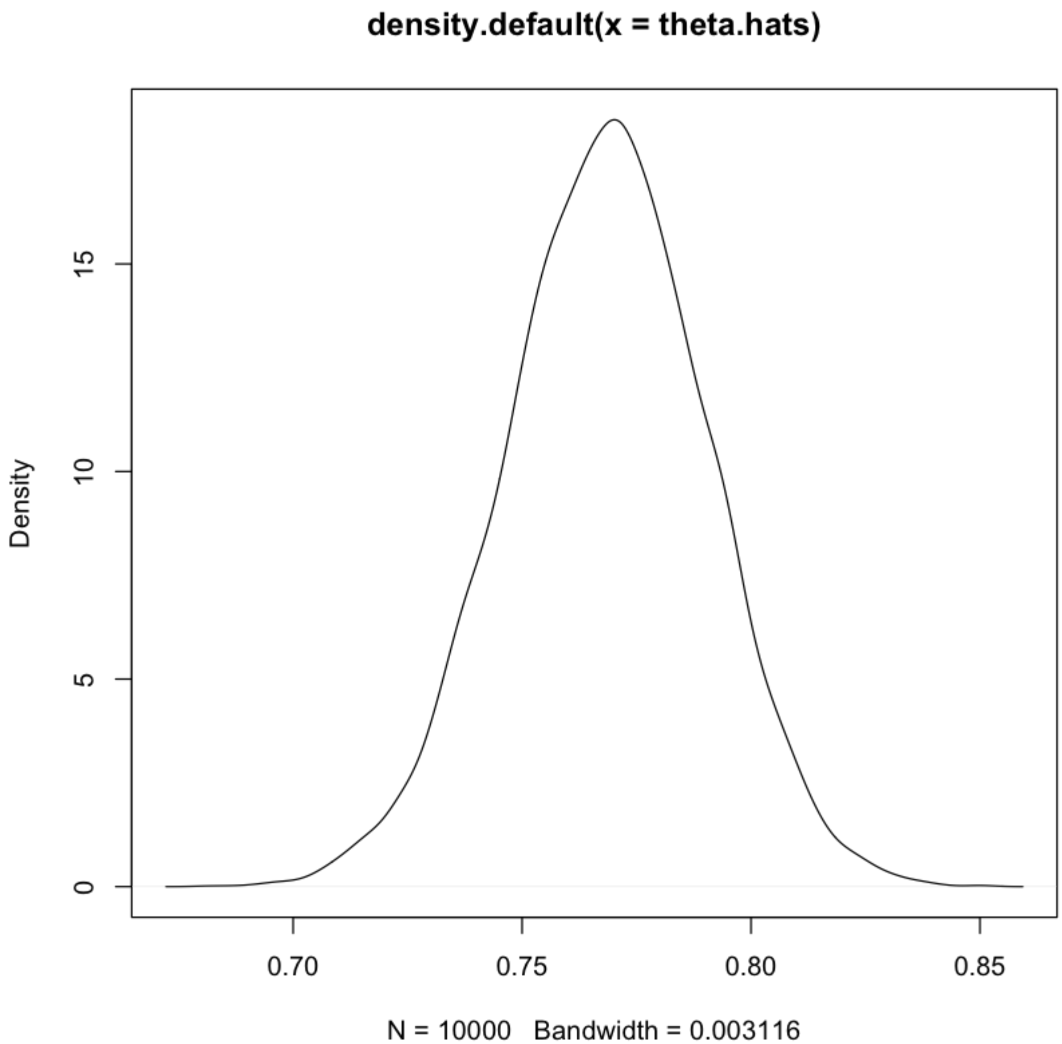
[15]:
```
#5d
sim.samples = rmultinom(10000, size=190, prob=c(0.0526, 0.35789, 0.58947)) #␣
↪prob = 10/190,68/190, 112/190
sim.samples[,1:10]

thetaMLE = function(n1, n2, n3) {
    return ((2*n3 + n2)/(2*(n1+n2+n3)))
}

theta.hats = apply(sim.samples, 2, function(x) {
    return (thetaMLE(x[1], x[2], x[3]))
 })
```

	11	9	9	9	15	5	10	12	9	13
A matrix: 3×10 of type int	59	63	67	78	62	87	67	69	71	51
	120	118	114	103	113	98	113	109	110	126

[16]:
```
#5d
theta.hats[1:5]
sd(theta.hats)
plot(density(theta.hats))
```

1. 0.786842105263158 2. 0.786842105263158 3. 0.776315789473684 4. 0.747368421052632 5. 0.757894736842105

0.0218459164262661



5d) The bootstrap sample is actually really close to the MLE done asymptotic. My asymptotic

MLE was .76842 and my bootstrap MLE (.768472) is very close to that!

[17]:
```
mean(theta.hats)
```

0.768278157894737

[18]:
```
#5e
denom = ((mean(theta.hats)) * (1 - (mean(theta.hats)))) #denominator
newd = (denom*190) #multiply by n
var = 1/newd #finish equation
print(var)
```

```
[1] 0.02956385
```

5e) our variance rounds also to .03

[19]:
```
#5f
exactCI = function(values, alpha) {
 x.bar = mean(values)
         n = length(values)
 t.n_1 = -qt(alpha/2, df=n-1)
         S = sd(values)
 CI = c(x.bar - (S*t.n_1)/sqrt(n),
 x.bar + (S*t.n_1)/sqrt(n))
 return(CI)
}
 (exact.CI = exactCI(theta.hats, alpha=0.01))
```

1. 0.767715336941264 2. 0.76884097884821

5f) our 99% CI using bootstrap is tighter than our approximation in our part 5c. Both our CI's contain our null value.