Project_2_Part_2

Carlos Sathler

3/13/2019

Contents

Per class lecture the percentile bootstrap "only has much theoretical support when the sampling distribution of the estimator is unbiased and symmetric." So, we expect the residual bootstrap to have better coverage for skewed distribution and biased estimators.

We provide two examples to confirm our expectations. The first one tests the coverage of the mean estimator for a chi-square distribution with df=3. The sample mean is an unbiased estimator of the population mean. So, in our first example we explore the impact of skewness in the coverage performance of the two bootstraps. We increase the skewness of the chi-square distributions (df=3) by increasing the size of the sample from 10, to 100, to 1,000.

```
set.seed(100)
library(e1071)
# skewness increases for chi-square disty as n increases
mean(replicate(1000,skewness(rchisq(10,3))))
```

```
## [1] 0.7007614
```

```
mean(replicate(1000,skewness(rchisq(100,3))))
```

```
## [1] 1.47595
```

```
mean(replicate(1000,skewness(rchisq(1000,3))))
```

```
## [1] 1.618732
```

Clearly, skewness increases with sample size.

Now, for the simulation:

```
# code adapted from example from class lectures

set.seed(100)

library(boot)
bootmean = function(x, indices){
  # returns sample mean
  return(mean(x[indices]))
}

exp.competition = function(n, reps){
  x = rchisq(n,3)
  x.bar = 3
  bootmean.dist = boot(x, bootmean, R = reps)
  cis = boot.ci(bootmean.dist, type=c("perc","basic"))
  perc = (cis$perc[4]<=x.bar) & (cis$perc[5]>=x.bar)
  basic = (cis$basic[4]<=x.bar) & (cis$basic[5]>=x.bar)
  return(c(perc, basic))
}
```

```r
for (i in 1:3) {
  print(10^i)
  results = replicate(1000, exp.competition(10^i, 1000))
  scores = apply(results, 1, sum)
  names(scores) = c("perc", "basic")
  # print scores as percentage
  print(scores/1000*100)
}
```

```
## [1] 10
##  perc basic
##  86.1  84.1
## [1] 100
##  perc basic
##  95.2  93.9
## [1] 1000
##  perc basic
##  94.1  95.0
```

The results show that for n=10 and n=100 the percentile bootstrap has better coverage than the residual bootstrap ('basic'). However, for n=1000, the skewness of the chi-square (df=3) will cause the residual bootstrap to come closer to the nominal level of coverage by 0.9 percent. Similar results were obtained with different seed values.

Next we show two examples that demonstrate the coverage for the residual bootstrap outperforms the coverage of the percentual bootstrap for a biased estimator. We estimate the standard deviation of a normal distribution N(0,1) first with a unbiased estimator, and then with a biased one. We use normal distributions because they are symetric by nature.

```r
mean(replicate(1000,skewness(rnorm(10, 0, 1))))
```

```
## [1] -0.00867698
```

```r
mean(replicate(1000,skewness(rnorm(100, 0, 1))))
```

```
## [1] 0.002183437
```

```r
mean(replicate(1000,skewness(rnorm(1000, 0, 1))))
```

```
## [1] 0.0009121219
```

Skewness is close to zero regardless of sample size. The distributions are mostly symetric.

We estimate the standard deviation of a normal distribution N(0,1) using the R sd function which has N-1 in the denominator and is an unbiased estimator of the population sd.

```r
# code adapted from example from class lectures

set.seed(100)

bootsd = function(x, indices){
  # returns sample mean
  return(sd(x[indices]))
}

exp.competition = function(n, reps){
  x = rnorm(n, 0, 1)
  x.sd = 1
```

```
  bootmean.dist = boot(x, bootsd, R = reps)
  cis = boot.ci(bootmean.dist, type=c("perc","basic"))
  perc = (cis$perc[4]<=x.sd) & (cis$perc[5]>=x.sd)
  basic = (cis$basic[4]<=x.sd) & (cis$basic[5]>=x.sd)
  return(c(perc, basic))
}

for (i in 1:3) {
  print(10^i)
  results = replicate(1000, exp.competition(10^i, 1000))
  scores = apply(results, 1, sum)
  names(scores) = c("perc", "basic")
  # print scores as percentage
  print(scores/1000*100)
}
```

```
## [1] 10
##  perc basic
##  76.8  83.1
## [1] 100
##  perc basic
##  93.2  93.3
## [1] 1000
##  perc basic
##  94.5  94.5
```

For the unbiased estimator the residual bootstrap only performed meaningfully better for sample size N = 10. Beyond that sample size, the percentual bootstrap performance is comparable to the residual bootstrap performance.

Next, we estimate the standard deviation of the normal distribution N(0,1) using a custom sd function with N in the denominator, i.e., we use a biased estimator of the population sd.

```
# code adapted from example from class lectures

set.seed(100)

bootsd = function(x, indices){
  # returns sample mean
  y = x[indices]
  n = length(y)
  return((sum((y - mean(y))^2)/n)^0.5)
}

exp.competition = function(n, reps){
  x = rnorm(n, 0, 1)
  x.sd = 1
  #x = rexp(n, 0.5); x.bar = 2
  bootmean.dist = boot(x, bootsd, R = reps)
  cis = boot.ci(bootmean.dist, type=c("perc","basic"))
  perc = (cis$perc[4]<=x.sd) & (cis$perc[5]>=x.sd)
  basic = (cis$basic[4]<=x.sd) & (cis$basic[5]>=x.sd)
  return(c(perc, basic))
}
```

```r
for (i in 1:3) {
  print(10^i)
  results = replicate(1000, exp.competition(10^i, 1000))
  scores = apply(results, 1, sum)
  names(scores) = c("perc", "basic")
  # print scores as percentage
  print(scores/1000*100)
}
```

```
## [1] 10
##  perc basic
##  71.1  81.8
## [1] 100
##  perc basic
##  92.3  93.2
## [1] 1000
##  perc basic
##  94.5  94.7
```

We notice now that the residual bootstrap outperforms the percentual bootstrap for all sample sizes, and "meaningfully" so for small and medium size samples (N = 10 and 100).

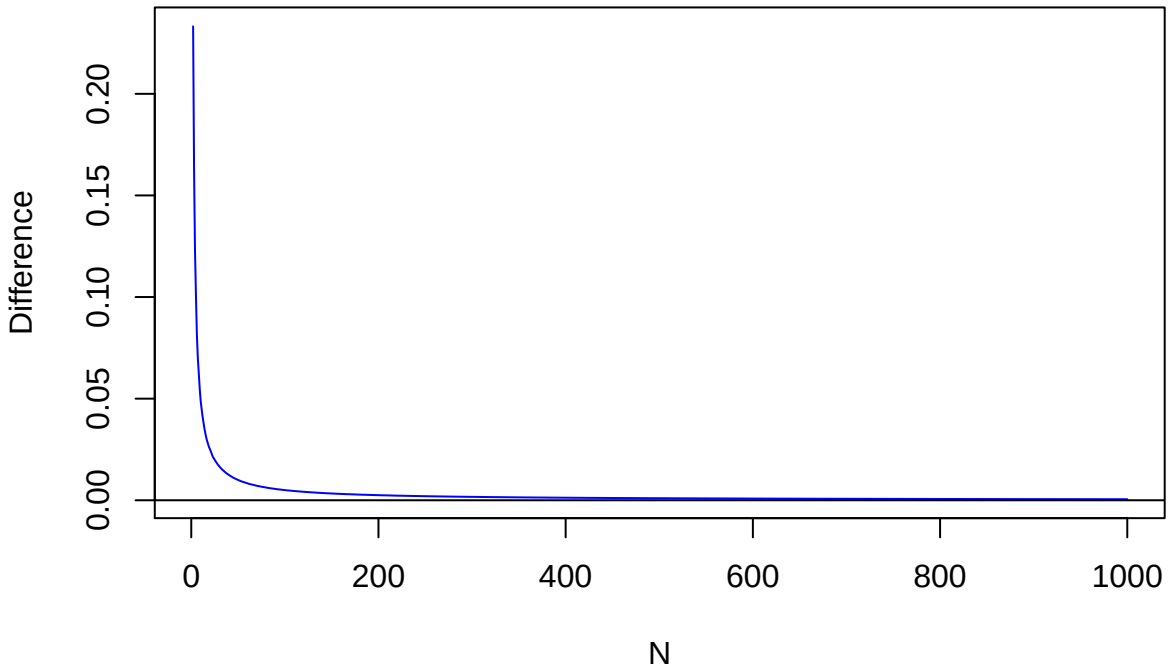
Note that we should not be surprised that the coverage of the two bootstraps converges for values of N > 100, since the difference between the biased and unbiased standard deviation drops quickly to zero for larger values of N.

See plot.

```r
set.seed(100)

get_sd_diff = function(x, n) {
  sd_biased = (sum((x - mean(x))^2)/n)^0.5
  sd_unbiased = sd(x)
  return(abs(sd_unbiased-sd_biased))
}

sd_diff = NULL
for (i in 2:1000) {
   sd_diff[i] = mean(replicate(1000, get_sd_diff(rnorm(i, 0, 1), i)))
}
plot(sd_diff, type='l', col = 'blue', main = 'Difference: Biased vs. Unbiased STD', xlab='N', ylab='Dif
abline(h=0)
```



In conclusion, we showed through two examples that the residual bootstrap is likely to have better coverage than the percentual bootstrap on skewed distributions, and for biased estimators, as suggested in the class materials.