Some questions about bootstrapping

T.Scofield

March 01, 2021

Some big-picture questions about bootstrapping

Does it really provide a good estimate of SE?

Standard error is the term given to the standard deviation of the sampling distribution. If we have access to the full population (which is rare), we can get a better approximation drawing our samples each time from the full population.

Example: Runs data from MLB 2018 data frame

```
mlb18 <- read.csv("http://scofield.site/teaching/data/csv/mlb18abEligible.csv")
```

We can sample 40 players at random using

```
mySample <- sample(mlb18, size=40)
head(mySample)     # first few rows of sampled player-data
```

```
##       X          name team position games  AB  R   H doubles triples HR RBI
## 267 267  Hernandez, T  TOR       LF   134 476 67 114      29       7 22  57
## 223 223    Renfroe, H   SD       RF   117 403 53 100      23       1 26  68
## 341 341     Kepler, M  MIN       RF   156 532 80 119      30       4 20  58
## 43   43       Soto, J  WSH       LF   116 414 77 121      25       1 22  70
## 14   14       Cain, L  MIL       CF   141 539 90 166      25       2 10  38
## 224 224   Pederson, J  LAD       LF   148 395 65  98      27       3 25  56
##     walks strike_outs stolen_bases caught_stealing_base   AVG   OBP   SLG   OPS
## 267    41         163            5                    5 0.239 0.302 0.468 0.771
## 223    30         109            2                    1 0.248 0.302 0.504 0.805
## 341    71          96            4                    5 0.224 0.319 0.408 0.727
## 43     79          99            5                    2 0.292 0.406 0.517 0.923
## 14     71          94           30                    7 0.308 0.395 0.417 0.813
## 224    40          85            1                    5 0.248 0.321 0.522 0.843
##     orig.id
## 267     267
## 223     223
## 341     341
## 43       43
## 14       14
## 224     224
```

Drawing such a sample can be immediately followed by a computation of the sample mean number of runs scored:

```
mean(~R, data=sample(mlb18, size=40))
```

```
## [1] 50.725
```

And, to simulate the sampling distribution, we repeat this calculation often.

```
manyXbarsFromPop <- do(5000) * mean(~R, data=sample(mlb18, size=40))
head(manyXbarsFromPop)
```

```
##     mean
## 1 49.300
## 2 50.150
## 3 36.200
## 4 48.300
## 5 44.025
## 6 46.600
```

At this point we can look at the histogram using command

```
gf_histogram(~mean, data=manyXbarsFromPop)
```

or obtain an approximate value for $\text{SE}_{\overline{x}}$ using

```
sd(~mean, data=manyXbarsFromPop)
```

```
## [1] 3.992975
```

That approach, while quite accurate, is generally not possible. But in this "laboratory"-like environment, where we have a set of data we are considering to be the full population, we see it as the target value we hope bootstrapping can reproduce.

In bootstrapping, we collect one random sample (for purposes of comparison, it also must be of size $n = 40$), and then draw repeated bootstrap samples from the original sample.

```
originalSamp <- sample(mlb18, size=40)
```

To obtain one bootstrap statistic involves

```
bstrapSamp <- resample(originalSamp)      # drawing
mean(~R, data=bstrapSamp)          # computing statistic
```

```
## [1] 48.725
```

This is what we must repeat often, always returning to `originalSamp`, and drawing with replacement.

```
manyBstrapXbars <- do(5000) * mean(~R, data=resample(originalSamp))
head(manyBstrapXbars)
```

```
##     mean
## 1 43.775
## 2 44.100
## 3 51.750
## 4 38.625
## 5 39.325
## 6 41.575
```

Once again, we can look at a histogram, or calculate directly the standard deviation of these bootstrapped $\overline{x}$-values. Let's cut to the chase and do the latter, seeing how close we have come to the target value.

```
sd(~mean, data=manyBstrapXbars)
```

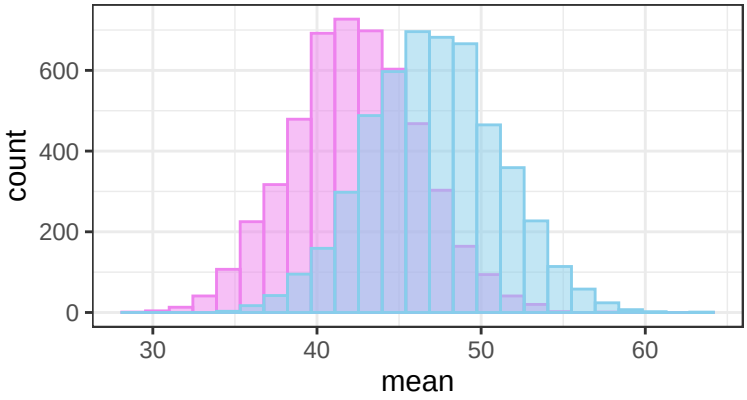
```
## [1] 3.889127
```

This value, 3.89, in comparison with our assumed-to-be-more-accurate 3.99, isn't too bad. Results *will* vary, as each simulation draws 5000 new bootstrap samples.

How alike are the sampling and bootstrap distributions?

Let's look at them with one laid on top of the other.

```
gf_histogram(~mean, data=manyBstrapXbars, color="violet", fill="violet") %>%
  gf_histogram(~mean, data=manyXbarsFromPop, color="skyblue", fill="skyblue", alpha=.5)
```



Note: There are *two* distributions here, not three. The violet-colored bars represent the bootstrap distribution, while the skyblue-colored ones are the simulated sampling distribution. Their similarities include

- shape and form
- spread (we already saw they have approximately the same standard deviations)

What differs is the **center**, the *mean* of the distribution. Since $\overline{x}$ is an unbiased estimator, the mean of its distribution will be (approximately)

- μ, the population mean, when samples are drawn *directly from the population*, as was done to get our "target" value.
- $\overline{x}$, the mean from the original sample, when *bootstrapping* (i.e, when further samples are drawn treating the original sample as a pseudo-population).