10
The bootstrap

Bootstrapping: standard errors through resampling

At the core of the resampling approach to statistical inference lies a simple idea. Most of the time, we can't feasibly take repeated samples of size n from the population, to see how our estimate changes from one sample to the next. But we can repeatedly take samples of size n *from the sample itself*, and apply our estimator afresh to each notional sample. The idea is that the variability of the estimates across all these samples can be used to approximate our estimator's true sampling distribution.

This process—pretending that our sample is the whole population, and taking repeated samples of size n with replacement from our original sample of size n—is called *bootstrap resampling*, or just *bootstrapping*.[1] Each block of n resampled data points is called a bootstrapped sample. To bootstrap, we write a computer program that repeatedly resamples our original sample and recomputes our estimate for each bootstrapped sample. Modern software makes a non-issue of the calculational tedium involved.

You may be puzzled by something here. There are n data points in the original sample. If we repeatedly resample n data points from our "pseudo-population" of size n, won't each bootstrapped sample be identical to the original sample? If so, and every bootstrapped sample looks the same, then how can this process be used to simulate sampling variability?

This fact highlights a key requirement of bootstrapping: the resampling must be done *with replacement* from the original sample, so that each bootstrapped sample contains duplicates and omissions from the original sample.[2] These duplicates and omissions induce variation from one bootstrapped sample to the next, mimicking the variation you'd expect to see across the real repeated samples that you can't take.

To summarize, let's say we have a data set D, consisting of n

[1] The term "bootstrapping" is a metaphor. It is an old-fashioned phrase that means performing a complex task starting from very limited resources. Imagine trying to climb over a tall fence. If you don't have a rope, just "pull yourself up by your own bootstraps."

[2] Imagine a lottery drawing, where there's a big urn with 60 numbered balls in it. We want to choose a random sample of 6 numbers from the urn. After we choose a ball, we could do one of two things: 1) put the ball to the side, or 2) record the number on the ball and then throw it back into the urn. If you set the ball aside, it can be selected only once; this is sampling without replacement, and it's what happens in a real lottery. But if instead you put the ball back into the urn, it has a chance of being selected more than once in the final sample; this is sampling with replacement, and it's what we do when we bootstrap.

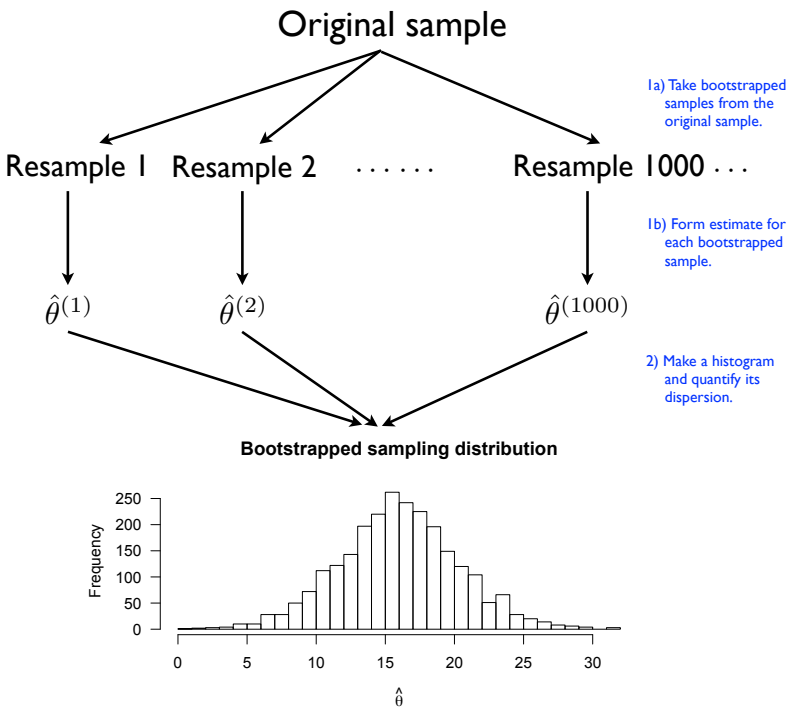


Figure 10.1: A stylized depiction of a bootstrapped sampling distribution of an estimator $\hat{\theta}$. We have a single original sample. We repeatedly take many bootstrapped samples (say, 1000) from the original sample (step 1a). For each resample, we compute the estimator $\hat{\theta}$ (step 1b). At the end, we combine all the estimates $\hat{\theta}^{(1)}, \ldots, \hat{\theta}^{(1000)}$ into a histogram of the bootstrapped sampling distribution, and we summarize the dispersion of that histogram (step 2). Compare with Figure 8.3.

cases. We want to understand how our estimator $\hat{\theta}$ might have behaved differently with a different sample of size n. To answer this question using bootstrapping, we follow two main steps.

(1) Repeat the following substeps many times (e.g. 1000 or more):

   a. Generate a new bootstrapped sample $D^{(r)}$ by taking n samples with replacement from D.

   b. Apply the estimator $\hat{\theta}$ to the bootstrapped sample $D^{(r)}$ and save the resulting estimate, $\hat{\theta}^{(r)}$.

(2) Take all of the $\hat{\theta}^{(r)}$'s you've generated and make a histogram. This is your estimate of the sampling distribution.

See Figure 10.1, and compare with Figure 8.3.

Resampling won't yield the true sampling distribution of an estimator, but it is often good enough for approximating the standard error (which you'll remember is just the standard deviation of the sampling distribution). We use the term *bootstrapped standard error* for the standard deviation of the bootstrapped sampling

distribution. The bootstrapped standard error is an estimate of the true standard error.

The quality of this estimate depends almost entirely on one thing: how closely the original sample resembles the wider population. This is a question of judgment best answered by someone with subject-area expertise relevant to the data set at hand. As a data analyst this often isn't under your control, and therefore it's almost worth remember that the bootstrap is not entirely free of assumptions. You can't magic your way to sensible estimates of the true sampling distribution by bootstrapping a biased, woefully small, or otherwise poor sample.

The quality of the Monte Carlo approximation also depends to a lesser extent on how many bootstrapped samples you take from the original sample. Simulating more bootstrapped samples help to reduce the variability inherent in any Monte Carlo simulation—up to a point. But taking more bootstrapped samples is never a substitute for having more actual samples in the real data set. Fundamentally, it is the size of your original sample that governs the precision of your estimates.

A natural question is: how well does bootstrapping work in practice? To see the procedure in action, let's reconsider the least-squares estimator of the slope (β_1) for the weight–volume line describing the fish in our hypothetical lake. The top row of Figure 10.2 shows three actual sampling distributions, corresponding to samples of size $n = 15$, $n = 50$, and $n = 100$ from the entire population. These were constructed using the Monte Carlo method described several pages ago, as depicted in Figures 8.2 and 8.3. For example, the top left panel (for $n = 15$) was constructed by taking 2,500 Monte Carlo samples from the true population in Figure 8.2, and computing the least-squares estimate of the slope for each sample as in Figure 8.3.

Below each true sampling distribution, we have focused on four of these 2500 samples. For each of these real samples, we ran the bootstrapping procedure by 2500 bootstrapped samples from the original sample of size n, treating it as a pseudo-population. For each bootstrapped sample, we compute the least-squares line for weight versus volume. These 2500 estimates of β_1 are what you see in each grey-colored panel of Figure 10.2. For example, the first grey panel in column 1 corresponds to the bootstrapped sampling distribution from the first sample of size 15; the second grey panel corresponds to the bootstrapped sampling distribution from the

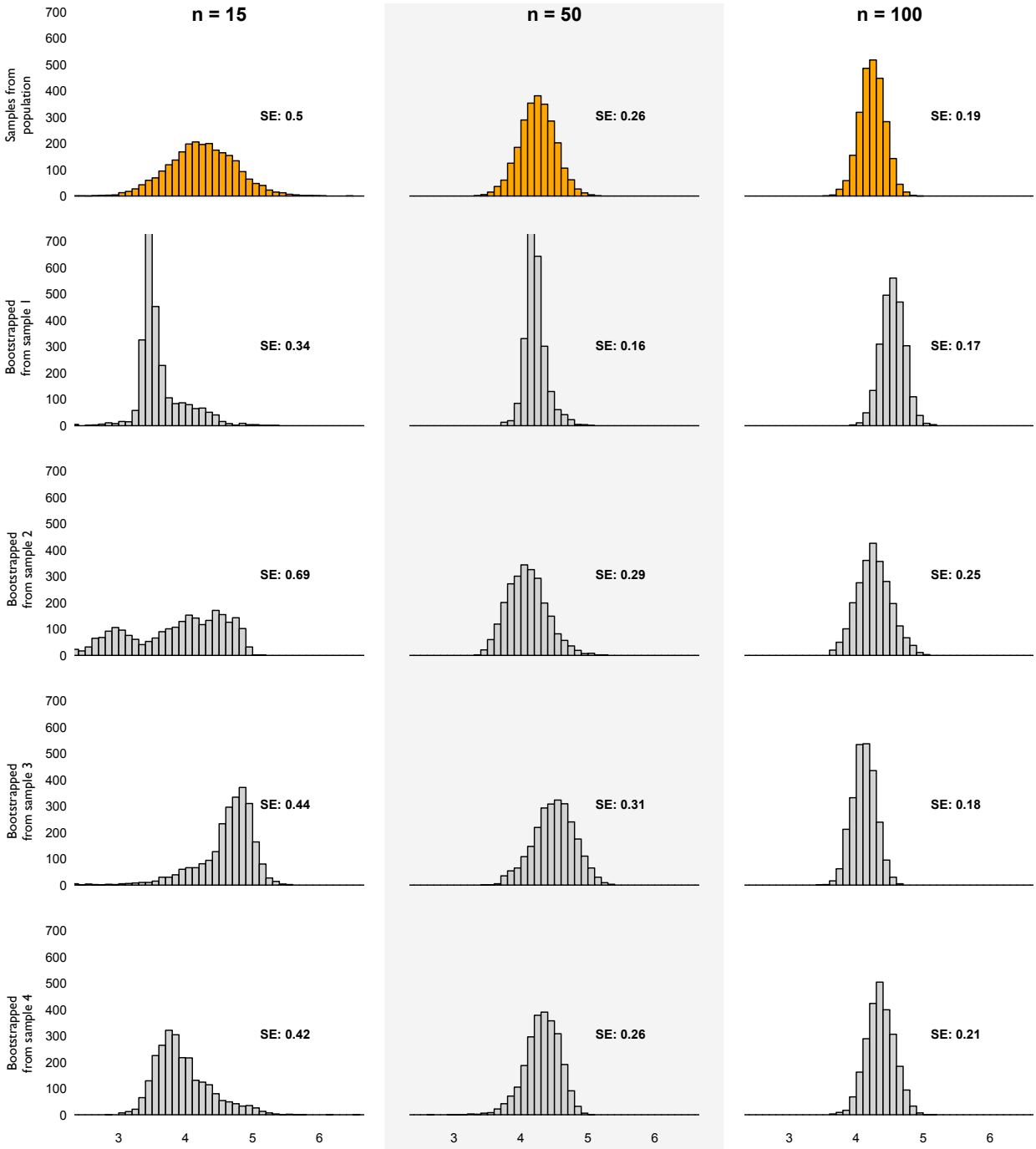


Figure 10.2: Actual (top, in orange) and bootstrapped sampling distributions (four replications) for the least-squares estimator of β_1 from Figure 8.1.

second sample of size 15; and so on for the rest of the grey panels.

If bootstrapping were perfect, each grey panel would look exactly like the corresponding orange panel above, regardless of the same size. But of course, bootstrapping isn't perfect. If you study these pictures closely, you'll notice a few things.

(1) The bootstrapped sampling distribution can differ substantially from one original sample to the next (top to bottom). The sample-to-sample differences are larger when the original sample size is small.
(2) The bootstrapped sampling distribution gets both closer to the truth, and less variable from one original sample to the next, as the original sample size gets larger.
(3) The bootstrapped standard errors (printed next to each histogram) are often closer to the true standard error than you might naïvely expect, based on the visual correspondence of the bootstrapped sampling distribution to the true one.

Confidence intervals and coverage

Now that we've learned to approximate an estimator's sampling distribution via bootstrapping, what do we do with this information? The answer is: we quantify the uncertainty of our estimate via a *confidence interval*: a range of plausible values for the true value of a parameter, together with an associated *confidence level* between 0% and 100%. The width of a confidence interval conveys the precision with which the data have allowed you to estimate the underlying population parameter. If your interval actually contains the true population value, we say that the interval *covers* the truth. If it doesn't, the interval *fails to cover* the truth. In real life, you won't know whether your interval covers. The confidence level expresses how confident you are that it actually does.

There are many ways of generating confidence intervals from bootstrapped sampling distributions, ranging from the simple to the highly sophisticated (and mathematically daunting). We'll focus on two simple ways here, with the understanding that the more technical ways we don't discuss are a bit more accurate.[3]

First, there's the basic standard-error method. Here, you quote a symmetric error bar centered on the estimate from the original sample, plus-or-minus some multiple k of the bootstrapped

[3] If you want to get an introduction to the more technical ways of getting confidence intervals from the bootstrap, see the following article: "Bootstrap confidence intervals: when, which, what? A practical guide for medical statisticians." James Carpenter and John Bithell. *Statistics in Medicine* 2000; 19:1141–64.

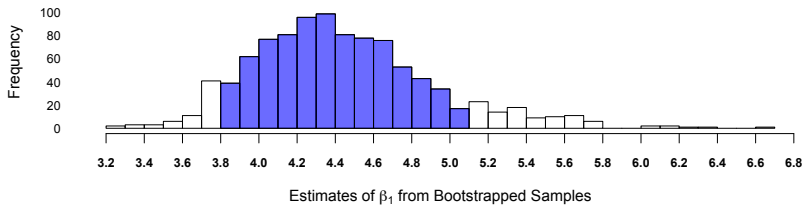


Figure 10.3: The estimated sampling distribution of $\hat{\beta}_1$ that arises from bootstrapping one sample of size 30 from the full fish population. The blue area reflects an 80% confidence interval generated by the coverage method, with symmetric tail areas of 10% above and 10% below the blue area.

standard error. To be precise, let's say that θ is some population parameter you're trying to estimate; that $\hat{\theta}$ is the estimate of θ generated by your actual sample; and that you've run the bootstrapping procedure on your sample and found that the bootstrapped standard error is $\hat{\sigma}$. Your confidence interval would then be

$$\theta \in \hat{\theta} \pm t^\star \hat{\sigma}\,,$$

where $t^\star$ is a chosen multiple. This number $t^\star$ is called the *critical value*. It is the number of standard errors you must go out from the center to capture a certain percentage of the sampling distribution. Typical values are $t^\star = 1$ (for an approximate 68% confidence interval) and $t^\star = 2$ (for an approximate 95% confidence interval).

The answer to the question of *why* $t^\star = 1$ corresponds to 68% and $t^\star = 2$ to 95% is beyond the scope of this chapter. It has to do with the normal distribution and something called the central limit theorem. For now, it is fine if you accept this is an empirical rule of thumb that statisticians have found gives a good approximation in situations where your bootstrapped sampling distribution looks approximately bell-shaped. Some of the more sophisticated bootstrap techniques, mentioned in Footnote 3, are focused on improving the choice of $t^\star$ given by these simple guidelines.

Second, there's the coverage-interval method, in which you simply calculate a coverage interval using the quantiles of your bootstrapped sampling distribution. For example, Figure 10.3 shows the bootstrapped sampling distribution for the slope of the weight–volume relationship arising from a single sample of 30 fish from the same lake as before. If you wanted to compute an 80% confidence interval based on this data, you would calculate the 10th and 90th percentiles of this histogram, giving you an interval that contains 80% of the bootstrapped estimates of the slope. In Figure 10.3, this interval is $(3.8, 5.1)$, shown in blue. This example highlights that, unlike the intervals generated by the standard-error method, the intervals generated by the coverage method

need not be symmetric about the estimate $\hat{\theta}$ derived from your actual sample.

Is one of these two methods better? Not as a general rule. The coverage-interval approach is more common in practice, and it's a fine default option. It's what we'll use throughout the course, and this book.

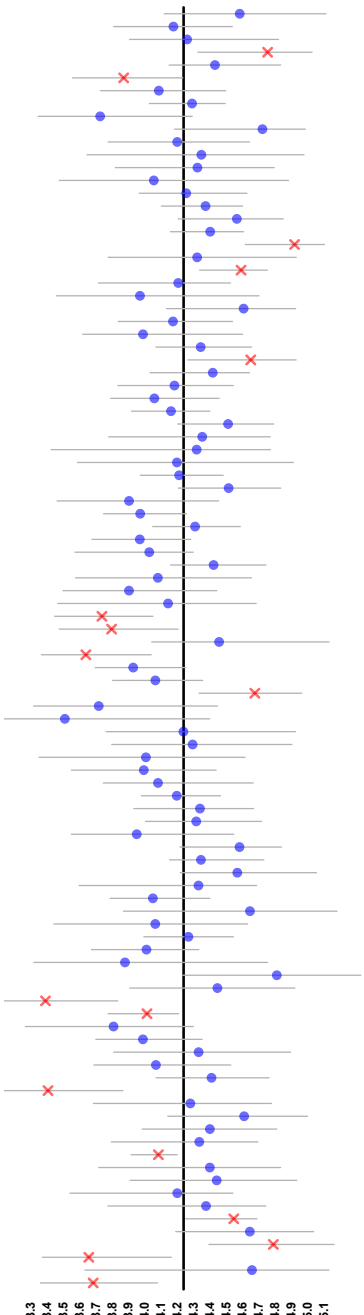
What does "confidence" mean?

The word "confidence," as it is used in the phrase "confidence interval," has a notoriously tricky interpretation. To put it concisely but opaqely, confidence intervals are intervals generated by a method that satisfies the frequentist coverage principle.

> *The frequentist coverage principle:* If you were to analyze one data set after another for the rest of your life, and you were to quote X% confidence intervals for every estimate you made, those intervals should cover their corresponding true values at least X% of the time. Here X can be any number between 0 and 100.

Let's unpack this a bit. Imagine that your interval was generated with a procedure that, under repeated use on one sample after the next, tends to yield intervals that cover the true value with a relative frequency of at least 80%. Then, and only then, may you claim a bona fide 80% confidence level for your specific interval. (You may, of course, aim for whatever coverage level you wish in lieu of 80%. Many people seem stuck on 95%, but it's entirely your choice.) Thus confidence intervals involve something of a bait-and-switch: they purport to answer a question about an individual interval, but instead give you information about some hypothetical assembly line that could be used to generate a whole batch of intervals. Nonetheless, there is an appealing "truth in advertising" property at play here: that if you're going to claim 80% confidence, you should be right 80% of the time over the long run.

An obvious question is: do bootstrapped confidence intervals satisfy the frequentist coverage property? If your sample is fairly representative of the population, then the answer is a qualified yes. That is, the bootstrapping procedure yields nominal X% intervals that cover the true value "approximately" X% of the time. Moreover, as the size of the original sample gets bigger, the quality of the approximation gets better. Alas, it is necessary to appeal to some very advanced probability theory to put both of these claims



on firm footing. (This is best deferred to another, much more advanced book. For those that like fancy math, the relevant branch of probability theory is called empirical-process theory, which part of a wider area called stochastic processes.)

For our purposes, it is better to show the procedure in action. Figure 10.4, for example, depicts the results of running 100,000 regressions—1,000 bootstrapped samples for each of 100 different real samples from the population in Figure 8.1. The vertical black line shows the true population value of the weight–volume slope ($\beta_1 = 4.24$) for our population of fish. Each row corresponds to a different actual sample of size $n = 30$ from the population. Dots and crosses indicate the least-squares estimate of the slope arising from that sample, while the grey bars show the corresponding 80% bootstrapped confidence intervals generated by the coverage method (just like the blue region in Figure 10.3).

The nominal confidence level of 80% for each individual interval must be construed as a claim about the *whole ensemble* of 100 intervals: 80% should cover, 20% shouldn't. In fact, 83 of these intervals cover and 17 don't, so the claim is approximately correct.