Sampling Variability and the Bootstrap

Data Science Team
Sample variability so far

- We have learned that data is often a sample from a larger population.
- Sometimes the larger population is defined by a concrete number of subjects, sometimes an abstraction.
- We have discovered that there are many models for these abstract populations. When they include an infinite number of continuous possible values we use densities to describe the frequencies of the values (we will get back to this later).
- We have seen that using summaries of the sample we can estimate the value of these summaries in the population.
- We noted that these estimates have some variability.
Stanford undergraduate population

We know that the proportion of “Black or African american” is 0.064.

If we do the thought experiment of obtaining 10000 samples of size 50, and calculating the proportion of “Black or African american” in each of them.

![Histogram of 10000 estimates](image-url)
How much we are off on average?

If we could average all the 10000 samples, we would be doing quite well

\[
\text{mean}(\text{SamplePropB})
\]

\[
\text{#} \ [1] \ 0.064272
\]

The “problem” is that we get to see only one sample. And samples have a sizeable variance

\[
\text{sqrt}(\text{var}(\text{SamplePropB}))
\]

\[
\text{#} \ [1] \ 0.03449448
\]
Interpreting the result from one sample

```r
ssize<-50
calculsampler<-sample(UGRace,ssize,replace=F)
sum(calculsampler==
"Black or African American, non-Hispanic")/ssize
```

```r
## [1] 0.04
```
How to get a sense of the potential error from one sample?

- In reality, we would only have one sample of size 50.
- If we could repeat the random process many times, we would be fine.
- To repeat the random process, we need a population to sample from.
- As a “best guess” for the population we can use the sample we actually observed.
- Repeated sampling with replacement for the population defined by the sample is an idea.
Pulling oneself up from one’s bootstraps

```r
ssize<-50
actualsample<-sample(UGRace,ssize,replace=F)
B<-10000
bsampleprop<-NULL
for(i in 1:B)
{
  bootsample<-sample(actualsample,ssize,replace=T)
  bsampleprop<-c(bsampleprop,
                  sum(bootsample=="Black or African American, non-Hispanic")/ssize )
}
```
Pulling oneself up from one’s bootstraps

10000 samples from population

10000 re-samples from sample

Proportion of Black in sample

Proportion of Black in re-sample

Frequency

Frequency
Bootstrap
<table>
<thead>
<tr>
<th>Population</th>
<th>Sample</th>
<th>Bootstrap samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>We do not observe</td>
<td>We obtain one sample from the population</td>
<td>We can resample with replacement from the sample as many times as we want</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Population Summary</th>
<th>Sample Summary</th>
<th>Bootstrap Summaries</th>
</tr>
</thead>
<tbody>
<tr>
<td>This is the quantity we would like to know, but we cannot observe</td>
<td>This is our estimate for the population summary</td>
<td>By calculating the standard deviation of the summaries of the different bootstrap samples we learn about the error of our estimate</td>
</tr>
</tbody>
</table>
A toy example

Here is a small population with its true mean

```r
population <- c(3, 3, 3, 5, 5, 5, 7, 7, 7, 9, 9, 9, 10, 10, 10)
mean(population)
```

## [1] 6.8

Here is our sample with its sample mean

```r
actualsample <- sample(population, 5, replace = FALSE)
mean(actualsample)
```

## [1] 5.8
A toy example, bootstrap samples

Now we construct multiple bootstrap samples, sampling with replacement from

```r
# [1] 9 5 9 3 3
```

<table>
<thead>
<tr>
<th>Boot Sample</th>
<th>B.obs1</th>
<th>B.obs2</th>
<th>B.obs3</th>
<th>B.obs4</th>
<th>B.obs5</th>
<th>B.Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>9</td>
<td>5.8</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>3</td>
<td>5</td>
<td>7.0</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>5</td>
<td>8.2</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>9</td>
<td>7.0</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>9</td>
<td>4.6</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>9</td>
<td>5</td>
<td>9</td>
<td>5</td>
<td>6.2</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>3.4</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>5</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>7.0</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>9</td>
<td>3</td>
<td>9</td>
<td>9</td>
<td>6.6</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>3</td>
<td>4.6</td>
</tr>
</tbody>
</table>
A toy example, evaluating variability

We can calculate the standard deviation of the bootstrap sample to estimate the standard error of the sample estimator

```
sqrt(var(bsamplemean))
```

```
## [1] 1.20398
```

Or we can calculate a range of possible estimated value using quantiles

```
quantile(bsamplemean, c(.05, .95))
```

```
## 5% 95%
## 3.8 7.8
```
A toy example, checking that the bootstrap helped

Histograms of variability of sample estimate

![Histograms](image)

- **Obtained with Bootstrap**
  - Sample mean

![Histograms](image)

- **Thought experiment**
  - Population mean

![Histograms](image)
Bootstrap principle

- Introduced by Prof. Brad Efron in early 1980s.
- The bootstrap works here because we are able to replicate the chance process of drawing (with replacement) from an unknown population by drawing with replacement from our observed sample.
- Works in the sense that it gives a pretty good estimate of the sampling variability. Even when we do not know the exact population generating our sample.
- Also works in that we did not have to use any complicated calculations to get this estimate.
Does it always work?

- If our chance process were different, then drawing repeatedly from our observed sample will not help us.
- **We have to believe that the resampling process we use is a reasonable approximation to the way our data was generated.**
- We might call this **no free lunch principle**: we can use the computer to do calculations for us about the sampling processes **only** when we know how to tell the computer to mimic the sampling process.
Let’s try some other examples

- **Population (unknown):** $X \sim \mathcal{N}(0, 10)$
- **Summary of interest:** Median of $X$
- **Estimating strategy:** take a random sample of size 20 and calculate sample median

```r
observation <- rnorm(20, mean = 0, sd = 10)
median(observation)
```

## [1] 3.829993

We will compare what we learn about variability of our estimate with:

1. A thought experiment done assuming we know the population
2. The bootstrap strategy applied to the observed sample
Sampling variability and its reconstruction by the bootstrap

Thought experiment

Population median

Sample median

Frequency

−10 −5 0 5 10
0 500 1000 1500 2000 2500

Bootstrap

Sample median

Bootstrap samples medians

Frequency

−5 0 5 10
0 200 600 1000 1400

Sample median

Bootstrap samples medians