Bootstrapping and confidence intervals

02.06.2022, Data Science (SpSe 2022): T15

Prof. Dr. Claudius Gräbner-Radkowitsch Europa-University Flensburg, Department of Pluralist Economics www.claudius-graebner.com | @ClaudiusGraebner | claudius@claudius-graebner.com





Prologue:



Prologue Feedback and exercises

- XX of you filled out the feedback survey. Main take-aways:
 - TBA
- What were the main problems with the exercises?



Learning Goals

- Understand the concept of bootstrapping and for what it is useful
- Understand the concept of a confidence interval and learn how to compute and interpret them
- Understand how we can become more confident in our estimation of population parameters



Motivation



Why bootstrapping?

- Usually we cannot study the populations of interest directly → statistical inference via random samples
- To interpret estimates from random samples, we must know about the properties of our samples
 - Especially: their **sampling distribution** → determines confidence in estimates
- In the previous session we learned how to study the sampling distributions of point estimates using MCS
- But this assumed that we can draw many samples, or even know certain properties of the population
 - In practice we can draw only one sample and know little about the population → bootstrapping









Why bootstrapping

- The idea of bootstrapping is to study the sampling distribution of our point estimate by re-sampling our sample
- This means: we draw many sub-samples from our sample and study them via MCS
- It turns out that this does **not** help us to improve our estimates...



 ...but gives accurate information about the sampling distribution of our estimate → quantify uncertainty due to random sampling



Re-sampling somehow allows us to "pull ourselfes up by your bootstraps"



- Suppose we want to know the average study semester of EUF students
 - **Population parameter** of interest: μ (average study semester)
 - We could make a census and ask all students but this is to much work
- We might therefore ask a random sample of 100 students about their study semester and infer the population parameter of interest
 - Sample statistic: sample mean \bar{x} (or $\hat{\mu}$) of the study semester



- Suppose that the distribution of study semesters in our sample is as such:
- The average study semester of our sample is given as $\bar{x} = 4.84$
- If our sample was drawn randomly, this would be a good guess for the average study semester in the population



- However, if we asked another 50 students, we would most likely get a different sample statistic → sample variation
- Thus, it might be good and honest to quantify the uncertainty around our guess of 4.84 \rightarrow requires knowledge about sampling distribution of \bar{x}



- But how can we get information about the sampling distribution without repeating the process of drawing a sample many, many times?
- The answer: re-sampling, i.e. drawing a sample from our sample
 - Draw 100 elements from our sample but with replacement (since n = 100)
 - What we are doing is called **re-sampling with replacement**
- Here is the result of a single re-sampling activity:







- The re-sample has some similarities to the original sample, but is not identical → sampling variation due to re-sampling
- But does this re-sample help us understand the sampling distribution of our guess \bar{x} for the population parameter μ ?
- No, since it is only a single re-sample!
 - Lets repeat this process 100 times and compute \bar{x} for each re-sample!



- The figure on the right side is called the bootstrap distribution
 - Result of re-sampling the original sample many times and compute the sample statistic of interest for each iteration
 - Note that this distribution looks approximately like a normal distribution
- The bootstrap distribution is an **approximation to the sampling distribution** of our sample statistic \bar{x}



Central take-aways and implementation

- We were interested in the population parameter $\mu,$ i.e. the average study semester of EUF students
- We drew a single random sample and computed the sample mean \bar{x}
- In principle, this is not a bad guess for µ, but we were aware that the sample mean is subject to sample variation
- To get information about the sample distribution of \bar{x} we did **re-sampling** with replacement and produced a **bootstrap distribution**
 - This is the distribution of sample means for 1000 re-samples from our original sample
- The bootstrap distribution **approximates the sample distribution** of \bar{x}



Central take-aways and implementation

 To draw a sample or repeated samples you may use the convenience function moderndive::rep_sample_n():



- The code above draws 1000 samples of size 50 from the tibble pop_data; each sample is drawn with replacement
- The function always produces tibbles of the following form:

A tibble: 50×3 replicate [1] # Groups: replicate Semester ΤD <int> <fct> <int> 1 2 4237 1 2 14 4818 3 3937 1 8 4089 4 14



Exercise 1: Constructing a bootstrap distribution

- Consider the data set T15 SemesterSample.csv from the course page
 - Contains a sample of EUF students and their study semester
- Compute the bootstrap distribution for the sample mean as discussed above to answer the following two questions:



- 1. What is the effect of the number of iterations during the bootstrap resampling process?
 - Look at the resulting distributions for 10, 50, 100, 500, and 1000 replications! What do you observe?
- 2. How could you use the bootstrap distribution to quantify your uncertainty about the sample mean and its usefulness to estimate μ ?

Exercise 1: Constructing a bootstrap distribution



- For higher number of iterations, the values look more and more normally distributed
- Example solution is available online

Confidence intervals



Motivation for confidence intervals

- We used the bootstrap to get an approximation of the sampling distribution of our point estimate \bar{x}
 - But we would we want this information in the first place?
- Since point estimates are different for each sample drawn, it would be nice quantify our confidence in the particular estimate
 - Are we sure that the estimate is very close to the true parameter μ ?
 - Or might the point estimate be rather far away due to sampling variation?
- If the sampling variation is very high, a single point estimate is more likely to be misleading than if the sampling variation is low
- Thus, a better (or at least: more honest) alternative to a point estimate is a **confidence interval**: an interval for which are are pretty sure it contains μ



Motivation for confidence intervals

• As alway, Ismay & Kim (2022) have a nice analogy:

Point estimate



Confidence interval

- The bootstrap distribution makes it easy to construct intervals for which we can be confident they contain μ

Image source: Ismay & Kim (2022)



Constructing a confidence interval

- To construct a confidence interval, the following steps are necessary:
 - 1. Choose the desired level of confidence
 - 2. Do the bootstrapping
 - 3. Choose the method to compute the confidence interval
 - 4. Compute the confidence interval
 - 5. If desired, visualise the results
- We now learn how this can be done using the package infer
- We focus on the **percentile method** to compute confidence intervals
 - An alternative method is described in the mandatory readings



Percentile method: intuition

- We first need to specify the desired level of confidence
 - Confidence is measure in percent, typical values are 90%, 95% or 99%
 - The higher the confidence the larger the confidence interval
- Assume we want to construct a 95%-confidence interval
 - We just pick the middle 95% of the bootstrap distribution:





Percentile method: intuition

- Assume we want to construct a 95%-confidence interval
 - We just pick the middle 95% of the bootstrap distribution
 - To get the quantitative thresholds, compute the 2.5 and 97.5th percentile:



- 1. Specify the variable that is of main interest using infer::specify()
- 2. Generate basis for the bootstrap distribution using infer::generate()
- 3. Generate the actual bootstrap distribution using infer::calculate()
- 4. Process the bootstrap distribution further, e.g. to create visualisations or to compute the actual CI



- 1. Specify the variable that is of main interest using infer::specify()
 - In the present case: \bar{x} as estimate for the the population parameter μ :

```
data_used %>%
 infer::specify(formula = MeanSemester ~ NULL)
```

- The ~ NULL part is because we are only interested in the sampling distribution of $\bar{x} \rightarrow$ later we will adjust this notation to, e.g., the regression context
- 2. Generate basis for the bootstrap distribution using infer::generate()

```
data_used %>%
 infer::specify(formula = MeanSemester ~ NULL)
 infer::generate(reps = 1000, type = "bootstrap")
```

• reps controls the number iterations, type should always be "bootstrap"



3. Generate the actual bootstrap distribution using infer::calculate()

```
bootstrap_dist <- data_used %>%
 infer::specify(formula = MeanSemester ~ NULL)
 infer::generate(reps = 1000, type = "bootstrap") %>%
 infer::calculate(stat = "mean")
```

- In our case the statistic of interest is the mean, so we set stat to "mean"
- The code above produces the bootstrap distribution that forms the basis for all further analysis steps
 - For instance, to compute the confidence intervals we use infer::get_ci():

```
conf_ints <- bootstrap_dist %>%
 infer::get_ci(
     level = 0.95,
     type = "percentile")
```

 We can also take a more visual approach to create a plot with the confidence intervals directly using infer::visualize():

```
visualize(student_boot_dist) +
 infer::shade_confidence_interval(
     endpoints = conf_ints)
```





Interpreting confidence intervals

- The interpretation of confidence intervals is not straightforward 🤓
- Its main purpose is of the CI is to provide a corridor for which we are confident that it contains the population parameter of interest
- The problem: we do not know the true value for μ so we will never know whether our CI actually contains μ or not \Im
- But what we can do is to consider an artificial situation in which we know μ and this way study the effectiveness of constructing CI using the method





Exercise 2: how well do confidence intervals work?

- Consider the data set
 DataScienceExercises::EUFstudents
- Contains a census for all EUF students and their height \rightarrow we know that $\mu = 166.5808$
 - This allows us to test whether our method to construct CI actually works



- To do so, we will conduct a MCS. To prepare it, do the following once:
 - Draw a random sample from the population
 - Compute the 95% percent confidence interval
 - Check whether the confidence interval contains the true average height
- To test whether an interval the true value you may use ifelse()!

The role of confidence

- We have the code to test whether a CI contains a true value once, now we iterate this process 100 times to draw more general conclusions
 - It turns out that about 95% if the CI contain the true value
- This is where the 95% come from:
 - We expect 95% of the CIs so constructed contain the true value
 - But in reality we only draw one sample and we can only construct the CI once
 - Nevertheless, this gives us a quantitative measure for the confidence in our statement
- What it we computed an 80% confidence interval?
 - Right, we expect 80% of the CIs so constructed contain the true value



How to interpret confidence intervals

- But what is the correct way to interpret a confidence interval?
- Assume we have $CI_{X\%} = [a; b]$, then the correct interpretation is:

If we repeated our sampling procedure a large number of times, we expect about X% of the resulting confidence intervals to capture the value of the population parameter.

• An informal variant frequently use is:

We are X% "confident" that $CI_{X\%} = [a; b]$ captures the value of the population parameter.

• A wrong interpretation is:

There is an X% probability that $CI_{X\%} = [a; b]$ contains μ .



Final remarks

- Note that there are two main determinants for the size of a CI:
 - First, the larger the confidence level, the larger the confidence interval
 - Second, the larger our sample, the smaller the confidence interval
- This indicates how important sample size is in practice
- Larger samples allow for more confident point estimates
- If we have a fixed sample size, we can increase confidence only by making less informative guesses





Summary & outlook





- In reality we can only use a single random sample to make an inference about an unknown population parameter
- The sample must be random in order to allow for effective inference
- But since it is random we also have random variation
- To quantify our confidence in our estimate we would like to take into account the effect of this random variation → get information about sampling distribution of the estimate
- A good approximation for the (unknown) sampling distribution is the bootstrap distribution
- The bootstrap distribution is obtained by doing re-sampling with replacement on our sample



Summary

- Bootstrap distributions cannot improve the point estimates as such
 - Their sample statistics differ from the population parameter of interest
- The standard error of the bootstrap distribution is good approximation for standard error of the (unknown) sampling distribution
- There are two main determinants for the size of a confidence interval:
 - The higher the confidence, the larger the interval
 - The larger the sample size, the smaller the interval
- This illustrates how important the sample size is in practice



Outlook

- In practice, confidence intervals are less frequently used than p-values, which we encounter in the next session
- But CI represent a more intuitive and transparent measure for the uncertainty associated with point estimates
- Next session we will learn how to assess hypotheses in a quantitative way

Tasks until next week:

- 1. Fill in the quick feedback survey on Moodle
- 2. Read the tutorials and lecture notes posted on the course page
- 3. Do the **exercises** provided on the course page and **discuss problems** and difficulties via the Moodle forum

