hw08

May 15, 2022

Name: Allan Gongora

Section: 0131

1 Homework 8: Confidence Intervals

Please complete this notebook by filling in the cells provided. Before you begin, execute the following cell to load the provided tests.

[1]:

```
pip install gofer-grader
```

```
Collecting gofer-grader
  Using cached gofer_grader-1.1.0-py3-none-any.whl (9.9 kB)
Requirement already satisfied: tornado in /opt/conda/lib/python3.7/site-packages
(from gofer-grader) (6.1)
Requirement already satisfied: jinja2 in /opt/conda/lib/python3.7/site-packages
(from gofer-grader) (3.0.3)
Requirement already satisfied: pygments in /opt/conda/lib/python3.7/site-
packages (from gofer-grader) (2.11.2)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/conda/lib/python3.7/site-
packages (from jinja2->gofer-grader) (2.0.1)
Installing collected packages: gofer-grader
Successfully installed gofer-grader-1.1.0
Note: you may need to restart the kernel to use updated packages.
```

[2]:

```
# Don't change this cell; just run it.

import numpy as np
from datascience import *

# These lines do some fancy plotting magic.
import matplotlib
%matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
import warnings
warnings.simplefilter('ignore', FutureWarning)
```

```
# These lines load the tests.

from gofer.ok import check
```

Recommended Reading: * Estimation 1) For all problems that you must write explanations and sentences for, you **must** provide your answer in the designated space. This can include: A) Sentence reponses to questions that ask for an explanation B) Numeric responses to multiple choice questions C) Programming code

2) Moreover, throughout this homework and all future ones, please be sure to not re-assign variables throughout the notebook! For example, if you use `max_temperature` in your answer to one question, do not reassign it later on. Otherwise, you will fail tests that you thought you were passing previously!

Once you're finished, select "Save and Checkpoint" in the File menu. Your name and course section number should be in the first and last cell of the assignment. Be sure you have run all cells with code and that the output from that is showing. Then click "Print Preview" in the File menu. Print a copy from there in pdf format. (This means you right click and choose print and choose "save as pdf" from your printer options.) You will need to submit the pdf in Canvas by the deadline.

The gopher grader output and/or output from your coding are essential to helping your instructor grade your work correctly and in a timely manner.

Files submitted that are missing the required output will lose some to all points so double check your pdf before submitting.

1.1 1. Plot the Vote

Four candidates are running for the President of Dataland. A polling company surveys 1000 people selected uniformly at random from among voters in Dataland, and it asks each one who they are planning to vote for. After compiling the results, the polling company releases the following proportions from its sample:

Candidate	Proportion
Candidate C	0.47
Candidate T	0.38
Candidate J	0.08
Candidate S	0.03
Undecided	0.04

These proportions represent a uniform random sample of the population of Dataland. We will attempt to estimate the corresponding *population parameters*, or the proportion of the votes that each candidate received from the entire population. We will use confidence intervals to compute a range of values that reflects the uncertainty of our estimate.

The table `votes` contains the results of the survey. Candidates are represented by their initials. Undecided voters are denoted by `U`.

[3]:

```
votes = Table().with_column('vote', np.array(['C']*470 + ['T']*380 + ['J']*80 +␣
 ↪['S']*30 + ['U']*40))
num_votes = votes.num_rows
votes.sample()
```

[3]:

```
vote
J
C
C
T
C
T
C
U
T
C
… (990 rows omitted)
```

Question 1.1 Below, complete the given code that will use bootstrapped samples from `votes` to compute estimates of the true proportion of voters who are planning on voting for **Candidate C**. Make sure that you understand what's going on here. It may be helpful to explain `proportions_in_resamples` to a friend or TA.

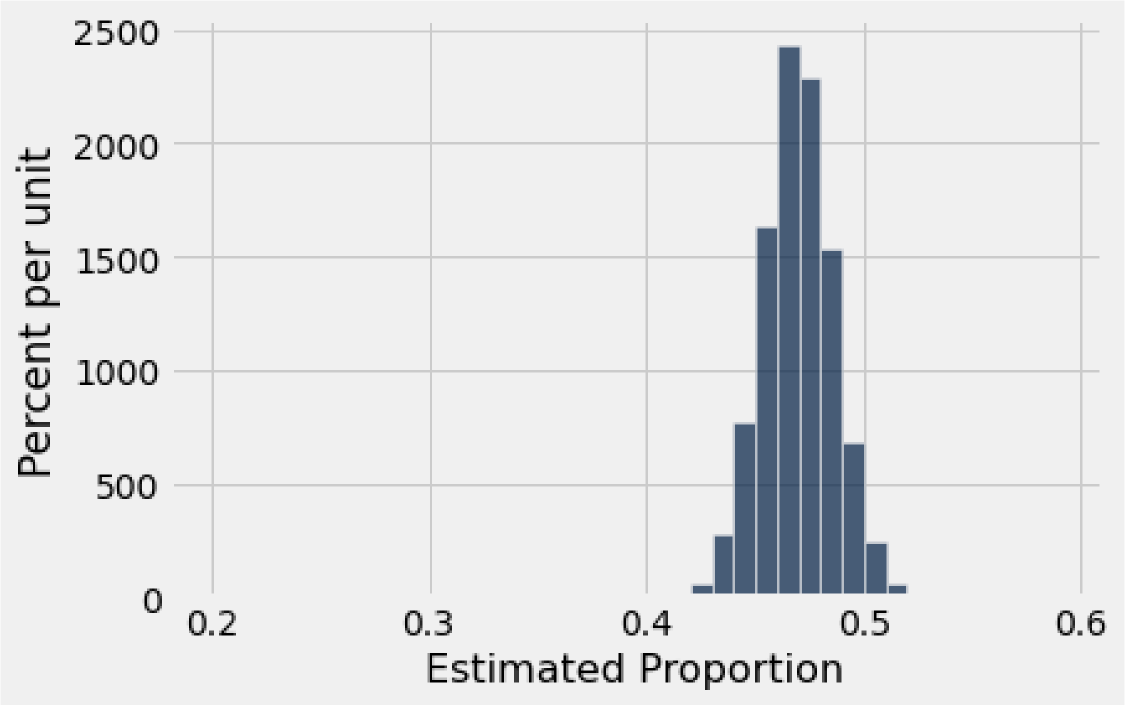
[8]:

```
def proportions_in_resamples():
    prop_c = make_array()
    for i in np.arange(5000):
        bootstrap = votes.sample()
        single_proportion = bootstrap.where("vote", "C").num_rows / bootstrap.
 ↪num_rows
        prop_c = np.append(prop_c, single_proportion)
    return prop_c
```

In the following cell, we run the function you just defined, `proportions_in_resamples`, and create a histogram of the calculated statistic for the 5,000 bootstraps. Based on what the original polling proportions were, does the graph seem reasonable? Talk to a friend or ask a TA if you are unsure!

[9]:

```
sampled_proportions = proportions_in_resamples()
Table().with_column('Estimated Proportion', sampled_proportions).hist(bins=np.
 ↪arange(0.2,0.6,0.01))
```



Question 1.2 Using the array `sampled_proportions`, find the values that bound the middle 95% of the values in the data. (Compute the lower and upper ends of the interval, named `c_lower_bound` and `c_upper_bound`, respectively.)

[13]:
```
c_lower_bound = percentile(2.5, sampled_proportions)
c_upper_bound = percentile(97.5, sampled_proportions)
print("Bootstrapped 95% confidence interval for the proportion of C voters in␣
 ↪the population: [{:f}, {:f}]".format(c_lower_bound, c_upper_bound))
```

```
Bootstrapped 95% confidence interval for the proportion of C voters in the
population: [0.439000, 0.502000]
```

[14]:
```
check('tests/q1_2.py')
```

[14]:
```
<gofer.ok.OKTestsResult at 0x7fe7a8b97e10>
```

Question 1.3 The survey results seem to indicate that Candidate C is beating Candidate T among voters. We would like to use confidence intervals to determine a range of likely values for her true *lead*. Candidate C's lead over Candidate T is:

$$\text{Candidate C's proportion of the vote} - \text{Candidate T's proportion of the vote.}$$

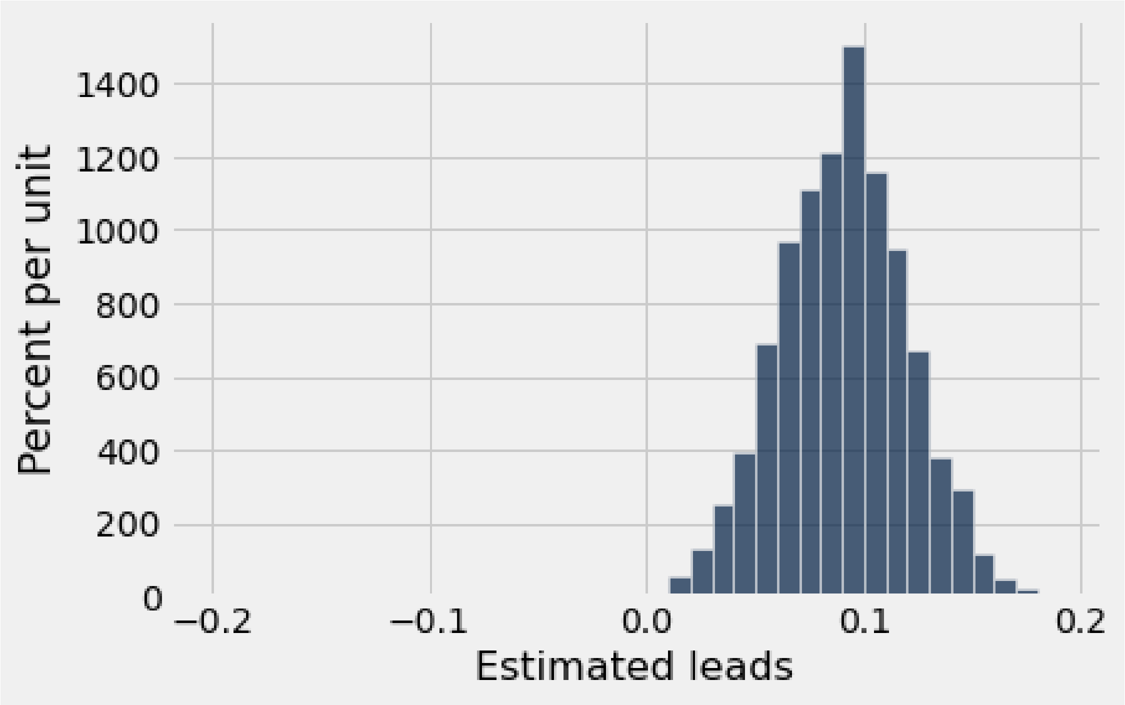
Using the function `proportions_in_resamples` above as a model, use the bootstrap to compute an approximate distribution for Candidate C's lead over Candidate T. Plot a histogram of the resulting samples.

[16]:

```
bins = np.arange(-0.2,0.2,0.01)

def leads_in_resamples():
    leads = []
    for i in range(5000):
        bootstrap = votes.sample()
        prop_c = bootstrap.where("vote", "C").num_rows / bootstrap.num_rows
        prop_t = bootstrap.where("vote", "T").num_rows / bootstrap.num_rows
        leads.append(prop_c - prop_t)
    return leads

sampled_leads = leads_in_resamples()
Table().with_column("Estimated leads", sampled_leads).hist(bins=bins)
```



[17]:

```
diff_lower_bound = percentile(2.5, sampled_leads)
diff_upper_bound = percentile(97.5, sampled_leads)
print("Bootstrapped 95% confidence interval for Candidate C's true lead over␣
 ↪Candidate T: [{:f}, {:f}]".format(diff_lower_bound, diff_upper_bound))
```

```
Bootstrapped 95% confidence interval for Candidate C's true lead over Candidate
T: [0.032000, 0.148000]
```

[18]:

```
check('tests/q1_4.py')
```

[18]:

```
<gofer.ok.OKTestsResult at 0x7fe7a6a8e1d0>
```

1.2 2. Interpreting Confidence Intervals

The staff computed the following 95% confidence interval for the proportion of Candidate C voters:

$$[.439, .5]$$

(Your answer may have been different; that doesn't mean it was wrong!)

Question 2.1 Can we say that 95% of the population lies in the range $[.439, .5]$? Explain your answer.

Not sure

Question 2.2 Can we say that there is a 95% probability that the interval [.439, .5] contains the true proportion of the population who is voting for Candidate C? Explain your answer.

Not sure

A note about this question (this is outside of the scope of this class. If you don't already know what Bayesian and Frequentist reasoning are, don't worry about it!): You may recall that there are different philosophical interpretations of probability. The Bayesian interpretation says that it is meaningful to talk about the probability that the interval covers the true proportion, but a Bayesian would perform a different calculation to calculate that number; we have no guarantee that it is 95%. All we are guaranteed is the statement in the answer to the next question.

Question 2.3 Suppose we produced 10,000 new samples (each one a uniform random sample of 1,000 voters) and created a 95% confidence interval from each one. Roughly how many of those 10,000 intervals do you expect will actually contain the true proportion of the population?

Assign your answer to `true_proportion_intervals`.

```
[24]: true_proportion_intervals = 10000 * .95
```

```
[25]: check('tests/q2_3.py')
```

```
[25]: <gofer.ok.OKTestsResult at 0x7fe7a69f1090>
```

Question 2.4 The staff also created 80%, 90%, and 99% confidence intervals from one sample, but forgot to label which confidence interval represented which percentages! Match the interval to the percent of confidence the interval represents. (Write the percentage after each interval below.) **Then**, explain your thought process.

Answers:

$[.444, .495]$:

$[.450, .490]$:

$[.430, .511]$:

80 -> [.450, .490]

smallest range = smallest CI because it only has to cover 80% of the data

90 -> [.444, .495]

process of elimination

99 -> [.430, .511]

largest range = largest CI interval because it has to cover more (99) of the data

Recall the second bootstrap confidence interval you created, estimating Candidate C's lead over Candidate T. Among voters in the sample, her lead was .09. The staff's 95% confidence interval for her true lead (in the population of all voters) was:

$$[.032, .15].$$

Suppose we are interested in testing a simple yes-or-no question:

> "Are the candidates tied?"

Our null hypothesis is that the proportions are equal, or, equivalently, that Candidate C's lead is exactly 0. Our alternative hypothesis is that her lead is not equal to 0. In the questions below, don't compute any confidence intervals yourself - use only the staff's 95% confidence interval.

Question 2.5 Say we use a 5% P-value cutoff. Do we reject the null, fail to reject the null, or are we unable to tell using our staff confidence interval?

Assign `candidates_tied` to the number corresponding to the correct answer.

1. Reject the null
2. Fail to reject the null
3. Unable to tell using our staff confidence interval

Hint: If you're confused, take a look at this chapter of the textbook.

```
[32]: candidates_tied = 1
```

```
[33]: check('tests/q2_5.py')
```

```
[33]: <gofer.ok.OKTestsResult at 0x7fe7a68b3dd0>
```

Question 2.6 What if, instead, we use a P-value cutoff of 1%? Do we reject the null, fail to reject the null, or are we unable to tell using our staff confidence interval?

Assign `cutoff_one_percent` to the number corresponding to the correct answer.

1. Reject the null
2. Fail to reject the null
3. Unable to tell using our staff confidence interval

```
[30]: cutoff_one_percent = 2
```

```
[31]: check('tests/q2_6.py')
```

```
[31]: <gofer.ok.OKTestsResult at 0x7fe7a68859d0>
```

Question 2.7 What if we use a P-value cutoff of 10%? Do we reject, fail to reject, or are we unable to tell using our confidence interval?

Assign `cutoff_ten_percent` to the number corresponding to the correct answer.

1. Reject the null
2. Fail to reject the null
3. Unable to tell using our staff confidence interval

```
[36]: cutoff_ten_percent = 3
```

```
[37]: check('tests/q2_7.py')
```

```
[37]: <gofer.ok.OKTestsResult at 0x7fe7a6891710>
```

2 Dunno what i was doing, was lost

2.1 3. Submission

Once you're finished, select "Save and Checkpoint" in the File menu. Your name and course section number should be in the first and last cell of the assignment. Be sure you have run all cells with code and that the output from that is showing.

Double check that you have completed all of the free response questions as the auto-grader does NOT check that and YOU are responsible for knowing those questions are there and completing them as part of the grade for this homework. When ready, click "Print Preview" in the File menu. Print a copy from there in pdf format. (This means you right click and choose print and choose "save as pdf" from your printer options.) You will need to submit the pdf in Canvas by the deadline.

The gopher grader output and/or output from your coding are essential to helping your instructor grade your work correctly and in a timely manner.

Files submitted that are missing the required output will lose some to all points so double check your pdf before submitting.

```
[ ]: # For your convenience, you can run this cell to run all the tests at once!
     import glob
     from gofer.ok import grade_notebook
     if not globals().get('__GOFER_GRADER__', False):
         display(grade_notebook('hw08.ipynb', sorted(glob.glob('tests/q*.py'))))
```

```
Bootstrapped 95% confidence interval for the proportion of C voters in the
population: [0.439000, 0.500000]
Bootstrapped 95% confidence interval for Candidate C's true lead over Candidate
T: [0.034000, 0.146000]
```

Name: Allan Gongora

Section: 0131