Confidence intervals

Put your name here

Put the date here

Introduction

Sampling variability means that we can never trust a single sample to identify a population parameter exactly. Instead of simply trusting a point estimate, we can look at the entire sampling distribution to create an interval of plausible values called a confidence interval. Our understanding of standard errors will help us determine how wide we need to make our confidence intervals in order to have some chance of capturing the true population value. Like hypothesis tests, confidence intervals are a form of inference because they use a sample to deduce something about the population.

Instructions

Presumably, you have already created a new project and downloaded this file into it. From the `Run` menu above, select `Run All` to run all existing code chunks.

When prompted to complete an exercise or demonstrate skills, you will see the following lines in the document:

ANSWER

These lines demarcate the region of the R Markdown document in which you are to show your work.

Sometimes you will be asked to add your own R code. That will appear in this document as a code chunk with a request for you to add your own code, like so:

```
# Add code here
```

Be sure to remove the line `# Add code here` when you have added your own code. You should run each new code chunk you create by clicking on the dark green arrow in the upper-right corner of the code chunk.

Sometimes you will be asked to type up your thoughts. That will appear in the document with the words, "Please write up your answer here." Be sure to remove the line "Please write up your answer here" when you have written up your answer. In these areas of the assignment, please use contextually meaningful full sentences/paragraphs (unless otherwise indicated) and proper spelling, grammar, punctuation, etc. This is not R code, but rather a free response section where you talk about your analysis and conclusions. You may need to use inline R code in these sections.

When you are finished with the assignment, knit to PDF and proofread the PDF file **carefully**. Do not download the PDF file from the PDF viewer; rather, you should export the PDF file to your computer by selecting the check box next to the PDF file in the Files pane, clicking the `More` menu, and then clicking `Export`. Submit your assignment according to your professor's instructions.

Load Packages

We load the standard `mosaic` package. We'll also need the `openintro` package later in the assignment for the `hsb2` data set. Finally, we introduce the `broom` package that cleans up the output of a wide variety of R commands.

```
library(openintro)
library(broom)
library(mosaic)
```

As always, we set the seed so that our results are reproducible.

```
set.seed(11111)
```

Sample statistics as an estimate of population parameters

All our previous simulations of sampling distributions were based on the assumption that we knew p, the true population proportion. For example, we assumed that a candidate in an election actually had 64% support and then deduced various properties of the sampling distribution model.

In reality, however, we never know p. In fact, that's the whole point of statistics: we do not know the true population parameters, so we gather samples. We will measure $\hat{p}$, our sample proportion, and we hope that $\hat{p}$ is a good estimate of p.

Because of sampling variability, $\hat{p}$ will almost never be exactly the same as p, but now we have a tool for figuring out how close together they should be. The sampling distribution model tells us that when we sample a value of $\hat{p}$, it should be within a few standard errors of p.

Estimating the standard error

We do need to make one adjustment to our standard error. The formula we gave in a previous assignment was

$$\sqrt{\frac{p(1-p)}{n}}.$$

However, in a situation where we have collected data to try to estimate p, clearly we do not know p. So we plug in the estimate from our sample instead and call this the standard error, which we abbreviate as SE:

$$SE = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}.$$

Building a confidence interval

Recall that about 95% of a distribution lies within two standard deviations of the mean. Therefore, it should be the case that the true proportion p should lie within two standard errors of $\hat{p}$ most of the time.

This is the idea of a confidence interval. Take $\hat{p}$ from the sample and add/subtract 2 standard errors:

$$(\hat{p} - 2SE, \hat{p} + 2SE)$$

$$= \left(\hat{p} - 2\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + 2\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right).$$

The quantity $2SE$ is called the *margin of error*.

Exercise

The number 2—derived from the 68-95-99.7 rule—is, in fact, slightly incorrect. Using your knowledge of normal models and the `qdist` command, what is the number of standard errors that would enclose *exactly* 95% of the middle of the sampling distribution? (Hint: `p = 0.95` is **not** the correct argument in the `qdist` function. The middle 95% does not occur at the 95th percentile. If you're still confused, go on and read the next section and revisit this problem later.)

ANSWER

```
# Add code here to calculate the exact number of standard errors
# from the mean that enclose the middle 95% of a normal distribution.
```

Please write up your answer here.

The interpretation is that when you go collect many samples, the confidence intervals you produce using your estimates $\hat{p}$ will capture the true population proportion 95% of the time.

There is no particular reason that we need to compute a 95% confidence interval, although that is the generally agreed-upon standard. We could compute a 90% confidence interval or a 99% confidence interval, or any other type of interval. (Having said that, if you choose other intervals besides these three, people might wonder if you're up to something.)

The general formula, then, is

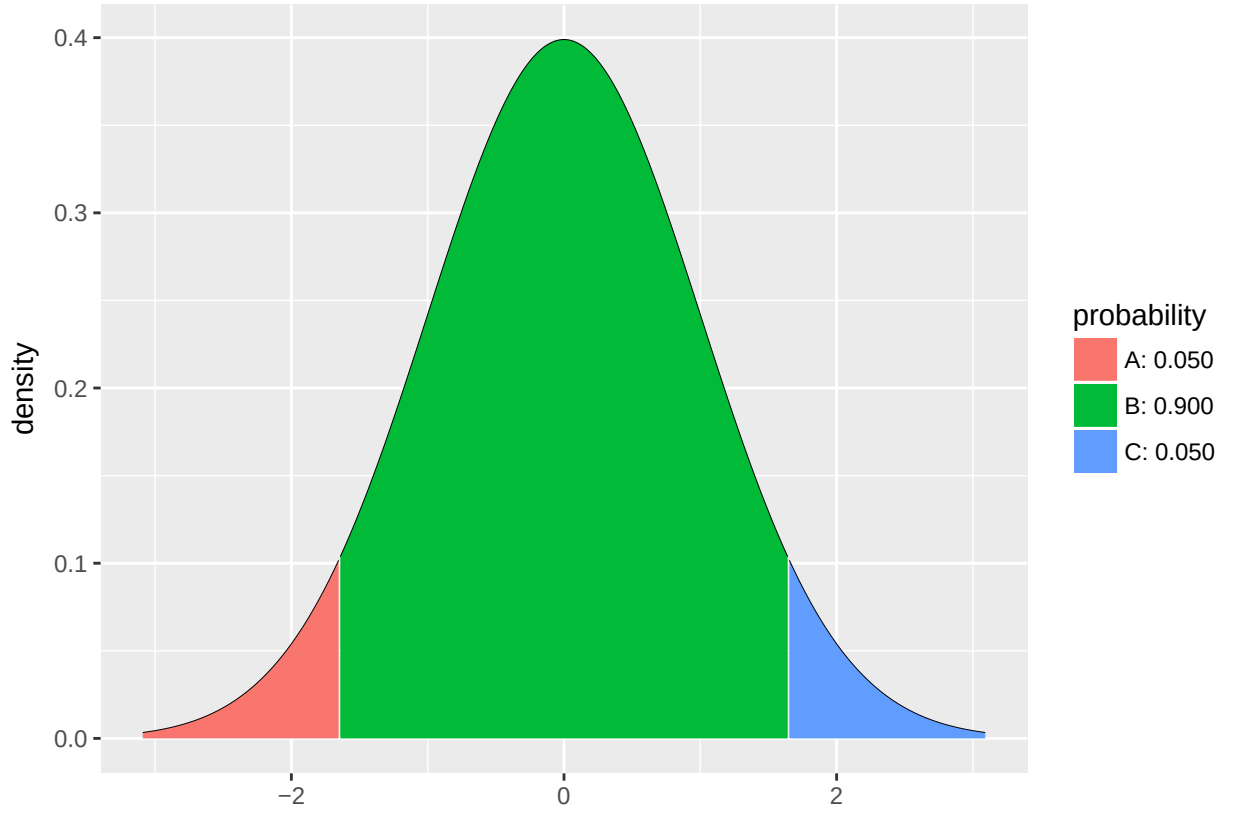
$$(\hat{p} - z^* SE, \hat{p} + z^* SE)$$

$$= \left(\hat{p} - z^* \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + z^* \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \right).$$

The new symbol z^* is called a *critical z-score*. This is the z-score that encloses the confidence level you want. The quantity $z^* SE$ is the new margin of error.

For example, suppose we wanted a 90% confidence interval. What is the corresponding critical z-score? Well, the middle 90% of a distribution leaves 5% in each tail. Therefore, we need to use `qdist` with the 5th and 95th percentiles.

```
qdist("norm", p = c(0.05, 0.95), invisible = TRUE)
```




```
qdist("norm", p = c(0.05, 0.95), plot = FALSE)
```

```
## [1] -1.644854  1.644854
```

The critical z-score is the positive answer, so we could use `qdist` simply to report 1.6448536. **Be careful!** The critical z-score for a 90% confidence interval requires `p = 0.95`. This is because the upper endpoint of a 90% interval is actually located at the 95th percentile.

(Also, if you had trouble with the question above that asked about the exact critical z-score for a 95% confidence interval, go back and try again now that you've seen another worked example.)

Exercise

Now do the same thing for a 99% confidence interval. In other words, calculate the critical z-score that encloses the middle 99% of the normal distribution.

ANSWER

```
# Add code here to calculate the exact number of standard errors
# from the mean that enclose the middle 99% of a normal distribution.
```

Please write up your answer here.

Checking conditions

Don't forget that there are always conditions to check. Before computing a confidence interval for a proportion, you must verify that the conditions are satisfied. These conditions are not really new; essentially, we are just checking the conditions that we already established for using a normal model as a sampling distribution model.

The "Random" and "10%" conditions will always be present. These are both necessary any time you want to infer from a sample.

The success/failure condition is also present, but here it looks a little different. Whereas before we computed np and $n(1-p)$, that was back when we were assuming we knew the true value of p. In reality, this is never the case; if we knew the true value of p already, we wouldn't be gathering a sample to try to infer it! So if we can't compute np and $n(1-p)$, what can we compute? Since the sample proportion $\hat{p}$ is our best guess for p, we will substitute its value everywhere there's a p.

So the new success/failure condition says that $n\hat{p}$ and $n(1-\hat{p})$ must be greater than 10. But, wait a minute, what are we really calculating with $n\hat{p}$ and $n(1-\hat{p})$? Since n is the sample size, and $\hat{p}$ is the sample proportion of successes, then $n\hat{p}$ is just the total number, or total raw count, of successes. And similarly, $n(1-\hat{p})$ is just the total number of failures.

Unlike np that describes the "theoretical" average expected number of successes—and therefore might not be a whole number—$n\hat{p}$ is the actual number of successes in our sample. This will always be a whole number! Same with the failures.

As an example, suppose a sample of size 67 has 52 successes. Then our sample proportion is

$$\hat{p} = \frac{52}{67} = 0.776$$

Keep in mind, though, that 0.776 is a rounded estimate. If we tried to compute $n\hat{p} = 67(0.776)$, we get 51.992. That's an absurd thing to do! The quantity $n\hat{p}$ is just the number of successes, and we already know it's 52. **Don't use a rounded (and therefore slightly incorrect) sample proportion to calculate the number of successes and failures. Just use the actual (whole) number of successes and failures.**

Okay, let's formally write down all the conditions that need to be checked:

- Random
  - The sample must be random (or hopefully representative).
- 10%
  - The sample size must be less than 10% of the size of the population.
- Success/failure
  - The number of successes and the number of failures **in our sample** must both exceed 10.

Using R to calculate confidence intervals

There are three forms in which you might have categorical data. Computing a confidence interval in R uses the command `prop.test` for all three forms, but in slightly different ways.

Method 1

You may just have a summary of the total number of successes and failures. For example, suppose we survey 326 people and 212 of them support a new initiative. (This means that 114 do not support it.) Assuming we have a random sample that is less than 10% of the population, and seeing as we have far more than 10 successes and failures, we get the confidence interval as follows:

```
prop.test(212, n = 326)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  212 out of 326
## X-squared = 28.862, df = 1, p-value = 0.00000007772
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.5954960 0.7015134
## sample estimates:
##         p
## 0.6503067
```

This does give us our confidence interval, but it also gives us a bunch of other stuff we don't need right now. (It's actually doing the "Mechanics" section for a full hypothesis test.) If we want just the confidence interval, the easiest thing to do is use the helpful `broom` package to take all this messy output and tidy it up first. As a matter of fact, the `broom` command that accomplishes this is called `tidy`. We'll assign the output of `prop.test` to a variable `test1` so we can access it. Then we'll apply the `tidy` command to `test1`.

```
test1 <- prop.test(212, n = 326)
test1_tidy <- tidy(test1)
test1_tidy
```

```
##    estimate statistic          p.value parameter conf.low conf.high
## 1 0.6503067  28.86196 0.00000007772435         1 0.595496 0.7015134
##                                                  method alternative
## 1 1-sample proportions test with continuity correction   two.sided
```

For now, ignore everything but the two numbers of interest: the low and high endpoints of our confidence interval given in the `conf.low` and the `conf.high` columns. (Depending on your screen size and resolution, you may have to click the right-facing black arrow in the output above to scroll over to where the `conf.low` and `conf.high` are listed.)

Here is how we report this inline:

> We are 95% confident that the true percentage of those who support the new initiative is captured in the interval (59.5495961%, 70.1513411%).

Notice that when communicating proportions to human beings, it's polite to convert them back to percentages.

Method 2

If you are given the percentages of successes and/or failures in your data, you'll have to convert them to whole number totals. For example, if we're told that 326 people were surveyed and 65% of them support the new initiative, then we have to do this:

```
test2 <- prop.test(round(326*0.65), n = 326)
test2_tidy <- tidy(test2)
test2_tidy
```

```
##    estimate statistic          p.value parameter conf.low conf.high
## 1 0.6503067  28.86196 0.00000007772435         1 0.595496 0.7015134
##                                                  method alternative
## 1 1-sample proportions test with continuity correction   two.sided
```

We have to round the number inside this command since 326 times 0.65 is not a whole number. It is the 65% that is, in fact, rounded, but we have no way of knowing what the exact number of successes was if all we're told is the proportion of successes. (It is possible that the true number of successes was actually 212, but 211 and 213 also round to 65%.)

The confidence interval is identical to the one from Method 1.

Method 3

Finally, it is possible that we have categorical data in a data frame. For example, what percentage of U.S. high school seniors go to private school? We use the `schtyp` variable in the `hsb2` data set.

First, check the conditions. The sample is presumably a representative sample of high school seniors from the U.S. as the survey was conducted by the National Center of Education Statistics. The sample size is 200, which is much less than 10% of the population of all U.S. high school seniors. Finally, we look at the number of successes and failures:

```
tally(~ schtyp, data = hsb2, margins = TRUE)
```

```
## schtyp
##  public private   Total
##     168      32     200
```

There are more than 10 successes and more than 10 failures (where a success is defined here to mean a senior who goes to private school).

Now we are ready to compute. If we do the following, though, we get the wrong answer:

```
test3a <- prop.test(~ schtyp, data = hsb2)
test3a_tidy <- tidy(test3a)
test3a_tidy
```

```
##   estimate statistic                        p.value parameter  conf.low
## 1     0.84    91.125 0.000000000000000000001348768         1 0.7800987
##   conf.high                                                 method
## 1 0.8864669 1-sample proportions test with continuity correction
##   alternative
## 1   two.sided
```

Exercise

Examine the output above and see if you can spot the problem. (Hint: does 32 students out of 200 fit with the confidence interval produced above?)

ANSWER

Please write up your answer here.

Try this instead:

```
test3b <- prop.test(~ schtyp, data = hsb2, success = "private")
test3b_tidy <- tidy(test3b)
test3b_tidy
```

```
##   estimate statistic                         p.value parameter  conf.low
## 1     0.16    91.125 0.000000000000000000001348768         1 0.1135331
##   conf.high                                                 method
## 1 0.2199013 1-sample proportions test with continuity correction
##   alternative
## 1   two.sided
```

We are 95% confident that the true percentage of U.S. high school seniors who attend private school is captured in the interval (11.3533121%, 21.9901306%).

Do you see what we changed in `test3b` that made it work properly?

Changing the confidence level

The default confidence level for a confidence interval is almost always 95%. It is possible, however, to use a different level.

```
test3c <- prop.test(~ schtyp, data = hsb2,
                    success = "private",
                    conf.level = 0.9)
test3c_tidy <- tidy(test3c)
test3c_tidy
```

```
##   estimate statistic                         p.value parameter  conf.low
## 1     0.16    91.125 0.000000000000000000001348768         1 0.1197396
##   conf.high                                                 method
## 1 0.2098621 1-sample proportions test with continuity correction
##   alternative
## 1   two.sided
```

Exercise

Is a 90% confidence interval wider or narrower than a 95% confidence interval? Explain why this is so. (In other words, from your understanding of how confidence intervals work, explain why it makes sense that a 90% confidence interval is wider or narrower than a 95% confidence interval.)

ANSWER

Please write up your answer here.

Rubric for confidence intervals

Calculating a confidence interval is a form of statistical inference. Typically, you will be asked to report a confidence interval after performing a hypothesis test. Whereas a hypothesis test gives you a "decision criterion" (using data to make a decision to reject the null or fail to reject the null), a confidence interval gives you an estimate of the "effect size" (a range of plausible values for the population parameter).

As such, there is a section in the "Rubric for Inference" that shows the steps of calculating and reporting a confidence interval. They are as follows:

1. Check the relevant conditions to ensure that model assumptions are met.
2. Calculate the confidence interval.
3. State (but do not overstate) a contextually meaningful interpretation.

Here is a worked example. Unless otherwise stated, we always use a 95% confidence level.

Some of the students in the "High School and Beyond" survey attended vocational programs. What percentage of all high school seniors attend vocational programs?

Check the relevant conditions to ensure that model assumptions are met.

- Random
  - The sample is presumably a representative sample of high school seniors from the U.S. as the survey was conducted by the National Center of Education Statistics.
- 10%
  - The sample size is 200, which is much less than 10% of the population of all U.S. high school seniors.
- Success/failure

```
tally(~ prog, data = hsb2, margins = TRUE)
```

```
## prog
##    general   academic vocational      Total
##         45        105         50        200
```

The number of "successes" (students in vocational programs) is 50, which is more than 10, and the number of "failures" (all other programs) is 150, also more than 10. (In this case, since there are three categories, we have to add up the `general` and `academic` totals to get the number of "failures".)

Calculate the confidence interval.

```
program <- prop.test(~ prog, data = hsb2, success = "vocational")
program_tidy <- tidy(program)
program_tidy
```

```
##   estimate statistic              p.value parameter  conf.low conf.high
## 1     0.25    49.005 0.000000000002553109         1 0.1928239 0.3169864
##                                                 method alternative
## 1 1-sample proportions test with continuity correction   two.sided
```

State (but do not overstate) a contextually meaningful interpretation.

We are 95% confident that the true percentage of U.S. high school seniors who attend a vocational program is captured in the interval (19.2823884%, 31.6986392%).

Your turn

Use the `smoking` data set from the `openintro` package. What percentage of the population of the U.K. smokes tobacco? (The information you need is in the `smoke` variable.)

Check the relevant conditions to ensure that model assumptions are met.

ANSWER

- Random
  - [Check condition here.]
- 10%
  - [Check condition here.]
- Success/failure
  - [Check condition here.]

Calculate the confidence interval.

ANSWER

```
# Add code here to calculate the confidence interval.
```

State (but do not overstate) a contextually meaningful interpretation.

ANSWER

Please write up your answer here.

Interpreting confidence intervals

Confidence intervals are notoriously difficult to interpret.[1]

Here are several *wrong* interpretations of a 95% confidence interval:

- 95% of the data lies in the interval.
- There is a 95% chance that the sample proportion lies in the interval.
- There is a 95% chance that the population parameter lies in the interval.

We'll take a closer look at these incorrect claims in a moment. First, let's see how confidence intervals work using simulation.

In order to simulate, we'll have to pretend temporarily that we know a true population parameter. Let's use the example of a candidate who has the support of 64% of voters. In other words, $p = 0.64$. We go out and get a sample of voters, let's say 50. From that sample we get a sample proportion $\hat{p}$ and construct a 95% confidence interval by adding and subtracting 1.96 standard errors from the value of $\hat{p}$. Most of the time,

[1] Several studies have given surveys to statistics students, teachers, and researchers, and find that even these people often misinterpret confidence intervals. See, for example, this paper: http://www.ejwagenmakers.com/inpress/HoekstraEtAlPBR.pdf

64% (the true value!) should be in our interval. But sometimes it won't be. We can get a "freak" sample that is far away from 64%, just by pure chance alone. (Perhaps we accidentally run into a bunch of people who oppose our candidate.)

Okay, let's do it again. We get a new sample, a new $\hat{p}$, and, therefore, a new confidence interval. We can do this over and over again through the magic of simulation!

Here's what this simulation looks like in R. One minor technical point: we will use the `mosaic` command `nflip` instead of `rflip` because its output is just a number (instead of all the extra stuff about heads and tails and all that). The following line of code looks complicated, but break it down from inside out. First, we use `nflip(50, prob = 0.64)` to flip a "weighted coin" 50 times as a proxy for surveying 50 voters about the candidate. The result of this (the number of successes) is then fed into the `prop.test` function that creates a confidence interval for a sample size of `n = 50`. Finally, `tidy` cleans up the output the same way we've seen all along in this assignment.

```
confint1 <- prop.test(nflip(50, prob = 0.64), n = 50)
confint1_tidy <- tidy(confint1)
confint1_tidy
```

```
##   estimate statistic   p.value parameter  conf.low conf.high
## 1      0.6      1.62 0.2030918         1 0.4520484 0.7326707
##                                                   method alternative
## 1 1-sample proportions test with continuity correction   two.sided
```

All we need to do for simulation purposes is to introduce the `do` command to do this a bunch of times. (We'll consolidate `tidy` and `prop.test` into the same line so that `do` can work its magic.)

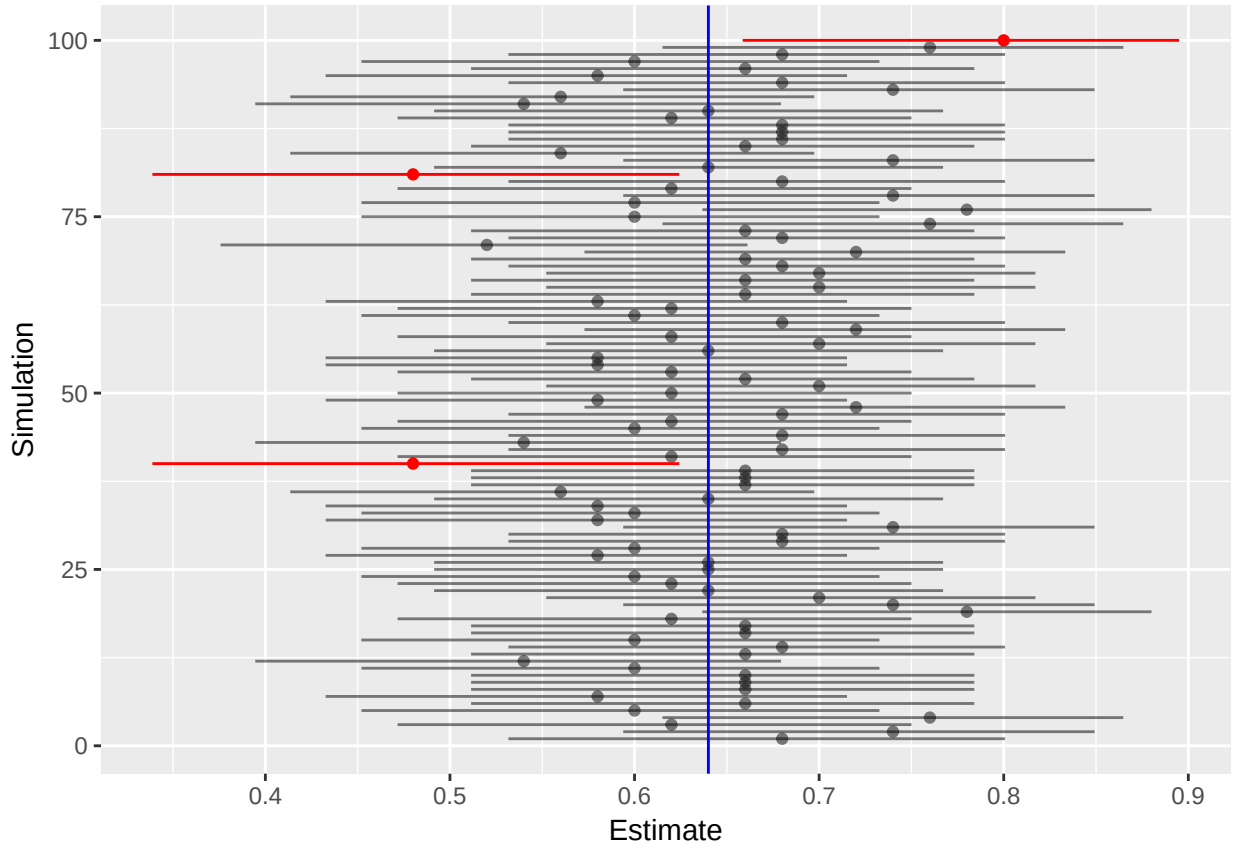
```
confint100 <- do(100) * tidy(prop.test(nflip(50, prob = 0.64), n = 50))
tail(confint100)
```

```
##     estimate statistic       p.value parameter  conf.low conf.high
## 95      0.58      0.98 0.32219880616         1 0.4326718 0.7151090
## 96      0.66      4.50 0.03389485352         1 0.5114459 0.7840536
## 97      0.60      1.62 0.20309178758         1 0.4520484 0.7326707
## 98      0.68      5.78 0.01620954141         1 0.5316882 0.8007219
## 99      0.76     12.50 0.00040695202         1 0.6151339 0.8647833
## 100     0.80     16.82 0.00004109788         1 0.6585632 0.8949784
##                                                     method alternative .row
## 95  1-sample proportions test with continuity correction   two.sided    1
## 96  1-sample proportions test with continuity correction   two.sided    1
## 97  1-sample proportions test with continuity correction   two.sided    1
## 98  1-sample proportions test with continuity correction   two.sided    1
## 99  1-sample proportions test with continuity correction   two.sided    1
## 100 1-sample proportions test with continuity correction   two.sided    1
##     .index
## 95      95
## 96      96
## 97      97
## 98      98
## 99      99
## 100    100
```

Each sample gives us a slightly different estimate, and therefore, a different confidence interval as well.

Let's visualize this in a graph. Don't worry about all the complex code here; just look at the pretty picture that follows.

```r
# Don't worry about the syntax here.
# You won't have to do this on your own.
true_val <- 0.64
confint100 <- confint100 %>%
    mutate(color = ifelse(conf.low <= true_val & true_val <= conf.high,
                          "black", "red"),
           alpha = ifelse(color == "black", 0.5, 1))
ggplot(confint100, aes(x = estimate, y = .index,
                       color = color, alpha = alpha)) +
    geom_point() +
    scale_color_manual(values = c("black", "red"), guide = FALSE) +
    geom_segment(aes(x = conf.low, xend = conf.high, yend = .index)) +
    geom_vline(xintercept = true_val, color = "blue") +
    scale_alpha_identity() +
    labs(y = "Simulation", x = "Estimate")
```



For each of the 100 simulated intervals, most of them (the black ones) do capture the true value of 0.64 (the blue vertical line). Occasionally they don't (the red ones). We expect 5 red intervals, but since randomness is involved, it won't necessarily be exactly 5. (Here there were only 3 bad intervals.)

This is the key to interpreting confidence intervals. The "95%" in a 95% confidence interval means that if we were to collect many random samples, about 95% of them would contain the true population parameter and about 5% would not.

So let's revisit the erroneous statements from the beginning of this section and correct the misconceptions.

- ~~95% of the data lies in the interval.~~

  - This doesn't even make sense. Our data is categorical. The confidence interval is a range of plausible values for the proportion of successes in the sample.
- ~~There is a 95% chance that the sample proportion lies in the interval.~~
  - No. There is 100% chance that the sample proportion lies in the interval. In fact, the sample proportion is always in the dead center of the interval. That's the way we compute the interval: we take $\hat{p}$ and add and subtract the margin of error.
- ~~There is a 95% chance that the population parameter lies in the interval.~~
  - This is wrong in a more subtle way. The problem here as that it takes our interval as being fixed and special, and then tries to declare that of all possible population parameters, we have a 95% chance of the true one landing in our interval. The logic is backwards. The population parameter is the fixed truth. It doesn't wander around and land in our interval sometimes and not at other times. It is our confidence interval that wanders; it is just one of many intervals we could have obtained from random sampling. When we say, "We are 95% confident that…," we are just using a convenient shorthand for, "If we were to repeat the process of sampling and creating confidence intervals many times, about 95% of those times would produce an interval that happens to capture the actual population proportion." But we're lazy and we don't want to say that every time.

Conclusion

A confidence interval is a form of statistical inference that gives us a range of numbers in which we hope to capture the true population parameter. Of course, we can't be certain of that. If we repeatedly collect samples, the expectation is that 95% of those samples will produce confidence intervals that capture the true population parameter, but that also means that 5% will not. We'll never know if our sample was one of the 95% that worked, or one of the 5% that did not. And even if we get one of the intervals that worked, all we have is a range of values and it's impossible to determine which of those values is the true population parameter. Because it's statistics, we just have to live with that uncertainty.