Notes 10 - Confidence Intervals for a Single Value

STS 2300 Introduction to Data Analytics

Updated: 2025-04-06

Table of Contents

[Reading for Notes 10 1](#_Toc194846018)

[Learning Goals for Notes 10 1](#_Toc194846019)

[Constructing confidence intervals for 2](#_Toc194846020)

[Using a bootstrap distribution 2](#_Toc194846021)

[Using theory-based methods 3](#_Toc194846022)

[Constructing confidence intervals for 4](#_Toc194846023)

[Using a bootstrap distribution 4](#_Toc194846024)

[Using theory-based methods 5](#_Toc194846025)

[Interpreting confidence intervals for or 6](#_Toc194846026)

[Revisiting the Learning Goals for Notes 10 7](#_Toc194846027)

# Reading for Notes 10

The reading for Notes 09 - 11 is interwoven throughout [Sections 8.4 - 8.7](https://moderndive.com/8-confidence-intervals.html#bootstrap-process) of the Modern Dive textbook.

# Learning Goals for Notes 10

* Be able to construct confidence intervals for a population proportion and for a population mean using bootstrap resampling (infer package) and using theory-based methods.
* Be able to accurately interpret confidence intervals for a population proportion and for a population mean in context.

I’ll be using the following packages in this set of notes, so I’ll load them before I get started.

library(infer)  
library(dplyr)  
library(ggplot2)  
library(palmerpenguins)

# Constructing confidence intervals for

## Using a bootstrap distribution

In the previous set of notes, we saw how functions from the infer package can help us generate bootstrap distributions. In cases where we are interested in a single population proportion, our code will have the following form:

boot\_dist <- data |>   
 specify(formula = response ~ NULL, success = "level") |>   
 generate(reps = 1000, type = "bootstrap") |>   
 calculate(stat = "prop")

From this template, we will need to make the following changes:

* boot\_dist – replace with name for our object (if desired)
* data – replace with name of our data frame we are using
* response – replace with our variable of interest
* level – replace with category to find a proportion of

We previously saw an example of this with our House of Representatives data.

house\_of\_reps <- read.csv("https://raw.githubusercontent.com/nbussberg/STS2300-Spring2025/refs/heads/main/Data/house\_of\_reps.csv")  
  
set.seed(82720)  
HOR\_samp <- sample\_n(house\_of\_reps, size = 30)  
  
HOR\_boot <- HOR\_samp |>   
 specify(formula = party ~ NULL, success = "Democratic") |>   
 generate(reps = 1000, type = "bootstrap") |>   
 calculate(stat = "prop")

Remember that once we have a bootstrap distribution, we can use the get\_ci() function to calculate a confidence interval using either the standard error method or the percentile method.

**Practice:** Calculate a 90% confidence interval using both the SE method and the percentile method.

**Answer:**

**Question:** What do you think the related sampling distribution would look like? Consider how it would be similar to or different from our bootstrap distribution.

**Answer:**

## Using theory-based methods

The prop.test() function can be used to calculate theory-based confidence intervals for a proportion. The function requires us to specify:

* x – the number of “successes”
* n – the sample size
* conf.level – a number between 0 and 1 for our confidence level (default of 0.95)

**Application:** In our House of Representatives example, 17 of the 30 observations in the notes were part of the Democratic party. Make a 90% theory-based confidence interval for the proportion of the House of Representatives that belongs to the Democratic party.

**Code / Answer:**

Did you get something similar to your bootstrap distribution intervals?

When you perform statistical tests, each test has certain **assumptions**. When you are calculating a z-interval for a proportion, the following assumptions need to be true:

* Data comes from a random sample. Note: every confidence interval (or hypothesis test) method that we will talk about (including the bootstrap distribution ones) will require that our data comes from a random sample.
* We have “enough” successes and failures - usually at least 5 successes and 5 failures is considered sufficient to satisfy this condition.

# Constructing confidence intervals for

## Using a bootstrap distribution

In cases where we are interested in a single population mean, our code to generate a boostrap distribution will have the following form:

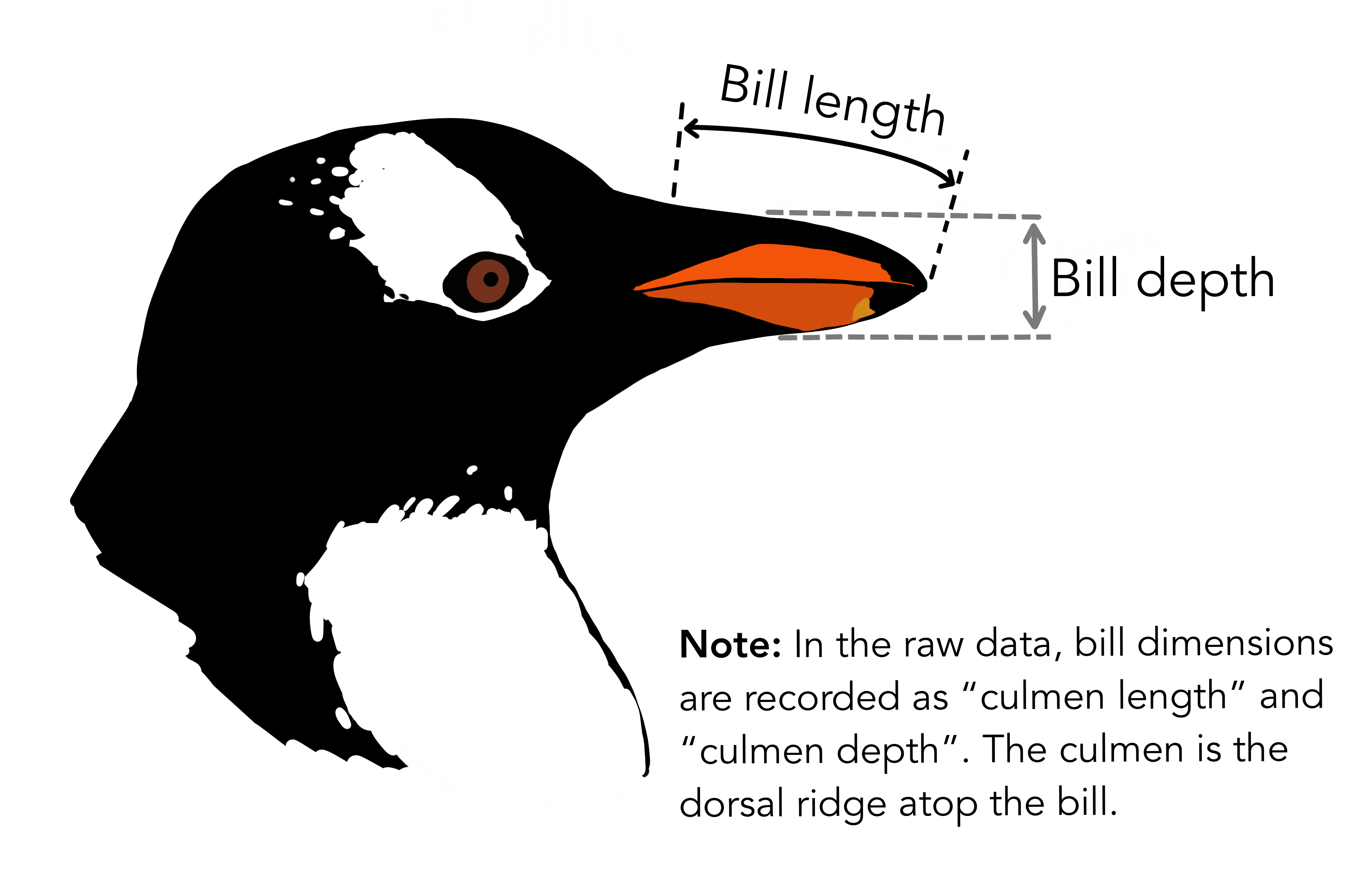
boot\_dist <- data |>   
 specify(formula = response ~ NULL) |>   
 generate(reps = 1000, type = "bootstrap") |>   
 calculate(stat = "mean")

From this template, we will need to make the following changes:

* boot\_dist – replace with name for our object (if desired)
* data – replace with name of our data frame we are using
* response – replace with our *quantitative* variable of interest

Notice that we no longer need to specify a success argument in specify() and that we use stat = "mean" instead of stat = "prop". Otherwise the process is the same as what we did for a population proportion.

For our example, let’s use the penguins data that we first saw in Activity 04. We will start by using this data as our sample to estimate the population mean () bill length (in mm) of Antarctic penguins.



Artwork by @allison\_horst

There are missing bill length values, so to find our estimate (), we will need to use the na.rm = TRUE argument in the mean() function.

bill\_xbar <- mean(penguins$bill\_length\_mm, na.rm = TRUE)  
bill\_xbar

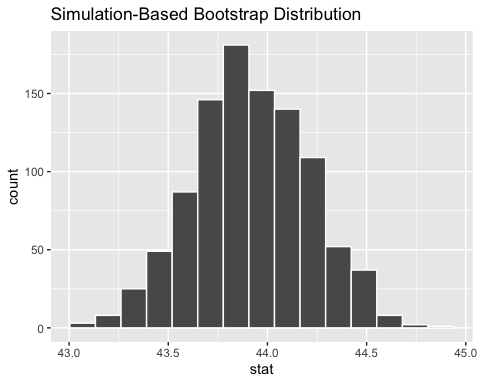
## [1] 43.92193

Now we can generate our bootstrap distribution. We should expect it to be centered on our sample mean of 43.92. (Remember we need to set a seed first!)

set.seed(4392)  
bill\_boot <- penguins |>   
 specify(formula = bill\_length\_mm ~ NULL) |>   
 generate(reps = 1000, type = "bootstrap") |>   
 calculate(stat = "mean")

## Warning: Removed 2 rows containing missing values.

visualize(bill\_boot)



**Practice:** Use the bootstrap distribution to calculate a 95% confidence interval for the population mean Antarctic penguin bill length using the SE method or the percentile method.

**Code and Answer:**

## Using theory-based methods

The t.test() function can be used to calculate theory-based confidence intervals for a mean. The function requires us to specify:

* x – a *vector* of our variable of interest
* conf.level – a number between 0 and 1 for our confidence level (default of 0.95)

**Application:** Calculate a 95% confidence interval for the population mean penguin bill length using the t.test() function.

**Code / Answer:**

Did you get something similar to your bootstrap distribution interval?

When you are calculating a t-interval for a mean, the following assumptions need to be true:

* Data comes from a random sample.
* Data comes from a population with a Normal distribution.

We can check the second assumption using a histogram or QQ plot to see how close our data is to Normal. However, as long as our sample size is “big enough,” this assumption won’t be as important. This is typically where you hear something like must be at least 30.

# Interpreting confidence intervals for or

We can think of each confidence interval as a net that we are casting out in hopes that they will “catch” the parameter (e.g., or ).

In the long run, the methods that we’re using should “catch” the parameter xx% of the time, where xx is the confidence level we choose. We will never know for sure if our specific interval contains the parameter of interest, but we can be xx% confident in the process we are using.

Thus, the interpretation of a 95% confidence interval for a single value (mean or proportion) would look something like:

**Generic Interpretation:** We are 95% confident that the population \_\_\_\_\_\_ is between \_\_\_ and \_\_\_.

* The first blank should be filled in by context that describes the specific proportion or mean we are interested in.
* The last two blanks should be filled in by the endpoints of our interval. Be sure to include units (when applicable) for these values.

**Practice:** Write an interpretation for a proportion interval we calculated in this set of notes.

**Answer:**

**Practice:** Write an interpretation for a mean interval we calculated in this set of notes.

**Answer:**

**Note:** When we make intervals for proportions, it is sometimes easier for the reader if we convert them to percentages. In these cases, we can rearrange our sentence a bit to make it flow better. Below is an example.

**House of Reps CI Interpretation (as a percentage):** We are 90% confident that between 43.2% and 70% of seats in the House of Representatives belong to the Democratic party.

**Review:** Notice that my interval is pretty wide in this example! It contains over a quarter of the possible values between 0 and 100%. What could I do if I wanted a narrower confidence interval?

**Answer:**

**Additional Practice:** Suppose I take a random sample of 30 Elon students and find that 8 of them are from North Carolina. I calculate a 95% confidence interval of (0.13, 0.46). Try writing an interpretation for this interval in context of the example.

**Interpretation:**

# Revisiting the Learning Goals for Notes 10

* Be able to construct confidence intervals for a population proportion and for a population mean using bootstrap resampling (infer package) and using theory-based methods.
* Be able to accurately interpret confidence intervals for a population proportion and for a population mean in context.