

STAT 201: Homework 5 (R)

Your name here

3/31/25

We'd like to learn about the median and mean year on U.S. pennies in circulation in 2025. Rather than getting access to all of the pennies that currently exist, we have a sample of 50-ish pennies obtained from a local bank.

Add code that loads in your package(s) for plotting/wrangling, and then run the following code chunk that also loads in the data frame as `pennies`.

1. In code that is *as reproducible as possible*, obtain two bootstrap distributions: one for the sample median year of pennies in circulation in 2025, and one for sample mean year. You should *not* need to run two separate `for` loops.

At the top of the code chunk, set a seed of 100. Run your bootstrap distribution for 5000 iterations.

2. Calculate and store the bounds of a 95% bootstrap confidence interval for each parameter of interest. Don't display any output here.
3. Now, let's visualize your bootstrap distribution of sample *medians* along with the corresponding confidence interval. Write code that creates a histogram of the bootstrap distribution that includes:
 - Vertical dashed lines at the lower and upper interval bounds in a color of your choice (remember the `geom_vline()` function. To make the lines dashed, specify the argument `linetype = "dashed"`)
 - Meaningful title and x-axis label
 - A caption that briefly explains what the lines represent
 - The bins should correspond to a single year
4. Using in-line code, interpret your two confidence intervals in context. You should include the notation of the population of interest in your interpretation.

Answer:

5. Obtain approximations of the standard errors of the sample median year and sample mean year of U.S. pennies in circulation in 2025. Store these two quantities as variables in R (don't show any output).

Then using in-line code, report these standard errors. Which approximate standard error is larger? What does this imply about the behavior of sample medians vs means?

(If the trailing decimals bother you, you can use the function `round(x, d)` where `x` is a number you'd like to round, and `d` tells us how many decimals to round to.)

Answer: