# Lab 1 – Sampling and Simulation with Diamond Prices

**NAME 1 – NETID**

**NAME 2 – NETID [if applicable]**

**NAME 3 – NETID [if applicable]**

## Formatting Instructions

- Please include all requested responses in a document, then save it as a **pdf** when done.
  - You may use this instructions document, or you may create a new document.
  - All responses should be numbered (leaving the original question text is optional!)
- Upload your report as a **pdf** to **Gradescope** when finished.
  - Please see the instructions on the Canvas assignment page about **matching pages** and **adding group members to your submission** if working with others.

## Assignment Overview

- For this lab, we will explore the diamonds dataset stored in the tidyverse package.

**Step 0** – *Come to Office Hours or Lab Day for help if you get stuck somewhere in Step 0!*

- **Complete all Pre-lab work identified on the Canvas assignment page**
  - Did you install R and RStudio ***OR*** create an RStudio Cloud Account?
  - Did you go through both pre-lab tutorials for Lab 1?
- **Open RStudio** (or RStudio Cloud) to get started
  - Be careful **not** to open **R** (this icon with just R and a swirly thing on the left).
  - Open up **RStudio** (this icon with the blue circle on the right!).

- **Open the starter script** linked in the assignment description.
  - I don't recommend coding directly into the console (command line). Coding in your script is much easier for editing your code, saving your code, and making comments for what each code does (video 3!)
- **Install and library tidyverse**
  - Write and run the following code: `install.packages("tidyverse")`
  - This will take a minute or two! Wait until the little stop sign disappears to proceed.
  - Next, you will want to run the following code: `library(tidyverse)`
- **Open the Data**
  - We will be using the `diamonds` data frame stored in the tidyverse package.
  - After librarying tidyverse, run the code: `View(diamonds)`.
  - Each row represents one diamond from a collection of over 53,000. We will treat this as our "population."

---

The diamonds dataset is a catalogue of over 53,000 diamonds. It includes a number of variables about each diamond, one of which is how many "carats" it is. A carat is a tiny unit of weight. It's about 0.2 grams!

**Question 1** (3pts): Create a histogram of the `carat` variable from the full diamonds dataset (*Think of this as your "population distribution" for the carat variable).* Set `breaks = 20` to keep a consistent number of bins.

**Include the image of your histogram in your report.** You may either save it to your computer and upload it, or include a properly cropped screenshot. *Including your code is optional.*

**Would you describe this distribution as symmetric or skewed?**

**Question 2** (3pts) Calculate the mean and standard deviation of the `price` variable. *This is your population mean and standard deviation.*

**Include the population mean and standard deviation values in your report.**

**Question 3** (5pts): Imagine that this collection of 53,000 diamonds is in front of you in a giant bin. You don't have time to take the carat measurements for all 53,000, so you'd just like to take 50 diamonds as a representative sample of this collection.

- Use R to take a random sample of 50 from this dataset, specifically sampling the carat measure. Sample without replacement (this will be the default option)
- Name this vector `fifty_diam` (If saved properly, you will see this in your global environment with a length of 50!)

Create a histogram of the `fifty_diam` vector. Set `breaks = 20` to keep a consistent number of bins.

**Include the image of this histogram in your report**

**If you were to take a much larger sample, the shape of your sample data distribution would look more and more like…what?** If you're not sure what we mean by this, check Chapter 3 again!

**Question 4** (5pts) Let's see how well your sample of 50 represents this population. Calculate the mean and standard deviation of the carat values you sampled. *Note that these values will change if you take a new sample, and that's ok! Just report the values you get for one particular sample.*

**Include your sample mean and standard deviation values in your report**

**What is the *absolute* error of *your* sample mean as an estimate of the population mean?** If you're not sure what we mean by this, check the second page of Chapter 3 again!

**Question 5** (5pts): But in general, would taking the mean carat measure from a sample of 50 provide a good estimate for the mean of the population? How far off would we generally expect to be?

To address this question, set up a `for` loop to simulate taking a sample of size 50 *at least* 10,000 times. Inside your loop, calculate the mean price and save it to a vector called `means_fifty`. *Please reference the entire "For Loops: Returning a Vector" section of the "Sampling and Simulation" tutorial for assistance on this part.*

After successfully running your simulation, create a histogram of your `means_fifty` vector and set `breaks` = 20 to keep a consistent number of bins.

**Include the image of your histogram in your report**

**Include the R code you used to generate this loop**

**Question 6** (5pts): As you should notice from your histogram, our sample means will vary with each sample we take. Calculate the standard deviation of the simulated sample means (`means_fifty` vector) you created.

**Include this standard deviation value in your report**

If you run the loop again and recalculate the standard deviation, you'll likely find that the number changed a little bit! What is the standard deviation of the simulated means approximating? **Report the <u>name</u> of this measure that we learned in the notes and calculate the <u>true value</u> for this measure using the formula we learned.** Check pages 3 and 9 of Chapter 3 if you're not sure!

**Question 7** (5pts): Repeat question 5, but now consider if we were only taking a sample of 10 diamonds rather than a sample of 50. Call your vector of sample means `means_ten`. After successfully running your simulation, create a histogram of your `means_ten` vector. Again, set `breaks = 20` to keep a consistent number of bins.

**Include the image of your histogram in your report**

**Include the R code you used to generate this loop**

**Question 8** (4pts) Let's compare the distribution of sample means when we took samples of size 50 versus when we took samples of size 10

**Is there any difference in the shapes of these distributions?**

**What is the Central Limit Theorem, and how does this relate to what you found in your previous answer?**