# Lab 5 - Potpourri

## Introduction

In this lab you'll review and get practice with a variety of concepts, methods, and tools you've encountered thus far.

## Part 1 - All about Quarto

### Question 1

a. Add each of strings below to the code chunk provided in your document, render the document, and determine if the string is a proper code chunk label. If not, explain why and describe how you could fix it so the document renders.

    - Chunk label 1:

```
#| label: a-label
#| with-a-line-break
```

    - Chunk label 2:

```
#| label: areaaaaaaaaaaaaaaaaaaaaaallllllllllllllllllyyyyylooooooooooooooooooonglabel
```

    - Chunk label 3:

```
#| label: label with spaces
```

    - Chunk label 4:

```
#| label: label-with-dashes
```

Tip

Try each label option in the code chunk provided in your document. If it gives you an error (document doesn't render), you know it's not a proper chunk label.

b. Which of the chunk label options above is the best option? Explain your reasoning in 1-2 sentences.

## Question 2

You have the following code chunk:

```r
ggplot(penguins, aes(x = bill_length_mm, y = bill_depth_mm)) +
  geom_point()
```

Add the following code chunk options, one at a time, and set each to `false` and then to `true`. After each value, render your document and observe its effect. Ultimately, choose the values that are the most appropriate for this code chunk. Based on the behaviors you observe, describe what each code chunk option does.

- `echo`
- `warning`
- `eval`

## Question 3

a. You have the following code chunk again.

   ```r
   ggplot(penguins, aes(x = bill_length_mm, y = bill_depth_mm)) +
     geom_point()
   ```

   Add `fig-width` and `fig-asp` as code chunk options. Try setting `fig-width` to values between 1 and 10. Try setting `fig-asp` to values between 0.1 and 1. Re-render the document after each value and observe its effect. Ultimately, choose values that make the plot look visually pleasing in the rendered document. Based on the behavior you observe, describe what each chunk option does.

b. You have the following code chunk, but look carefully, it's not exactly the same!

   ```r
   gplot(penguins, aes(x = bill_length_mm, y = bill_depth_mm)) +
     geom_point()
   ```

Add `error` as a code chunk option and set it to `false` and then set it to `true`. After each value, render your document and observe its effect. Ultimately, choose the value that allows you to render your document without altering the code. Based on the behavior you observe, describe what this code chunk option does.

Tip

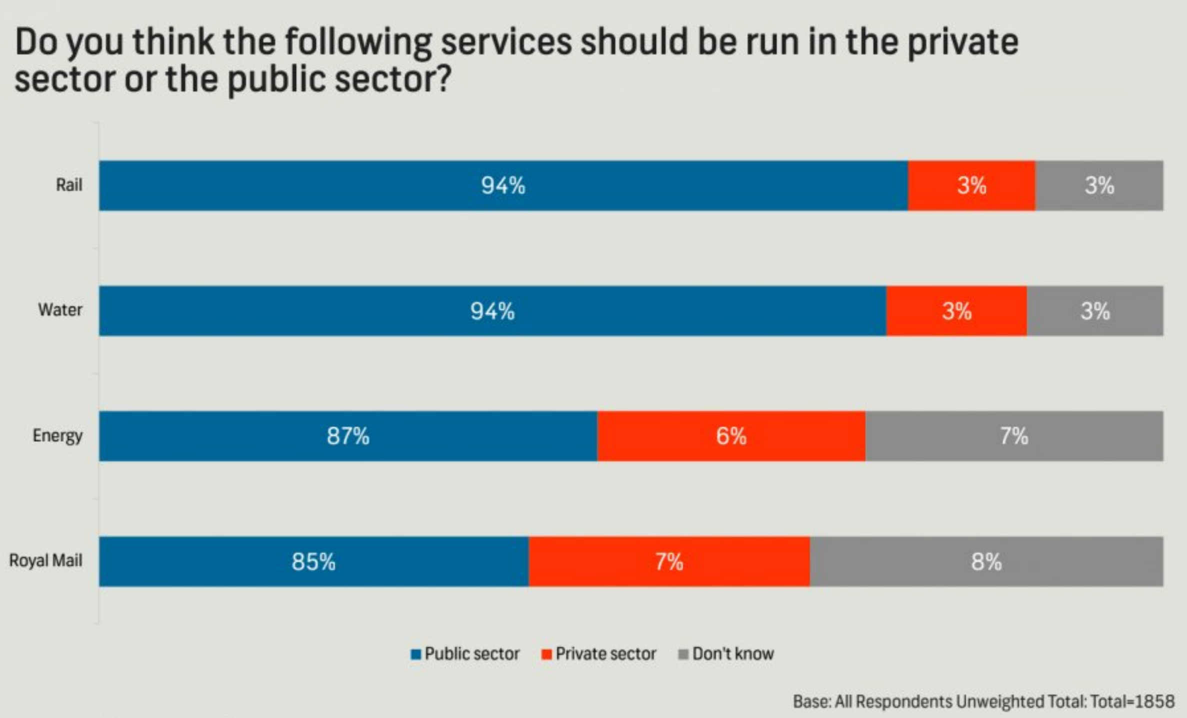
Reading the documentation might also be hepful.

Render, commit, and push one last time. Make sure that you commit and push all changed documents and your Git pane is completely empty before proceeding.

## Part 2 - Misrepresentation

### Question 4

The following chart was shared by @GraphCrimes on X/Twitter on September 3, 2022.



a. What is misleading about this graph?

b. Suppose you wanted to recreate this plot, with improvements to avoid its misleading pitfalls from part (a). You would obviously need the data from the survey in order to be able to do that. How many observations would this data have? How many variables (at least) should it have, and what should those variables be?
c. Load the data for this survey from `data/survation.csv`. Confirm that the data match the percentages from the visualization. That is, calculate the percentages of public sector, private sector, don't know for each of the services and check that they match the percentages from the plot.

### Question 5

Create an improved version of the visualization. Your improved visualization:

- should also be a stacked bar chart with services on the y-axis, presented in the same order as the original plot, and services to create the segments of the plot, and presented in the same order as the original plot
- should have the same legend location
- should have the same title and caption
- does not need to have a bolded title or a gray background

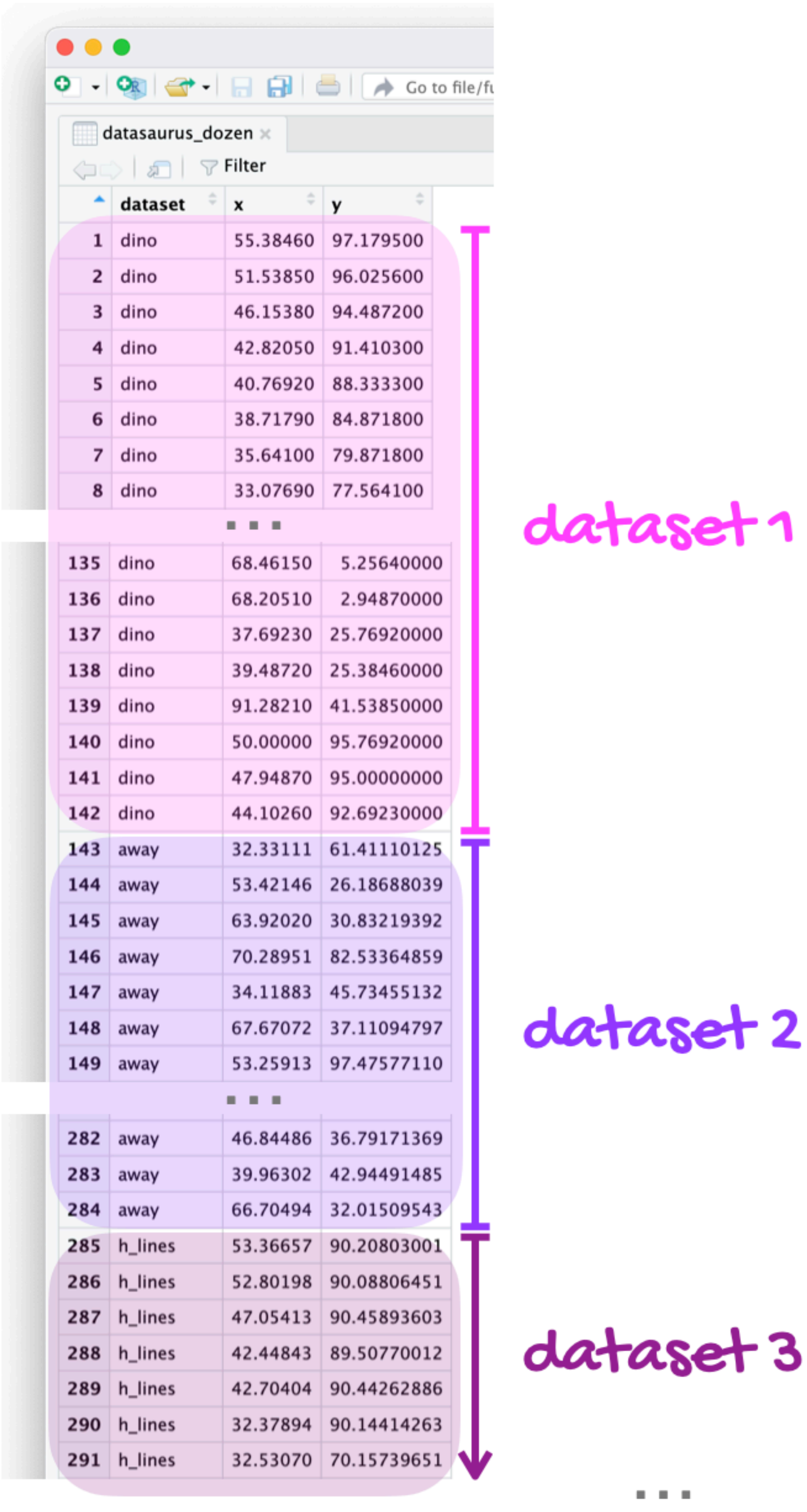
How does the improved visualization look different than the original? Does it send a different message at a first glance?

> Tip
>
> Use `\n` to add a line break to your title. And note that since the title is very long, it might run off the page in your code. That's ok!
> Additionally, the colors used in the plot are `gray`, `#FF3205`, and `#006697`.

Render, commit, and push one last time. Make sure that you commit and push all changed documents and your Git pane is completely empty before proceeding.

## Part 3 - DatasauRus

The data frame you will be working with in this part is called `datasaurus_dozen` and it's in the **datasauRus** package. This single data frame contains 13 datasets, designed to show us why data visualization is important and how summary statistics alone can be misleading. The different datasets are marked by the `dataset` variable, as shown in Figure 1.



| | dataset | x | y |
|---|---|---|---|
| 1 | dino | 55.38460 | 97.179500 |
| 2 | dino | 51.53850 | 96.025600 |
| 3 | dino | 46.15380 | 94.487200 |
| 4 | dino | 42.82050 | 91.410300 |
| 5 | dino | 40.76920 | 88.333300 |
| 6 | dino | 38.71790 | 84.871800 |
| 7 | dino | 35.64100 | 79.871800 |
| 8 | dino | 33.07690 | 77.564100 |
| | ■ ■ ■ | | |
| 135 | dino | 68.46150 | 5.25640000 |
| 136 | dino | 68.20510 | 2.94870000 |
| 137 | dino | 37.69230 | 25.76920000 |
| 138 | dino | 39.48720 | 25.38460000 |
| 139 | dino | 91.28210 | 41.53850000 |
| 140 | dino | 50.00000 | 95.76920000 |
| 141 | dino | 47.94870 | 95.00000000 |
| 142 | dino | 44.10260 | 92.69230000 |
| 143 | away | 32.33111 | 61.41110125 |
| 144 | away | 53.42146 | 26.18688039 |
| 145 | away | 63.92020 | 30.83219392 |
| 146 | away | 70.28951 | 82.53364859 |
| 147 | away | 34.11883 | 45.73455132 |
| 148 | away | 67.67072 | 37.11094797 |
| 149 | away | 53.25913 | 97.47577110 |
| | ■ ■ ■ | | |
| 282 | away | 46.84486 | 36.79171369 |
| 283 | away | 39.96302 | 42.94491485 |
| 284 | away | 66.70494 | 32.01509543 |
| 285 | h_lines | 53.36657 | 90.20803001 |
| 286 | h_lines | 52.80198 | 90.08806451 |
| 287 | h_lines | 47.05413 | 90.45893603 |
| 288 | h_lines | 42.44843 | 89.50770012 |
| 289 | h_lines | 42.70404 | 90.44262886 |
| 290 | h_lines | 32.37894 | 90.14414263 |
| 291 | h_lines | 32.53070 | 70.15739651 |

Figure 1: The ‘datasaurus_dozen‘ data frame stacks 13 datasets on top of each other. This figure shows the first three datasets.

Note

If it’s confusing that the data frame is called `datasaurus_dozen` when it contains 13 datasets, you’re not alone! Have you heard of a baker’s dozen?

Here is a peek at the top 10 rows of the dataset:

```r
datasaurus_dozen
```

```
# A tibble: 1,846 x 3
   dataset     x     y
   <chr>   <dbl> <dbl>
 1 dino     55.4  97.2
 2 dino     51.5  96.0
 3 dino     46.2  94.5
 4 dino     42.8  91.4
 5 dino     40.8  88.3
 6 dino     38.7  84.9
 7 dino     35.6  79.9
 8 dino     33.1  77.6
 9 dino     29.0  74.5
10 dino     26.2  71.4
# i 1,836 more rows
```

### Question 6

In a single pipeline, calculate the mean of `x`, mean of `y`, standard deviation of `x`, standard deviation of `y`, and the correlation between `x` and `y` for each level of the `dataset` variable. Then, in 1-2 sentences, comment on how these summary statistics compare across groups (datasets).

Tip

There are 13 groups but `tibbles` only print out 10 rows by default. Add `print(n = 13)` as the last step of your pipeline to display all rows.

### Question 7

Create a scatterplot of `y` versus `x` and color and facet it by `dataset`. Then, in 1-2 sentences, how these plots compare across groups (datasets). How does your response in this question compare to your response to the previous question and what does this say about using visualizations and summary statistics when getting to know a dataset?

Tip

When you both color and facet by the same variable, you'll end up with a redundant legend. Turn off the legend by adding `show.legend = FALSE` to the geom creating the legend.

Render, commit, and push one last time. Make sure that you commit and push all changed documents and your Git pane is completely empty before proceeding.

## Part 4 - Election polling

SurveyUSA polled 1,500 US adults between January 31, 2024 and February 2, 2024. Of the 1,500 adults, 1,259 were identified by SurveyUSA as being registered to vote, and of these 1,048 were found to be likely to vote in the 2024 November election for President.[1] The following question was asked to these 1,048 adults:

> 1,048 were found to be likely to vote in the 2024 November election for President and were asked the substantive questions which follow.

Responses were broken down into the following categories:

| Variable | Levels |
|---|---|
| Age | 18-49; 50+ |
| Vote | Donald Trump (R); Joe Biden (D); Undecided |

Of the 1,048 responses, 507 were between the ages of 18-49. Of the individuals that are between 18-49, 238 individuals responded that they would vote for Donald Trump, 237 said they would vote for Joe Biden, and the remainder were undecided. Of the individuals that are 50+, 271 individuals responded that they would vote for Donald Trump, 228 said they would vote for Joe Biden, and the remainder were undecided.

[1] Full survey results can be found at https://www.surveyusa.com/client/PollReport.aspx?g=300d50f5-303b-4652-b59e-6fbf1b87e24a.

## Question 8

a. Fill in the code below to create a two-way table that summarizes these data.

```
survey_counts <- tibble(
  age = c(),
  vote = c(),
  n = c()
  )

survey_counts |>
  pivot_wider(
    names_from = ___,
    values_from = ___
  ) |>
  kable()
```

For parts b-d below, use a your response single pipeline starting with `survey_counts`, calculate the desired proportions, and make sure the result is an **ungrouped** data frame with a column for relevant counts, a column for relevant proportions, and a column for the groups you're interested in.

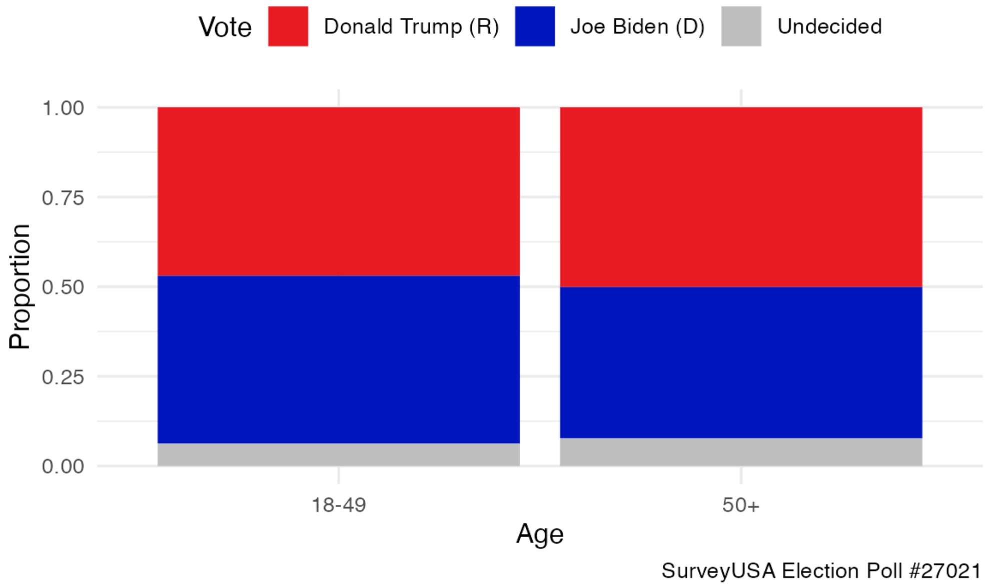
b. Calculate the proportions of 18-49 year olds and 50+ year olds in this sample.

c. Calculate the proportions of those who want to vote for Donald Trump, Joe Biden, and those who are undecided in this sample.

d. Calculate the proportions of individuals in this sample who are planning to vote for each of the candidates or are undecided among those who are 18-49 years old as well as among those who are 50+ years old.

## Question 9

a. Re-create the following visualization that displays relationship between `age` and `vote`.



> **Tip**
>
> The colors used in the plot are `"#E81B23"`, `"#0015BC"`, and `"gray"`. The theme is `theme_minimal()`.

b. Based on your calculations so far, as well as your visualization, write 1-3 sentences that describe the relationship, in this sample, between age and plans for presidential vote.

Render, commit, and push one last time. Make sure that you commit and push all changed documents and your Git pane is completely empty before proceeding.

## Part 5 - Ethics

### Question 10

To complete this exercise you will first need to watch the documentary Coded Bias. To do so, you either need to be on the Duke network or connected to the Duke VPN. Then go to https://find.library.duke.edu/catalog/DUKE009834953 and click on "View Online". Once you watch the video, write a reflection in 2-5 bullet points highlighting at least one thing that you already knew about (from the course prep materials) and at least one thing you learned from the movie as well as any other aspects of the documentary that you found interesting / enlightening.

! Important

This question requires no code, only narrative. Remember that, based on the syllabus, you may not use generative AI tools (e.g., Chat GPT) to write narrative on assignments.

Render, commit, and push one last time. Make sure that you commit and push all changed documents and your Git pane is completely empty before proceeding.

## Wrap-up

### Submission

Once you are finished with the lab, you will submit your final PDF document to Gradescope.

⚠ Warning

Before you wrap up the assignment, make sure all of your documents are updated on your GitHub repo. We will be checking these to make sure you have been practicing how to commit and push changes.
You must turn in a PDF file to the Gradescope page by the submission deadline to be considered "on time".

To submit your assignment:

- Go to http://www.gradescope.com and click *Log in* in the top right corner.
- Click *School Credentials* → *Duke NetID* and log in using your NetID credentials.
- Click on your *STA 199* course.
- Click on the assignment, and you'll be prompted to submit it.
- Mark all the pages associated with question. All the pages of your lab should be associated with at least one question (i.e., should be "checked").

! Checklist

Make sure you have:

- attempted all questions
- rendered your Quarto document
- committed and pushed everything to your GitHub repository such that the Git pane in RStudio is empty
- uploaded your PDF to Gradescope
- selected pages associated with each question on Gradescope

## Grading

The lab is graded out of a total of 50 points.

You can earn up to 5 points on each question:

- 5: Response shows excellent understanding and addresses all or almost all of the rubric items.
- 4: Response shows good understanding and addresses most of the rubric items.
- 3: Response shows understanding and addresses a majority of the rubric items.
- 2: Response shows effort and misses many of the rubric items.
- 1: Response does not show sufficient effort or understanding and/or is largely incomplete.
- 0: No attempt.