Group 1-4 Project 1 - Data Carpentry & Visualization Basics

Ryo Iwata Stephan Quintin Zachary Strickland

Group GROUPNUMBERHERE members:

names here Note who is assigned leader

Group Assignment 1

Instructions - Everyone should work on their own computer on a separate copy of this, then work together to create a single very well polished version to submit for the assignment. The assigned group leader should submit the assignment for the group, but every individual should thoroughly understand every answer. In most cases, everyone should roughly be on the same question as the same time when working together simultaneously (e.g., in class or during group meetings). When asked, you (all) need to be ready to explain the thought process behind the plan, approach, revisions, and final code that was written. Now is the time to build your understanding of the language and approach to these data carpentry and visualization methods. My expectation is that every submitted assignment will be nearly perfect given how we will be approaching this series of problems both in class and out.

I have copied together all the questions in one place so everyone doesn't have to waste their time doing so, but these are directly from the reading (R4DS 2E). I am leaving the formatting plain, so refer to the book itself for the example plots and pretty formatting.

Please make it easy to grade this! Your text answers should be in the main document (i.e., not in a code chunk), and *please make them bold so they stand out.* Write code, including using comments, as if you are writing code that will be published. Build good habits now!

1.2.5

1

How many rows are in penguins? How many columns?

Answer: 344 rows and 8 columns

```r
str(penguins)
```

```
## tibble [344 x 8] (S3: tbl_df/tbl/data.frame)
##  $ species          : Factor w/ 3 levels "Adelie","Chinstrap",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ island           : Factor w/ 3 levels "Biscoe","Dream",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ bill_length_mm   : num [1:344] 39.1 39.5 40.3 NA 36.7 39.3 38.9 39.2 34.1 42 ...
##  $ bill_depth_mm    : num [1:344] 18.7 17.4 18 NA 19.3 20.6 17.8 19.6 18.1 20.2 ...
##  $ flipper_length_mm: int [1:344] 181 186 195 NA 193 190 181 195 193 190 ...
##  $ body_mass_g      : int [1:344] 3750 3800 3250 NA 3450 3650 3625 4675 3475 4250 ...
##  $ sex              : Factor w/ 2 levels "female","male": 2 1 1 NA 1 2 1 2 NA NA ...
##  $ year             : int [1:344] 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 ...
```

2

What does the bill_depth_mm variable in the penguins data frame describe? Read the help for ?penguins to find out.

3

Make a scatterplot of bill_depth_mm vs. bill_length_mm. That is, make a scatterplot with bill_depth_mm on the y-axis and bill_length_mm on the x-axis. Describe the relationship between these two variables.

4

What happens if you make a scatterplot of species vs. bill_depth_mm? What might be a better choice of geom?

5

Why does the following give an error and how would you fix it?

```
#ggplot(data = penguins) +
#  geom_point()
```

6

What does the na.rm argument do in geom_point()? What is the default value of the argument? Create a scatterplot where you successfully use this argument set to TRUE.

7

Add the following caption to the plot you made in the previous exercise: “Data come from the palmerpenguins package.” Hint: Take a look at the documentation for labs().

8

Recreate the following visualization. What aesthetic should bill_depth_mm be mapped to? And should it be mapped at the global level or at the geom level?

9

Run this code in your head and predict what the output will look like.

ggplot(data = penguins, mapping = aes(x = flipper_length_mm, y = body_mass_g, color = island)) + geom_point() + geom_smooth(se = FALSE)

9b

10

Will these two graphs look different? Why/why not?

ggplot(data = penguins, mapping = aes(x = flipper_length_mm, y = body_mass_g)) + geom_point() + geom_smooth()

ggplot() + geom_point(data = penguins, mapping = aes(x = flipper_length_mm, y = body_mass_g)) + geom_smooth(data = penguins, mapping = aes(x = flipper_length_mm, y = body_mass_g))

10b confirm your answer:

1.4.3 Exercises

1

Make a bar plot of species of penguins, where you assign species to the y aesthetic. How is this plot different?

2

How are the following two plots different? Which aesthetic, color or fill, is more useful for changing the color of bars?

```
ggplot(penguins, aes(x = species)) +
  geom_bar(color = "red")

ggplot(penguins, aes(x = species)) +
  geom_bar(fill = "red")
```

3

What does the bins argument in geom_histogram() do?

4

Make a histogram of the carat variable in the diamonds dataset that is available when you load the tidyverse package. Experiment with different binwidths. What binwidth reveals the most interesting patterns?

1.5.5 Exercises

1

The mpg data frame that is bundled with the ggplot2 package contains 234 observations collected by the US Environmental Protection Agency on 38 car models. Which variables in mpg are categorical? Which variables are numerical? (Hint: Type ?mpg to read the documentation for the dataset.) How can you see this information when you run mpg?

2

Make a scatterplot of hwy vs. displ using the mpg data frame. Next, map a third, numerical variable to color, then size, then both color and size, then shape. How do these aesthetics behave differently for categorical vs. numerical variables?

3

In the scatterplot of hwy vs. displ, what happens if you map a third variable to linewidth?

4

What happens if you map the same variable to multiple aesthetics?

5

Make a scatterplot of bill_depth_mm vs. bill_length_mm and color the points by species. What does adding coloring by species reveal about the relationship between these two variables? What about faceting by species?

6

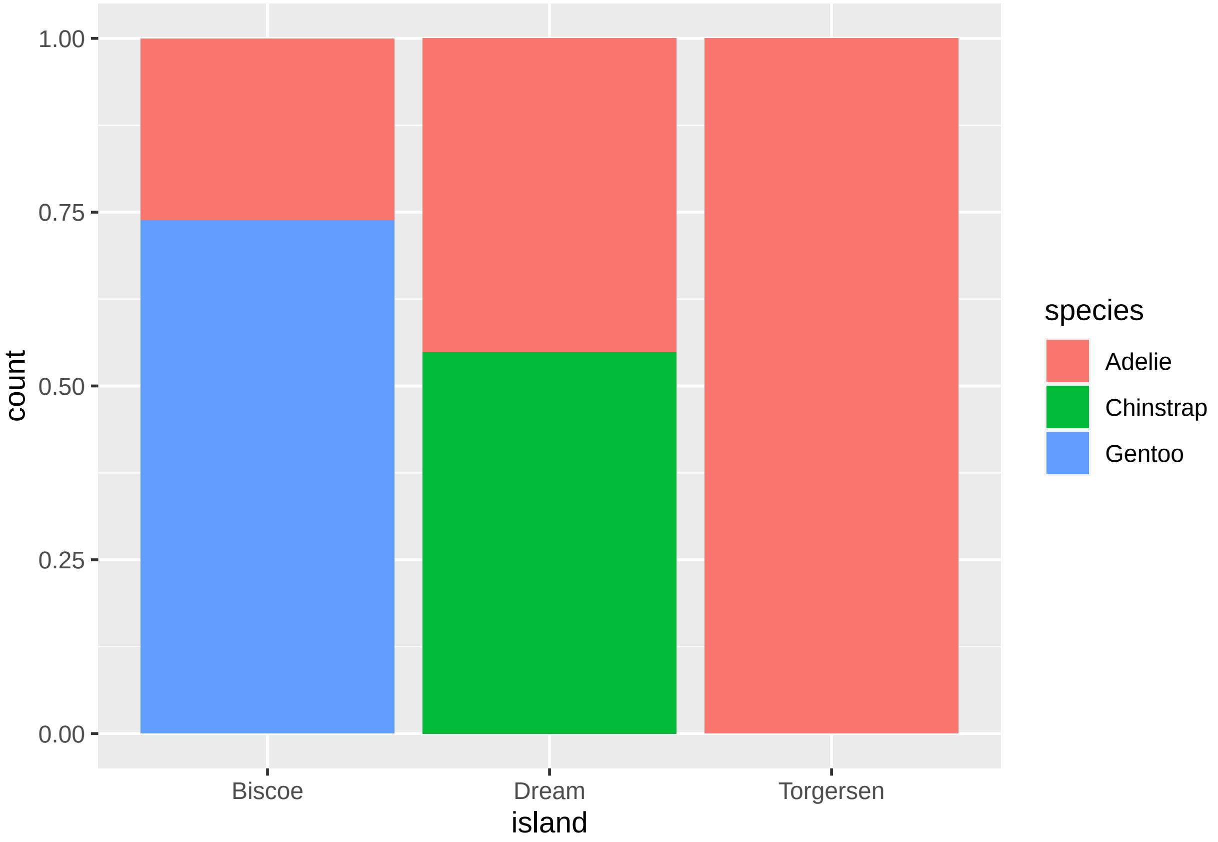
Why does the following yield two separate legends? How would you fix it to combine the two legends?

ggplot(data = penguins, mapping = aes(x = bill_length_mm, y = bill_depth_mm, color = species, shape = species)) + geom_point() + labs(color = “Species”)

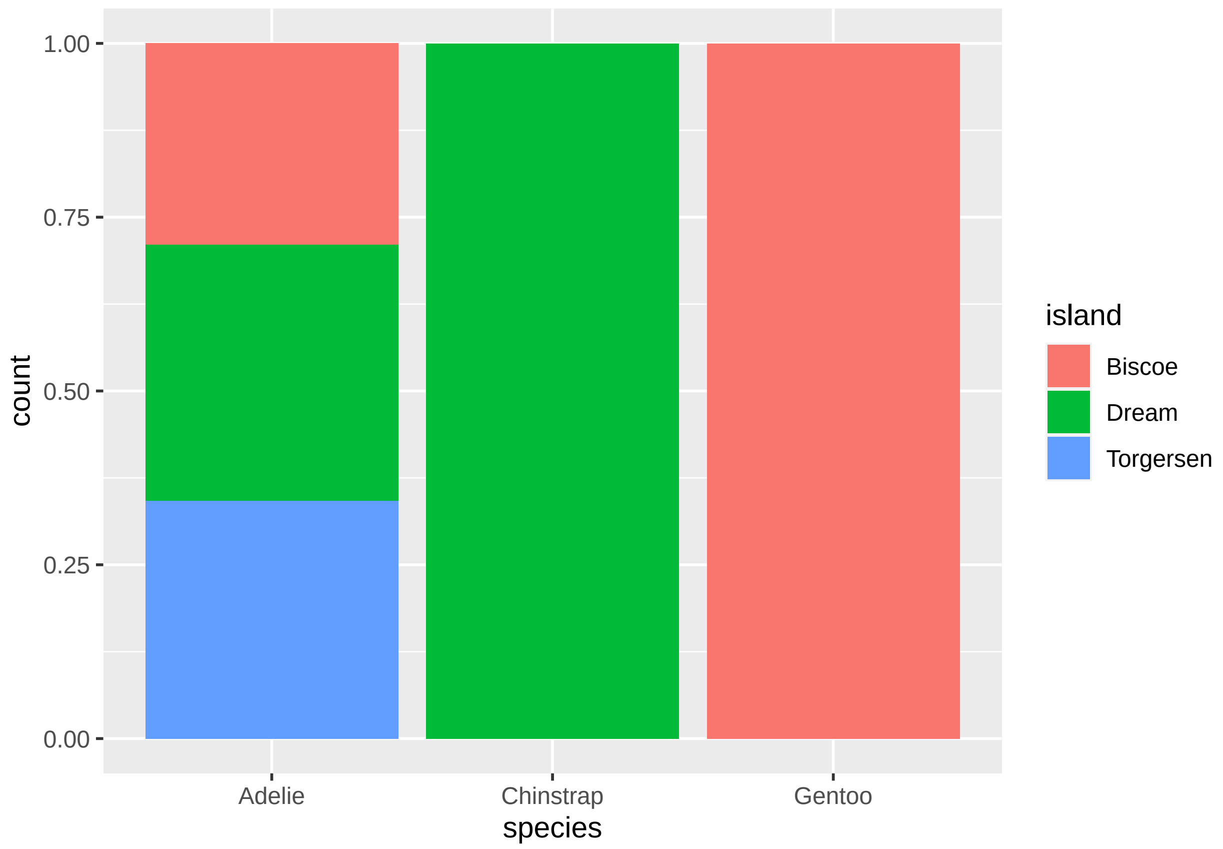
7

Create the two following stacked bar plots. Which question can you answer with the first one? Which question can you answer with the second one?

```r
ggplot(penguins, aes(x = island, fill = species)) +
  geom_bar(position = "fill")
```



```r
ggplot(penguins, aes(x = species, fill = island)) +
  geom_bar(position = "fill")
```



2.5 Exercises

1

Why does this code not work?

my_variable <- 10 my_varıable #> Error in eval(expr, envir, enclos): object 'my_varıable' not found

Look carefully! (This may seem like an exercise in pointlessness, but training your brain to notice even the tiniest difference will pay off when programming.)

2

Tweak each of the following R commands so that they run correctly:

```
# remove the comment, i.e. # symbol from the below lines by selecting and pressing control-shift-c (i.e
# libary(todyverse)
#
# ggplot(dTA = mpg) +
#   geom_point(maping = aes(x = displ y = hwy)) +
#   geom_smooth(method = "lm)
```

3

Press Option + Shift + K / Alt + Shift + K. What happens? How can you get to the same place using the menus?

4

Let's revisit an exercise from the Section 1.6. Run the following lines of code. Which of the two plots is saved as mpg-plot.png? Why?

```
my_bar_plot <- ggplot(mpg, aes(x = class)) +
  geom_bar()
my_scatter_plot <- ggplot(mpg, aes(x = cty, y = hwy)) +
  geom_point()
ggsave(filename = "mpg-plot.png", plot = my_bar_plot)
```

```
## Saving 6.5 x 4.5 in image
```

3.2.5 Exercises

1

In a single pipeline for each condition, find all flights that meet the condition: Had an arrival delay of two or more hours Flew to Houston (IAH or HOU) Were operated by United, American, or Delta Departed in summer (July, August, and September) Arrived more than two hours late, but didn't leave late Were delayed by at least an hour, but made up over 30 minutes in flight

2

Sort flights to find the flights with longest departure delays. Find the flights that left earliest in the morning.

3

Sort flights to find the fastest flights. (Hint: Try including a math calculation inside of your function.)

4

Was there a flight on every day of 2013?

5

Which flights traveled the farthest distance? Which traveled the least distance?

6

Does it matter what order you used filter() and arrange() if you're using both? Why/why not? Think about the results and how much work the functions would have to do.

3.3.5 Exercises

1

Compare dep_time, sched_dep_time, and dep_delay. How would you expect those three numbers to be related?

2

Brainstorm as many ways as possible to select dep_time, dep_delay, arr_time, and arr_delay from flights.

3

What happens if you specify the name of the same variable multiple times in a select() call?

4

What does the any_of() function do? Why might it be helpful in conjunction with this vector?

variables <- c(“year”, “month”, “day”, “dep_delay”, “arr_delay”)

5

Does the result of running the following code surprise you? How do the select helpers deal with upper and lower case by default? How can you change that default?

flights |> select(contains(“TIME”))

6

Rename air_time to air_time_min to indicate units of measurement and move it to the beginning of the data frame.

7

Why doesn’t the following work, and what does the error mean?

```
# remove comments by selecting all desired lines of code and using ctrl-C (control-shift-c) i.e. commen
#
# flights |>
#   select(tailnum) |>
#   arrange(arr_delay)
# #> Error in `arrange()`:
# #> i In argument: `..1 = arr_delay`.
# #> Caused by error:
# #> ! object 'arr_delay' not found
```

3.5.7 Exercises

1

Which carrier has the worst average delays? Challenge: can you disentangle the effects of bad airports vs. bad carriers? Why/why not? (Hint: think about flights |> group_by(carrier, dest) |> summarize(n()))

2

Find the flights that are most delayed upon departure from each destination.

3

How do delays vary over the course of the day. Illustrate your answer with a plot.

4

What happens if you supply a negative n to slice_min() and friends?

5

Explain what count() does in terms of the dplyr verbs you just learned. What does the sort argument to count() do?

6

Suppose we have the following tiny data frame:

```
    # df <- tibble(
    #   x = 1:5,
    #   y = c("a", "b", "a", "a", "b"),
    #   z = c("K", "K", "L", "L", "K")
    # )
```

6.a Write down what you think the output will look like, then check if you were correct, and describe what group_by() does.

```
# df |>
#   group_by(y)
```

6.b Write down what you think the output will look like, then check if you were correct, and describe what arrange() does. Also comment on how it's different from the group_by() in part (a)?

```
#
# df |>
#   arrange(y)
```

6.c Write down what you think the output will look like, then check if you were correct, and describe what the pipeline does.

```
# df |>
#   group_by(y) |>
#   summarize(mean_x = mean(x))
```

6.d Write down what you think the output will look like, then check if you were correct, and describe what the pipeline does. Then, comment on what the message says.

```
# df |>
#   group_by(y, z) |>
#   summarize(mean_x = mean(x))
```

6.e Write down what you think the output will look like, then check if you were correct, and describe what the pipeline does. How is the output different from the one in part (d).

```
# df |>
#   group_by(y, z) |>
#   summarize(mean_x = mean(x), .groups = "drop")
```

6.f Write down what you think the outputs will look like, then check if you were correct, and describe what each pipeline does. How are the outputs of the two pipelines different?

```
# df |>
#   group_by(y, z) |>
#   summarize(mean_x = mean(x))
#
# df |>
#   group_by(y, z) |>
#   mutate(mean_x = mean(x))
```