Module 2, Assignment 2

Ellen Bledsoe

2026-02-19

Assignment Details

Purpose

The goal of this assignment is to practice developing successful workflows to get from the starting data to a plot.

Task

Write R code which produces the correct answers and reflect on workflow.

Criteria for Success

- Code is within the provided code chunks
- Code is commented with brief descriptions of what the code does
- Code chunks run without errors
- Code produces the correct result
  - Code that produces the correct answer will receive full credit
  - Code attempts with logical direction will receive partial credit
- Written answers address the questions in sufficient detail

Due Date

Feb 26 before class

Assignment Questions

Our goal for this assignment is to start with a data frame of data, summarize the data in constructive ways, and plot the data to answer some questions. To get from Point A to Point D requires some planning.

In this assignment, we will first make a plan, execute how we would actually go about the process, and then evaluate how well our original plan matches with the path we actually used.

Data Summary

The aquaculture scientists on Team Antarctica have been working on developing a new diet for tilapia (a type of fish) based on soy-protein, and they are interested in whether incorporating this into fish diets will result in faster growth rates.

They are measuring the amount of growth based on the change in the weight of each fish after 30 days on the new diet.

They've provided us with the data below to analyze.

First, let's take a look at the data we will be using for the assignment.

1. As always, we start by loading the `tidyverse`. (1 point)

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.2
## v ggplot2   4.0.0     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become e
```

2. Next, we need to load in our tilapia growth dataset. Call the data frame `growth`. Then use both the `head` and `tail` functions to take a look at the data. (1 point)

```
growth <- read_csv("../data/tilapia_growth.csv")
```

```
## Rows: 320 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): soy_protein, tank_category
## dbl (4): tank_id, fish_id, avg_tank_temp, day_30_weight
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(growth)
```

```
## # A tibble: 6 x 6
##   tank_id fish_id soy_protein avg_tank_temp tank_category day_30_weight
##     <dbl>   <dbl> <chr>               <dbl> <chr>                 <dbl>
## 1       1       1 low                  77.2 warm                  334.
## 2       1       2 low                  77.2 warm                  198.
## 3       1       3 low                  77.2 warm                  315.
## 4       1       4 low                  77.2 warm                  316.
## 5       1       5 low                  77.2 warm                   89.4
## 6       1       6 low                  77.2 warm                   74.7
```

```
tail(growth)
```

```
## # A tibble: 6 x 6
##   tank_id fish_id soy_protein avg_tank_temp tank_category day_30_weight
##     <dbl>   <dbl> <chr>               <dbl> <chr>                 <dbl>
## 1      16     315 very high           76.1 warm                  1228.
## 2      16     316 very high           76.1 warm                   630.
## 3      16     317 very high           76.1 warm                   508.
## 4      16     318 very high           76.1 warm                   443.
## 5      16     319 very high           76.1 warm                   495.
## 6      16     320 very high           76.1 warm                  1078.
```

3. Before we can plan any strategy, we need to understand the data that we have. Answer the following questions about the data. (0.5 points each, 2 points total)

   a. What does one row represent? (One tank? One temperature? One treatment? One fish?)

   One fish

b. How many tanks of fish were sampled?

16

c. How many fish were sampled?

320

d. How many different soy protein levels (treatments) are there?

4

Our Task

We have been asked by our aquaculture specialists to provide them with the following:

a. a data frame with the average growth of fish per diet treatment (`soy_protein`); growth is being measured by the change in weight of each fish after 30 days on the diet (`day_30_weight`).
b. a plot that shows the relationship (or lack thereof) between the amount of soy protein in the diet and how much the fish have growth
c. a data frame with the average growth for each combination of diet treatment *and* tank category (warm or cold)
d. a plot that shows the relationship (or lack thereof) between average tank temperature, the amount of growth, and the amount of soy in the diet.

They've also asked that we provide all weights in kilograms instead of grams (the `day_30_weight` is currently in grams).

Prediction

4. Spend some time thinking about each one of these steps. What steps will you need to take to produce the end result? What data frame will you use? What columns will you use? What functions will you use? How will you plot things?

   For each of the 4 tasks listed above in "Our Task," *describe* (do NOT code) how you will get from the starting point (a data frame) to the result (another data frame or a plot).

   This question will be grade ***only on completion***, not on whether or not your plan is *correct*. (2 points)

   Task (a):

   Task (b):

   Task (c):

   Task (d):

Execution

Now let's actually go ahead and complete our tasks with code.

** NOTE! We need to run the line of code below. It's a little wonky but will ultimately be helpful. This line of code tells R that we want the soy proteins levels to be listed in a specific order (low to very high) rather than alphabetically. **

```
growth <- growth %>%
  mutate(soy_protein = factor(soy_protein, levels = c("low", "medium", "high", "very high")))
```

Because the aquaculture team has asked for everything to use kilograms instead of grams, it makes sense for us to add a column with the fish weights in kilograms (kg) instead of grams (g) to our data frame before we tackle any of the specific tasks.

5. Use the `mutate()` function to add a `day_30_weight_kg` column to the growth data frame with the weights in kilograms. (Hint: there are 1000 grams in 1 kilogram)

   Save the output as `growth_kg` (2 points)

   ** NOTE: This is the data frame you will use for the remainder of the assignment. **

```
growth_kg <- growth %>%
  mutate(day_30_weight_kg = day_30_weight / 1000)
growth_kg
```

```
## # A tibble: 320 x 7
##    tank_id fish_id soy_protein avg_tank_temp tank_category day_30_weight
##      <dbl>   <dbl> <fct>               <dbl> <chr>                 <dbl>
##  1       1       1 low                  77.2 warm                  334.
##  2       1       2 low                  77.2 warm                  198.
##  3       1       3 low                  77.2 warm                  315.
##  4       1       4 low                  77.2 warm                  316.
##  5       1       5 low                  77.2 warm                   89.4
##  6       1       6 low                  77.2 warm                   74.7
##  7       1       7 low                  77.2 warm                  142.
##  8       1       8 low                  77.2 warm                   20.8
##  9       1       9 low                  77.2 warm                   57.3
## 10       1      10 low                  77.2 warm                  159.
## # i 310 more rows
## # i 1 more variable: day_30_weight_kg <dbl>
```

Task (a)

A data frame with the average growth of fish per diet treatment

6. Create a new data frame called `growth_by_treatment` with the average growth for each level of soy protein. Remember to use the `growth_kg` data frame as your starting point. (2 points)

```
growth_by_treatment <- growth_kg %>%
  group_by(soy_protein) %>%
  summarize(mean_weight_kg = mean(day_30_weight_kg))
growth_by_treatment
```

```
## # A tibble: 4 x 2
##   soy_protein mean_weight_kg
##   <fct>                <dbl>
## 1 low                  0.276
## 2 medium               0.441
## 3 high                 0.618
## 4 very high            0.829
```

Task (b)

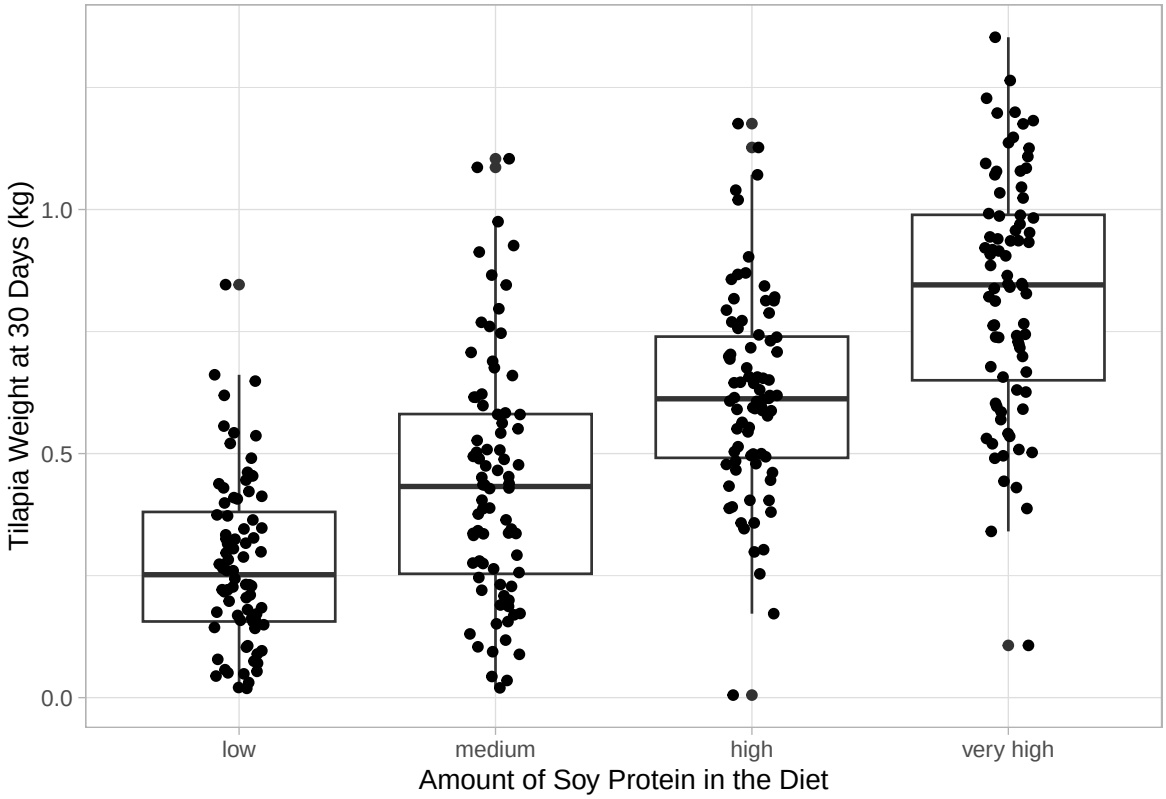
A box plot that shows the relationship (or lack thereof) between the amount of soy protein in the diet and the weight at 30 days (growth)

Based on the values we calculated in task (a), it looks like there is probably a positive relationship between the percent of soy protein in tilapia diet and growth. Let's plot the data to confirm.

7. Make a box plot to show this relationship. Change the axis labels to be more easily understood and add a theme. (2 point)

(Hint: because we want to plot *all* of the values, not just the mean values, we need to use the original `growth_kg` data frame, not the data frame we created in task (a))

```
ggplot(growth_kg, aes(soy_protein, day_30_weight_kg)) +
  geom_boxplot() +
  geom_jitter(width = 0.1) +
  labs(x = "Amount of Soy Protein in the Diet",
       y = "Tilapia Weight at 30 Days (kg)") +
  theme_light()
```



Task (c)

A data frame with the the average growth for each combination of diet treatment *and* tank category

8. Create a new data frame called `growth_by_temp_treatment`. It should have groups for each combination of the amount of soy protein and whether tanks are warm or cold. Remember, you can create groups with multiple columns! Calculate the average growth for each combination. (2 points)

```
growth_by_treatment_and_temp <- growth_kg %>%
  group_by(soy_protein, tank_category) %>%
  summarise(mean_weight_kg = mean(day_30_weight_kg))
```

```
## `summarise()` has grouped output by 'soy_protein'. You can override using the
## `.groups` argument.
```

```
growth_by_treatment_and_temp
```

```
## # A tibble: 8 x 3
## # Groups:   soy_protein [4]
```

```
##   soy_protein tank_category mean_weight_kg
##   <fct>       <chr>                  <dbl>
## 1 low         cold                   0.279
## 2 low         warm                   0.272
## 3 medium      cold                   0.420
## 4 medium      warm                   0.448
## 5 high        cold                   0.617
## 6 high        warm                   0.620
## 7 very high   cold                   0.824
## 8 very high   warm                   0.833
```

Task (d)

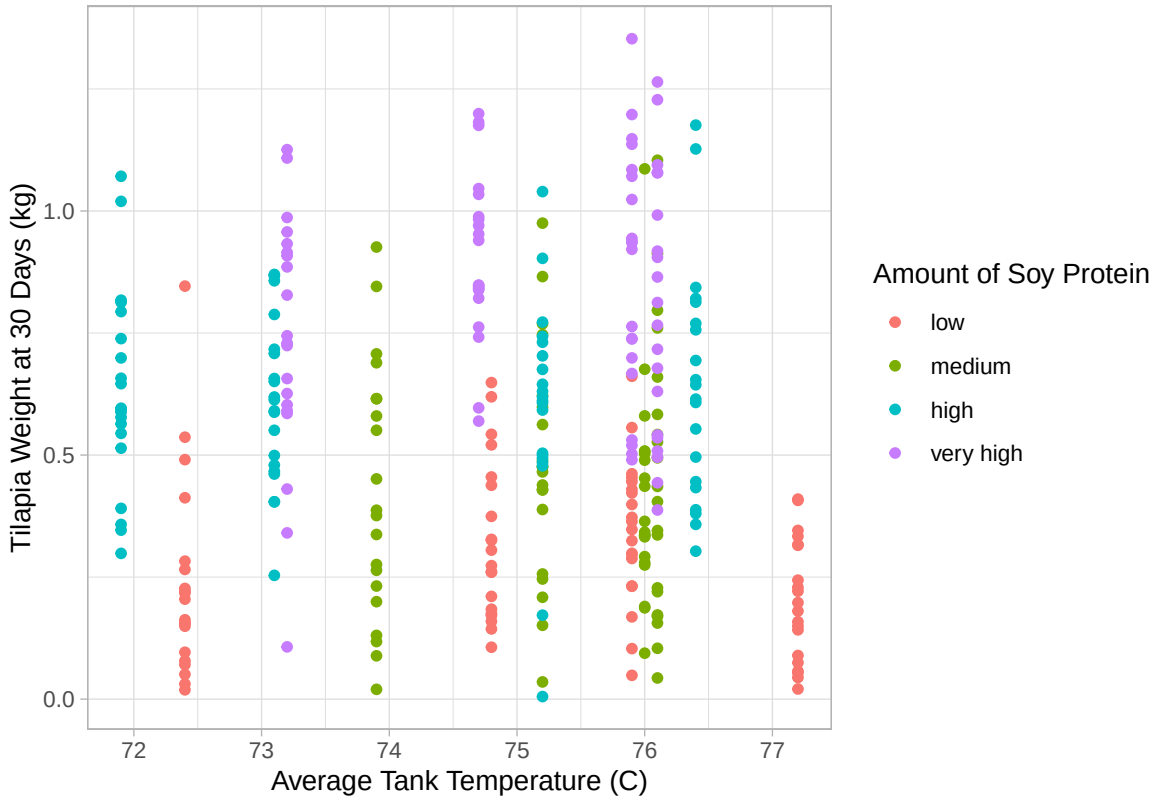
A scatter plot that show the relationship (or lack thereof) between average tank temperature and the weight at 30 days

Based on our results above, do we think the tank being warm or cold has much of an influence? Let's plot our data to confirm.

9. Make a multiple scatter plot with the average water temperature in the tank on the x-axis (the actual number, not whether the tank in "cold" or "warm") and the amount of growth on the y-axis (vertical). Change the color of each point so that they represent the amount of soy protein in the diet. Change the axis labels to be more easily understood and add a theme. (2 point)

   (Hint: because we want to plot *all* of the values, not just the mean values, we need to use the original `growth_kg` data frame)

```
ggplot(growth_kg, aes(avg_tank_temp, day_30_weight_kg, color = soy_protein)) +
  geom_point() +
  labs(x = "Average Tank Temperature (C)",
       y = "Tilapia Weight at 30 Days (kg)",
       color = "Amount of Soy Protein") +
  theme_light()
```



As suspected, there doesn't seem to be much of a difference based on tank temperatures.

Reflection

10. Imagine we had decided to treat `soy_protein` as a continuous variable. What type of plot we have used for task (b)? Would we have been able to complete task (d)? Why or why not? (2 points)

    Answer: Yes or no is acceptable. I have not taught them how to plot 3 continuous variables on a scatter plot, but it is doable. If they say no, you can only plot 2 continuous and 1 categorical, that is valid. If they say you *can* plot 3 because you can use color as a spectrum, that is also viable. Just make sure they explain why it would (or would not) work.

11. Write 3-5 sentences about if and how your predictions and execution differed and what you learned through the process. (3 points)

    Examples of questions to answer: How did your initial prediction of how you expected to accomplish the 4 tasks match up with how we actually went about doing it? Were they similar? Were there common mistakes that you made beforehand? Did you plan a different execution from what we did above that you think would also work?

    Answer: