DataWrangling_Challenge5

Temitope Ruth Folorunso and Nneka Vivian Iduu

2025-03-20

Contents

This assignment will help you practice integrating some of the tidyverse functions into your R scripts. It will also involve some more practice with GitHub. You may collaborate with a partner to enhance your learning experience. Please ensure the following:

Question 1

3pts. Download two .csv files from Canvas called DiversityData.csv and Metadata.csv, and read them into R using relative file paths.

Question 2

4pts. Join the two dataframes together by the common column 'Code'. Name the resulting dataframe alpha

```
alpha <- left_join(Metadata, DiversityData, by = "Code")
head(alpha)
```

```
##     Code Crop Time_Point Replicate Water_Imbibed  shannon invsimpson   simpson
## 1 S01_13 Soil          0         1            NA 6.624921   210.7279 0.9952545
## 2 S02_16 Soil          0         2            NA 6.612413   206.8666 0.9951660
## 3 S03_19 Soil          0         3            NA 6.660853   213.0184 0.9953056
## 4 S04_22 Soil          0         4            NA 6.660671   204.6908 0.9951146
## 5 S05_25 Soil          0         5            NA 6.610965   200.2552 0.9950064
## 6 S06_28 Soil          0         6            NA 6.650812   199.3211 0.9949830
##   richness
## 1     3319
## 2     3079
## 3     3935
## 4     3922
## 5     3196
## 6     3481
```

Question 3

4 pts. Calculate Pielou's evenness index: Pielou's evenness is an ecological parameter calculated by the Shannon diversity index (column Shannon) divided by the log of the richness column. a. Using mutate, create a new column to calculate apl. b. Name the resulting dataframe alpha_even.

```
 # Create a new column called logRich
alpha_log <- mutate(alpha, logRich = log(richness)) # create new column named logRich that has the log
alpha_even <- mutate(alpha_log, eveness_index = shannon / logRich) # create new column named Pielou's e
```

Question 4

4pts. Using tidyverse language of functions and the pipe, use the summarise function and tell me the mean and standard error evenness grouped by crop over time. a. Start with the alpha_even dataframe b. Group the data: group the data by Crop and Time_Point. c. Summarize the data: Calculate the mean, count, standard deviation, and standard error for the even variable within each group. d. Name the resulting dataframe alpha_average

```
alpha_average <- alpha_even %>%
  group_by(Crop, Time_Point) %>%
  summarise(mean.even =mean(eveness_index),
            n= n(),
            sd.dev =sd(eveness_index)) %>%
            mutate(std.err =sd.dev/sqrt(n))
```

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

Question 5

4. Pts. Calculate the difference between the soybean column, the soil column, and the difference between the cotton column and the soil column

a. Start with the alpha_average dataframe
b. Select relevant columns: select the columns Time_Point, Crop, and mean.even.
c. Reshape the data: Use the pivot_wider function to transform the data from long to wide format, creating new columns for each Crop with values from mean.even.
d. Calculate differences: Create new columns named diff.cotton.even and diff.soybean.even by calculating the difference between Soil and Cotton, and Soil and Soybean, respectively.
e. Name the resulting dataframe alpha_average2

```
alpha_average2 <- alpha_average %>%
  select(Time_Point, Crop, mean.even) %>%
  pivot_wider(names_from = Crop, values_from = mean.even) %>%
  mutate(diff.cotton.even = Soil - Cotton, diff.soybean.even = Soil - Soybean )
```

Question 6

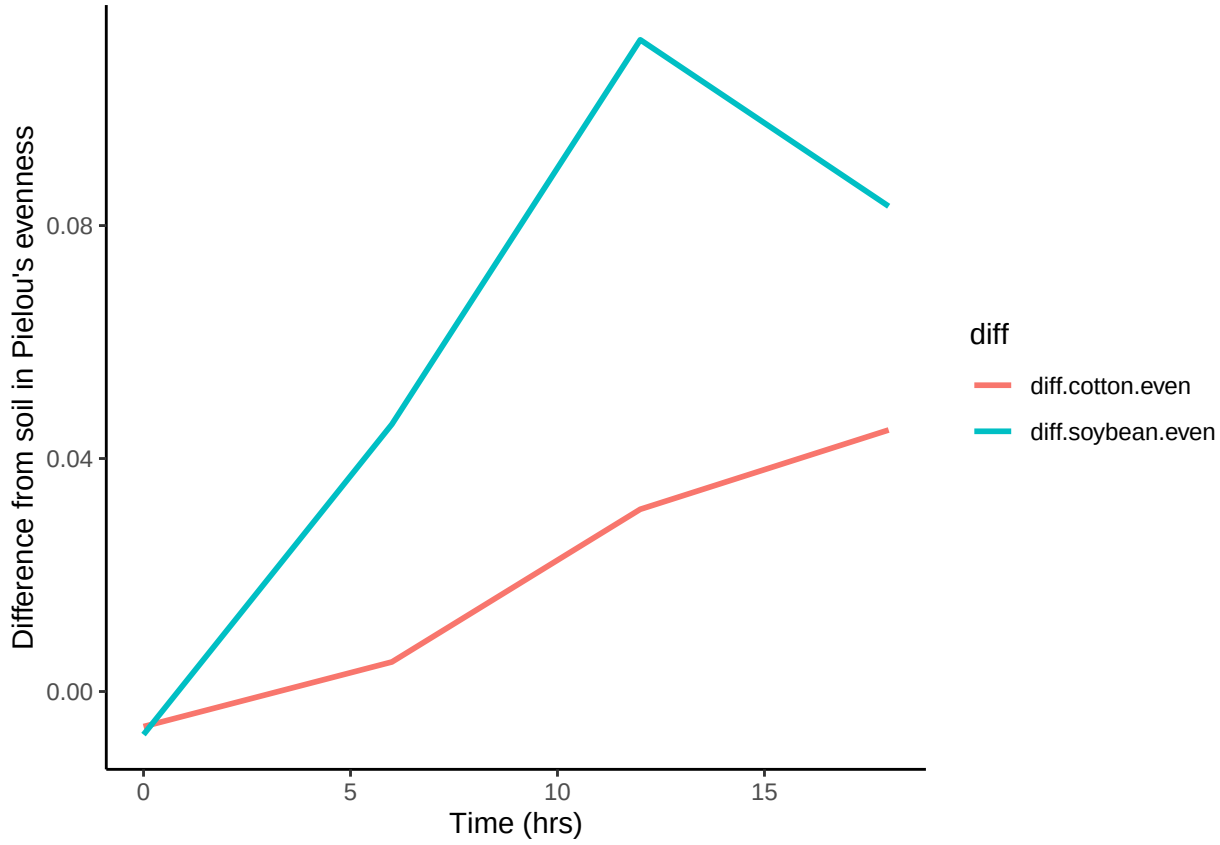
4 pts. Connecting it to plots a. Start with the alpha_average2 dataframe b. Select relevant columns: select the columns Time_Point, diff.cotton.even, and diff.soybean.even. c. Reshape the data: Use the pivot_longer function to transform the data from wide to long format, creating a new column named diff that contains the values from diff.cotton.even and diff.soybean.even. i. This might be challenging, so I'll give you a break. The code is below.

pivot_longer(c(diff.cotton.even, diff.soybean.even), names_to = "diff")

d. Create the plot: Use ggplot and geom_line() with 'Time_Point' on the x-axis, the column 'values' on the y-axis, and different colors for each 'diff' category. The column named 'values' come from the pivot_longer. The resulting plot should look like the one to the right.

```
alpha_average2 %>%
select(Time_Point, diff.cotton.even, diff.soybean.even) %>%
pivot_longer(c(diff.cotton.even, diff.soybean.even), names_to = "diff") %>%
ggplot(aes(x = Time_Point, y = value, color = diff)) +
geom_line(size = 1) +
labs(x = "Time (hrs)", y = "Difference from soil in Pielou's evenness") +
theme_classic()
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



Question 7

2 pts. Commit and push a gfm .md file to GitHub inside a directory called Coding Challenge 5. Provide me a link to your github written as a clickable link in your .pdf or .docx

https://github.com/NVI0001/Reproducibility_Project/tree/main/CodingChallenge5_files