Assignment 5: Data Visualization

Janice Ye

Spring 2024

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Visualization

Directions

1. Rename this file `<FirstLast>_A05_DataVisualization.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure your code is tidy; use line breaks to ensure your code fits in the knitted output.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up your session

1. Set up your session. Load the tidyverse, lubridate, here & cowplot packages, and verify your home directory. Read in the NTL-LTER processed data files for nutrients and chemistry/physics for Peter and Paul Lakes (use the tidy `NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv` version in the Processed_KEY folder) and the processed data file for the Niwot Ridge litter dataset (use the `NEON_NIWO_Litter_mass_trap_Processed.csv` version, again from the Processed_KEY folder).

2. Make sure R is reading dates as date format; if not change the format to date.

```
#1
#load packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- 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 errors
```

```
library(lubridate)
library(here)
```

```
## here() starts at /home/guest/EDA_Spring2024
```

```
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```
#verify directory
getwd()
```

```
## [1] "/home/guest/EDA_Spring2024"
```

```
#read files
NTL <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
                stringsAsFactors = TRUE)
NIWO <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
                 stringsAsFactors = TRUE)

#2
NTL$sampledate <- as.Date(NTL$sampledate)
NIWO$collectDate <- as.Date(NIWO$collectDate)
```

Define your theme

3. Build a theme and set it as your default theme. Customize the look of at least two of the following:

- Plot background
- Plot title
- Axis labels
- Axis ticks/gridlines
- Legend

```
#3
theme <- theme_light() +
  theme(
    axis.text = element_text(color = "black"),
    legend.position = "top"
  )
```

Create graphs

For numbers 4-7, create ggplot graphs and adjust aesthetics to follow best practices for data visualization. Ensure your theme, color palettes, axes, and additional aesthetics are edited accordingly.

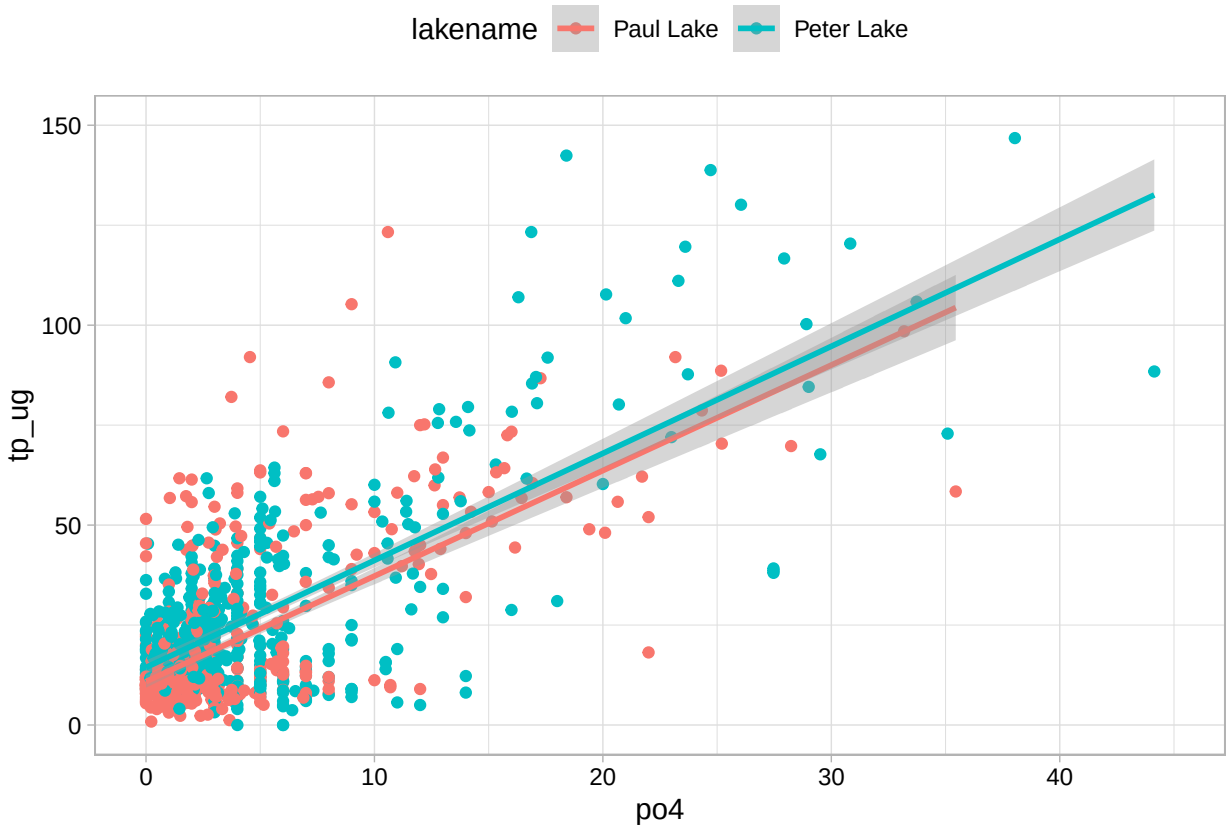
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add line(s) of best fit using the `lm` method. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```
#4
tp_ugplot <-
  ggplot(NTL, aes(x = po4, y = tp_ug, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm) +
  xlim(0, 45) +
  ylim(0, 150) +
```

```
  theme

print(tp_ugplot)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 21948 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 21948 rows containing missing values (`geom_point()`).
```



5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

Tips: * Recall the discussion on factors in the lab section as it may be helpful here. * Setting an axis title in your theme to `element_blank()` removes the axis title (useful when multiple, aligned plots use the same axis values) * Setting a legend's position to "none" will remove the legend from a plot. * Individual plots can have different sizes when combined using `cowplot`.

```
#5
aplot <-
  ggplot(NTL, aes(x = month, y = temperature_C, color = lakename)) +
  geom_boxplot() +
  theme(axis.title.x = element_blank(),
              legend.position = "none")

bplot <-
  ggplot(NTL, aes(x = month, y = tp_ug, color = lakename)) +
```

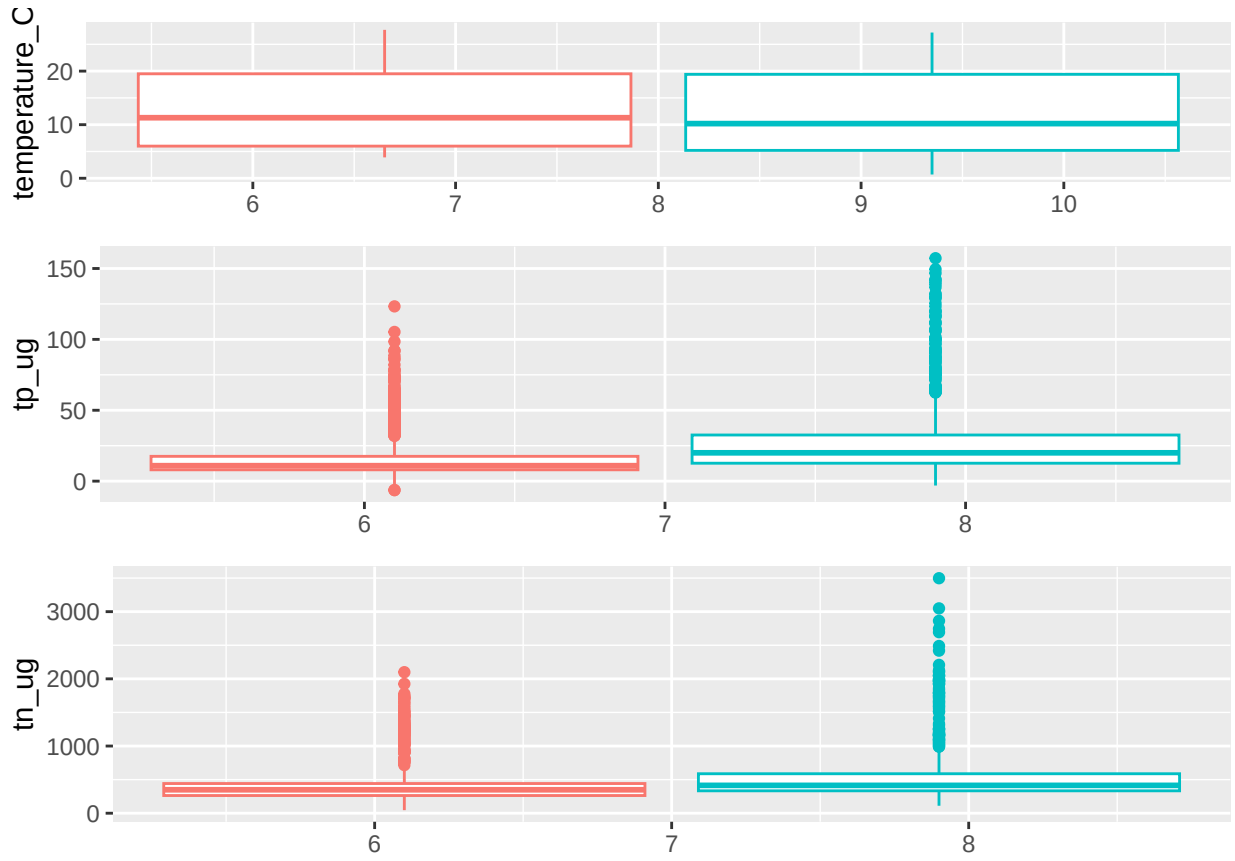
```
  geom_boxplot() +
  theme(axis.title.x = element_blank(),
              legend.position = "none")

cplot <-
  ggplot(NTL, aes(x = month, y = tn_ug, color = lakename)) +
  geom_boxplot() +
  theme(axis.title.x = element_blank(),
              legend.position = "none")

allplot <- plot_grid(aplot, bplot, cplot, nrow = 3, align = 'h', rel_heights = c(0.7, 1, 1))
```

```
## Warning: Removed 3566 rows containing non-finite values (`stat_boxplot()`).
## Warning: Removed 20729 rows containing non-finite values (`stat_boxplot()`).
## Warning: Removed 21583 rows containing non-finite values (`stat_boxplot()`).
```

```
print(allplot)
```



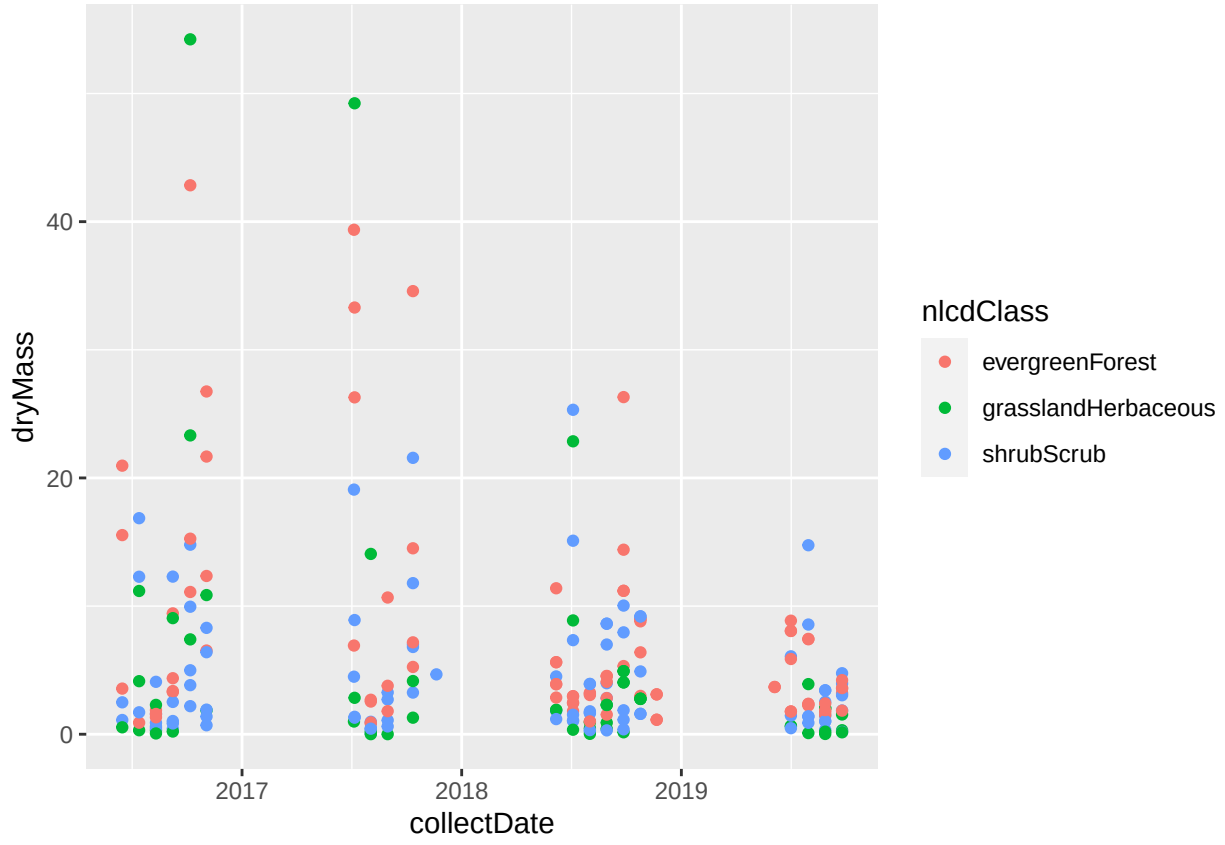
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Peter Lake tends to have greater amount of nutrients than Paul Lake, and Peter Lake is more active in dates later than Paul Lake.

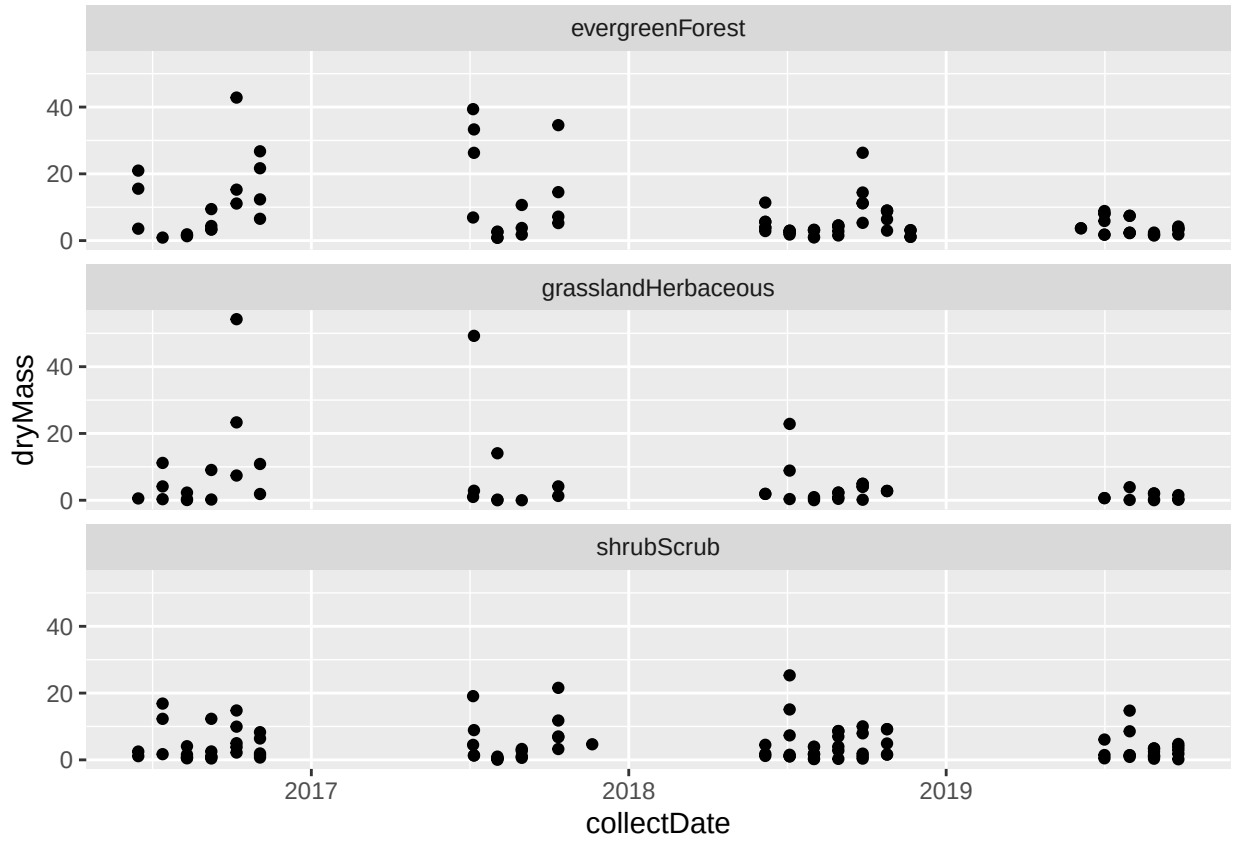
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
#6
Niwot1 <-
  ggplot(subset(NIWO, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point()
print(Niwot1)
```




```
#7
Niwot2 <-
  ggplot(subset(NIWO, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass)) +
  geom_point() +
  facet_wrap(vars(nlcdClass), nrow = 3)
print(Niwot2)
```



Question: Which of these plots (6 vs. 7) do you think is more effective, and why?

Answer: I think plot 7 is mor effective because it separates the data based on additional layer. It shows the distribution by year and by nlcdClass whereas plot 6 crumples up all three nlcdClass together.