Assignment 5: Data Visualization

Hanna Karnei

Fall 2023

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.

```r
#1

# Upload packages

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(here)
```

```
## here() starts at /Users/hannakarnei/Desktop/EDA/EDE_Fall2023
```

```r
library(ggplot2)
library(cowplot)
```

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

```r
# Check working directory
getwd()
```

```
## [1] "/Users/hannakarnei/Desktop/EDA/EDE_Fall2023"
```

```r
# Load data
processed_data = "Data/Processed_KEY"

PeterPaul.chem.nutrients <- read.csv(
  here(processed_data,"NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"),
  stringsAsFactors = TRUE)

head(PeterPaul.chem.nutrients)
```

```
##    lakename year4 daynum month sampledate depth temperature_C dissolvedOxygen
## 1 Paul Lake  1984    148     5 1984-05-27  0.00          14.5             9.5
## 2 Paul Lake  1984    148     5 1984-05-27  0.25            NA              NA
## 3 Paul Lake  1984    148     5 1984-05-27  0.50            NA              NA
## 4 Paul Lake  1984    148     5 1984-05-27  0.75            NA              NA
## 5 Paul Lake  1984    148     5 1984-05-27  1.00          14.5             8.8
## 6 Paul Lake  1984    148     5 1984-05-27  1.50            NA              NA
##   irradianceWater irradianceDeck tn_ug tp_ug nh34 no23 po4
## 1            1750           1620    NA    NA   NA   NA  NA
## 2            1550           1620    NA    NA   NA   NA  NA
## 3            1150           1620    NA    NA   NA   NA  NA
## 4             975           1620    NA    NA   NA   NA  NA
## 5             870           1620    NA    NA   NA   NA  NA
## 6             610           1620    NA    NA   NA   NA  NA
```

```r
Niwot.litter<- read.csv(
  here(processed_data,"NEON_NIWO_Litter_mass_trap_Processed.csv"),
  stringsAsFactors = TRUE)

head(Niwot.litter)
```

```
##     plotID       trapID collectDate functionalGroup dryMass qaDryMass subplotID
## 1 NIWO_062 NIWO_062_050  2016-06-16           Seeds    0.00         N        31
## 2 NIWO_061 NIWO_061_169  2016-06-16           Other    0.27         N        41
## 3 NIWO_062 NIWO_062_050  2016-06-16  Woody material    0.12         N        31
## 4 NIWO_064 NIWO_064_103  2016-06-16           Seeds    0.00         N        32
## 5 NIWO_058 NIWO_058_101  2016-06-16         Needles    1.11         Y        32
## 6 NIWO_058 NIWO_058_101  2016-06-16          Leaves    0.00         N        32
##   decimalLatitude decimalLongitude elevation       nlcdClass plotType
## 1        40.05114        -105.5858    3477.0      shrubScrub    tower
## 2        40.04762        -105.5861    3413.4 evergreenForest    tower
## 3        40.05114        -105.5858    3477.0      shrubScrub    tower
## 4        40.04737        -105.5840    3373.2 evergreenForest    tower
## 5        40.04872        -105.5872    3446.4      shrubScrub    tower
## 6        40.04872        -105.5872    3446.4      shrubScrub    tower
##   geodeticDatum
## 1         WGS84
## 2         WGS84
## 3         WGS84
## 4         WGS84
## 5         WGS84
## 6         WGS84
```

```r
#2 Fix date format

class(PeterPaul.chem.nutrients$sampledate)
```

```
## [1] "factor"
```

```r
class(Niwot.litter$collectDate)
```

```
## [1] "factor"
```

```r
PeterPaul.chem.nutrients$sampledate <- ymd(PeterPaul.chem.nutrients$sampledate)
Niwot.litter$collectDate <-ymd(Niwot.litter$collectDate)

class(PeterPaul.chem.nutrients$sampledate)
```

```
## [1] "Date"
```

```r
class(Niwot.litter$collectDate)
```

```
## [1] "Date"
```

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 Create a custom theme

my_theme <- theme_classic(base_size = 10) +
  theme(
    legend.background = element_rect(
      color='red',
      fill = 'grey'),
    legend.position = "bottom",
    plot.title = element_text(
      color='red' )
  )
```

Create graphs

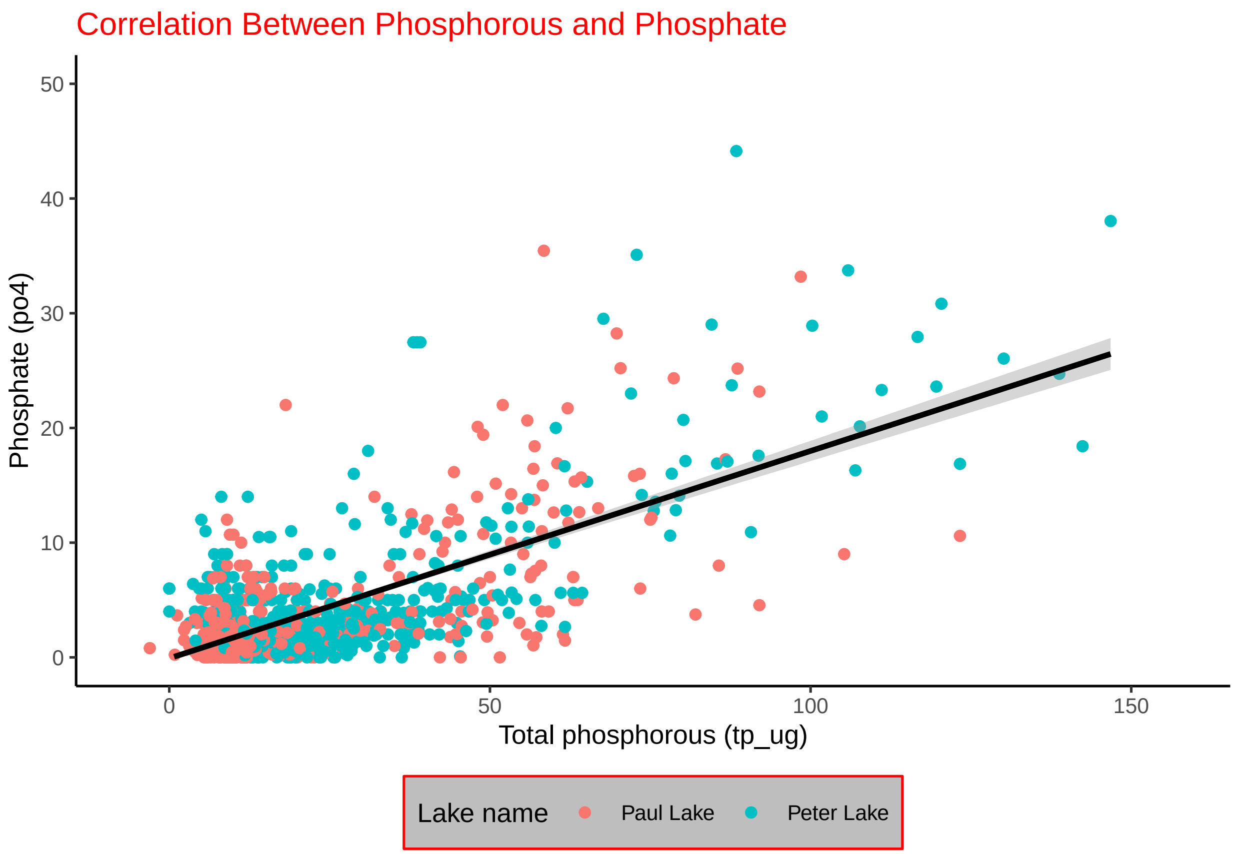
For numbers 4-7, create ggplot graphs and adjust aesthetics to follow best practices for data visualization. Ensure your theme, color palettes, axes, and additional aesthetics are edited accordingly.

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

```
#4 Plot total phosphorous by phosphate

p_po4_plot <- ggplot(PeterPaul.chem.nutrients, aes(x = tp_ug, y = po4, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, color='black') +
  ylim(0,50) +
  xlab('Total phosphorous (tp_ug)') +
  ylab('Phosphate (po4)') +
  ggtitle ('Correlation Between Phosphorous and Phosphate') +
  labs(colour="Lake name") +
  my_theme

print(p_po4_plot)
```

```
## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 21947 rows containing non-finite values (stat_smooth).

## Warning: Removed 21947 rows containing missing values (geom_point).
```

```
## Warning: Removed 2 rows containing missing values (geom_smooth).
```



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.

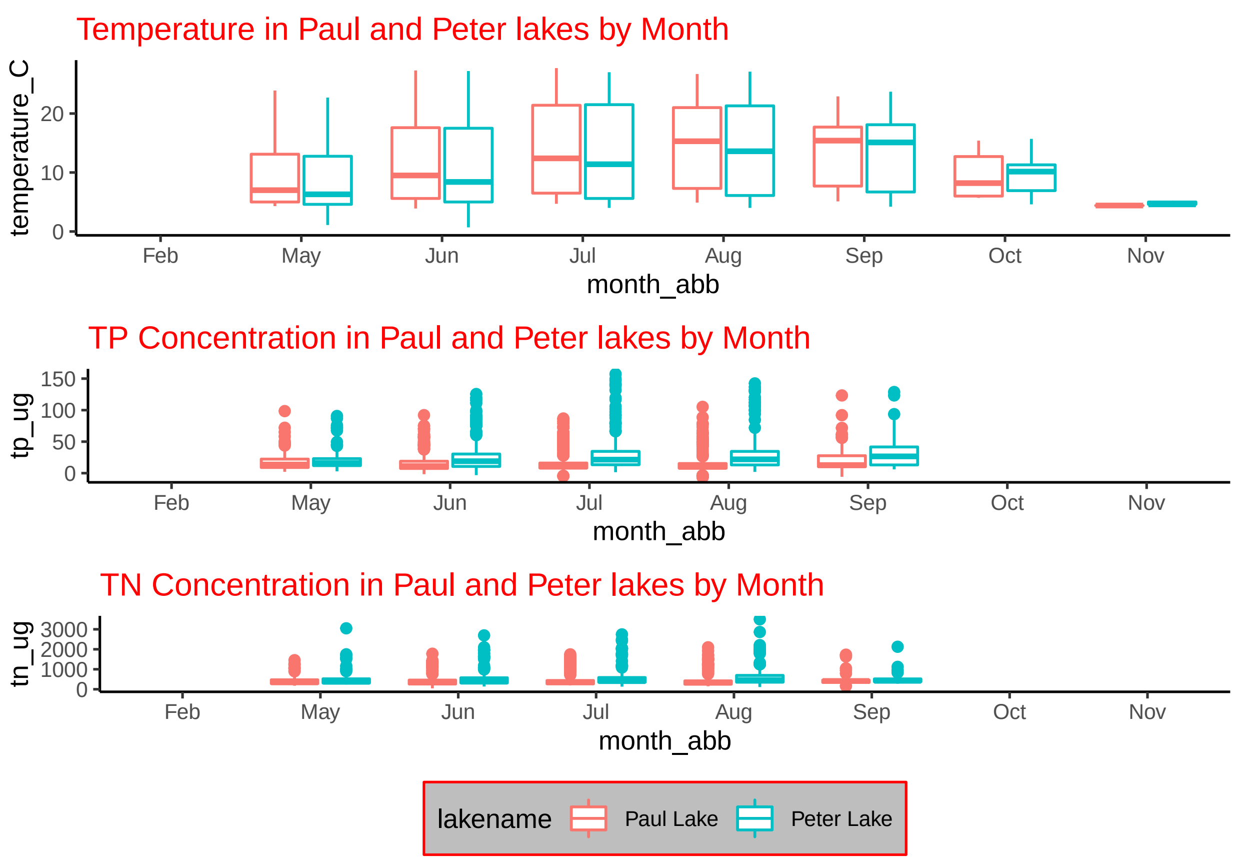
Tip: * Recall the discussion on factors in the previous section as it may be helpful here. * R has a built-in variable called `month.abb` that returns a list of months; see https://r-lang.com/month-abb-in-r-with-example

```
#5 Create boxplots

PeterPaul.chem.nutrients$month_abb <- month.abb[PeterPaul.chem.nutrients$month]

PeterPaul.chem.nutrients$month_abb <- factor(PeterPaul.chem.nutrients$month_abb, levels = month.abb)

temp.plot <- ggplot(PeterPaul.chem.nutrients, aes(x = month_abb, y = temperature_C, color = lakename)) +
  geom_boxplot() +
  ggtitle ('Temperature in Paul and Peter lakes by Month') +
  my_theme +
  theme(legend.position = "none")

tp.plot <- ggplot(PeterPaul.chem.nutrients, aes(x = month_abb, y = tp_ug, color = lakename)) +
  geom_boxplot() +
  ggtitle ('TP Concentration in Paul and Peter lakes by Month') +
```

```
  my_theme +
  theme(legend.position = "none")

tn.plot <- ggplot(PeterPaul.chem.nutrients, aes(x = month_abb, y = tn_ug, color = lakename)) +
  geom_boxplot() +
  ggtitle ('TN Concentration in Paul and Peter lakes by Month') +
  my_theme

all_graphs <- plot_grid(temp.plot, tp.plot, tn.plot,
                        nrow = 3, align = 'h', axis = 'x',
                        rel_heights = c(1.25, 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(all_graphs)
```



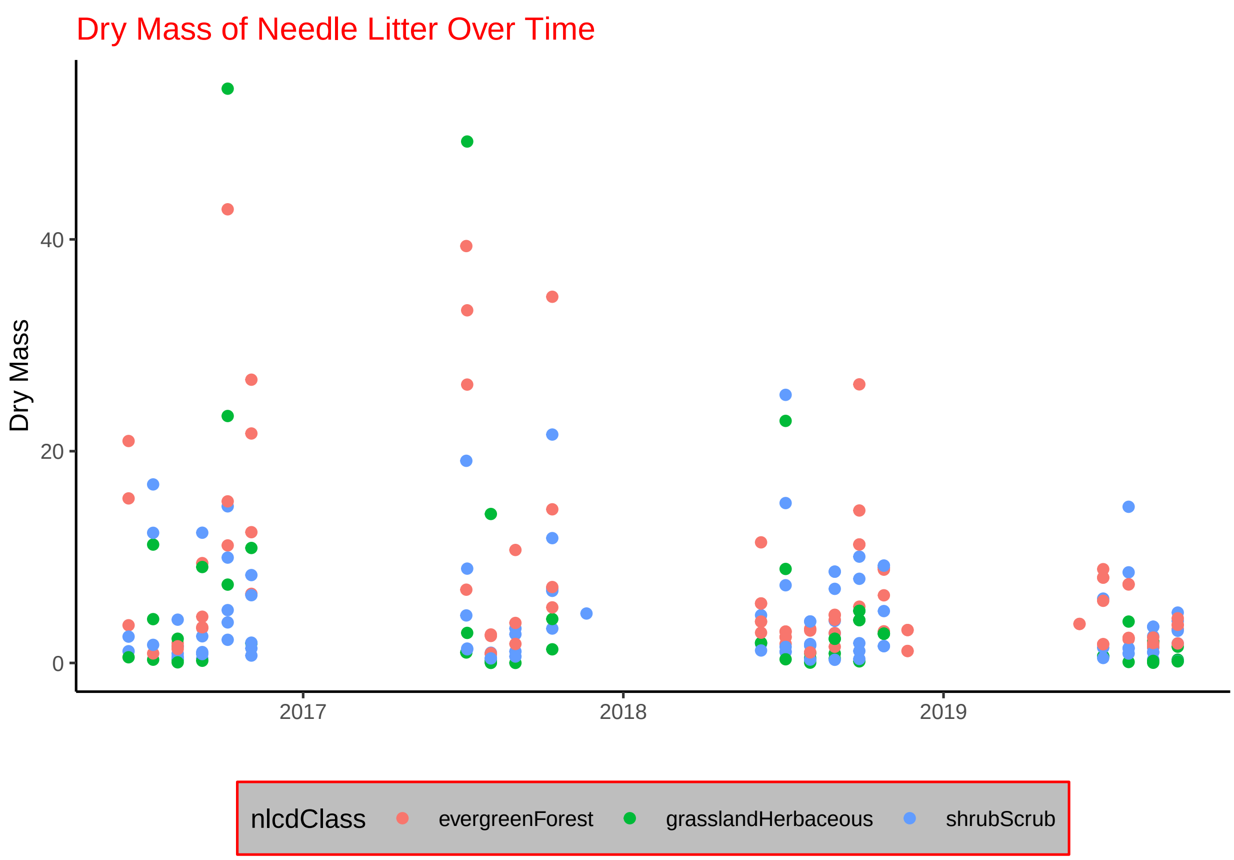
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: On average, values for all three variables tend to be lower for Paul lake – the trend is clear in the TP and TN data, and less pronounced in the temperature data. In addition, data for Peter lake is more widely spread. As for temporal differences, data appears to be higher in the summer months for the temperature and TP graphs.

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)

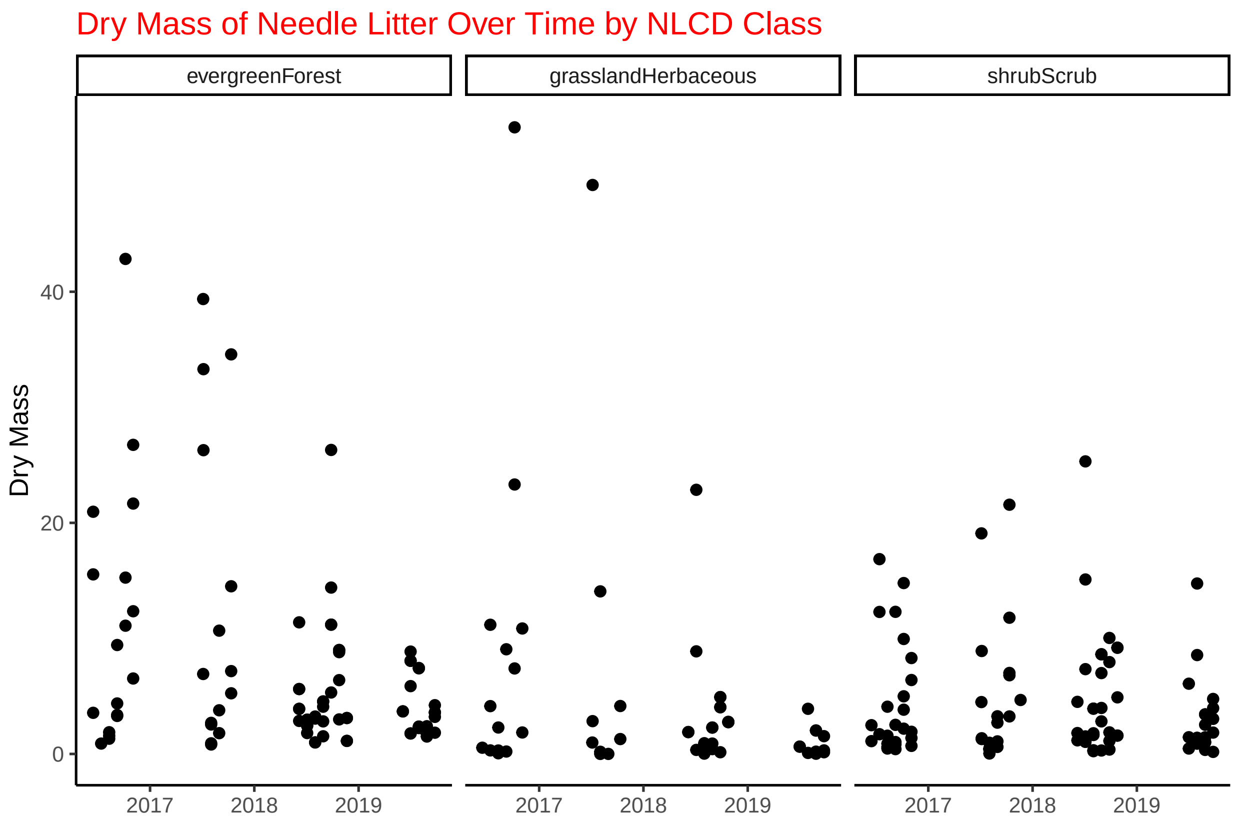
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
#6 Plot the needle plot

needles <- ggplot(subset(Niwot.litter, functionalGroup == "Needles"),
                  aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point() +
  ggtitle('Dry Mass of Needle Litter Over Time') +
  xlab('') +
  ylab('Dry Mass') +
  my_theme

print(needles)
```




```
#7 Plot the needles plot with faucets

needles_facets <- ggplot(subset(Niwot.litter, functionalGroup == "Needles"),
                         aes(x = collectDate, y = dryMass)) +
  geom_point() +
  facet_wrap(~ nlcdClass) +
  ggtitle('Dry Mass of Needle Litter Over Time by NLCD Class') +
```

```
  xlab('') +
  ylab('Dry Mass') +
  my_theme

print(needles_facets)
```



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

> Answer: Plot No. 7 is better-suited graph because it communicates the diffirence in the distribution of data for each classs more effectively. Although it does not have color, we are able to quickly identify trends in the dry mass distribution over time and compare them across classes thanks to the facets. In Plot No. 6, in contrast, data points are clustered together, making any comparisons between classes challenging.