

# Assignment 5: Data Visualization

Cammie Moore

Spring 2025

## 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
```

```
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   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## 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(cowplot)
```

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

```
library(here)
```

```
## Warning: package 'here' was built under R version 4.4.2
```

```
## here() starts at C:/Users/cammi/OneDrive/Documents/EDE_Spring2025
```

```
library(extrafont)
```

```
## Registering fonts with R
```

```
getwd()
```

```
## [1] "C:/Users/cammi/OneDrive/Documents/EDE_Spring2025"
```

```
here()
```

```
## [1] "C:/Users/cammi/OneDrive/Documents/EDE_Spring2025"
```

```
peterpaul <- read.csv(here("Data", "Processed_KEY", "NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"))
```

```
litter <- read.csv(here("Data", "Processed_KEY", "NEON_NIWO_Litter_mass_trap_Processed.csv"))
```

```
#2
```

```
peterpaul$sampldate <- as.Date(peterpaul$sampldate)
```

```
litter$collectDate <- as.Date(litter$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

```
mytheme <-  
  theme_classic()+  
  theme(  
    plot.background = element_rect(  
      fill = "white"  
    ),  
    plot.title = element_text(  
      color = "black",  
      size = 16),  
    axis.text = element_text(  
      color = "black",  
      size = 12  
    ),  
    panel.background = element_rect(  
      color = "azure2",  
      fill = "azure2"  
    )  
  )
```

## 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

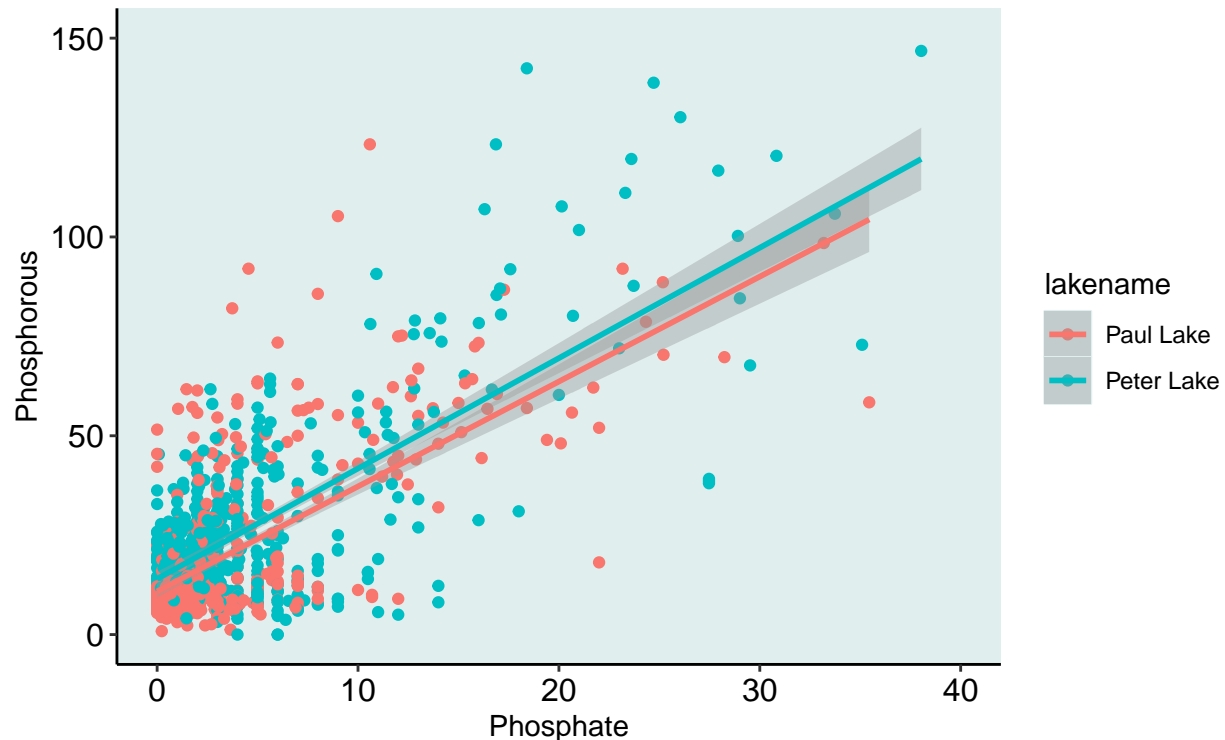
```
phosphorus_phosphate_plot <- peterpaul |>  
  ggplot(aes(x = po4, y = tp_ug))+  
  geom_point(aes(colour = lakenam))+  
  labs(  
    title = "Positive Correlation between Phosphorus and \nPhosphate in Peter and Paul Lakes",  
    x = "Phosphate",  
    y = "Phosphorous")+  
  xlim(0, 40)+  
  ylim(0, 150)+  
  geom_smooth(method = lm,  
    aes(group = lakenam, color = lakenam))+  
  mytheme  
  
print(phosphorus_phosphate_plot)
```

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

```
## Warning: Removed 21949 rows containing non-finite outside the scale range  
## ('stat_smooth()').
```

```
## Warning: Removed 21949 rows containing missing values or values outside the scale range  
## ('geom_point()').
```

## Positive Correlation between Phosphorus and Phosphate in Peter and Paul Lakes



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
box_temp <- peterpaul |>
  ggplot(aes(x=factor(month, levels = 5:11, labels = month.abb[5:11]),
             y = temperature_C,
             color = lakename))+
  geom_boxplot()+
  labs(title = "Temperature (C) over Time",
       x = "Month",
       y = "Temparture (C)")+
  scale_x_discrete(drop=F, name = "Month")+
  mytheme+
  theme(axis.title.x = element_blank(),
        legend.position = "none")

box_tp <- peterpaul |>
  ggplot(aes(x=factor(month, levels=5:11, labels = month.abb[5:11]),
```

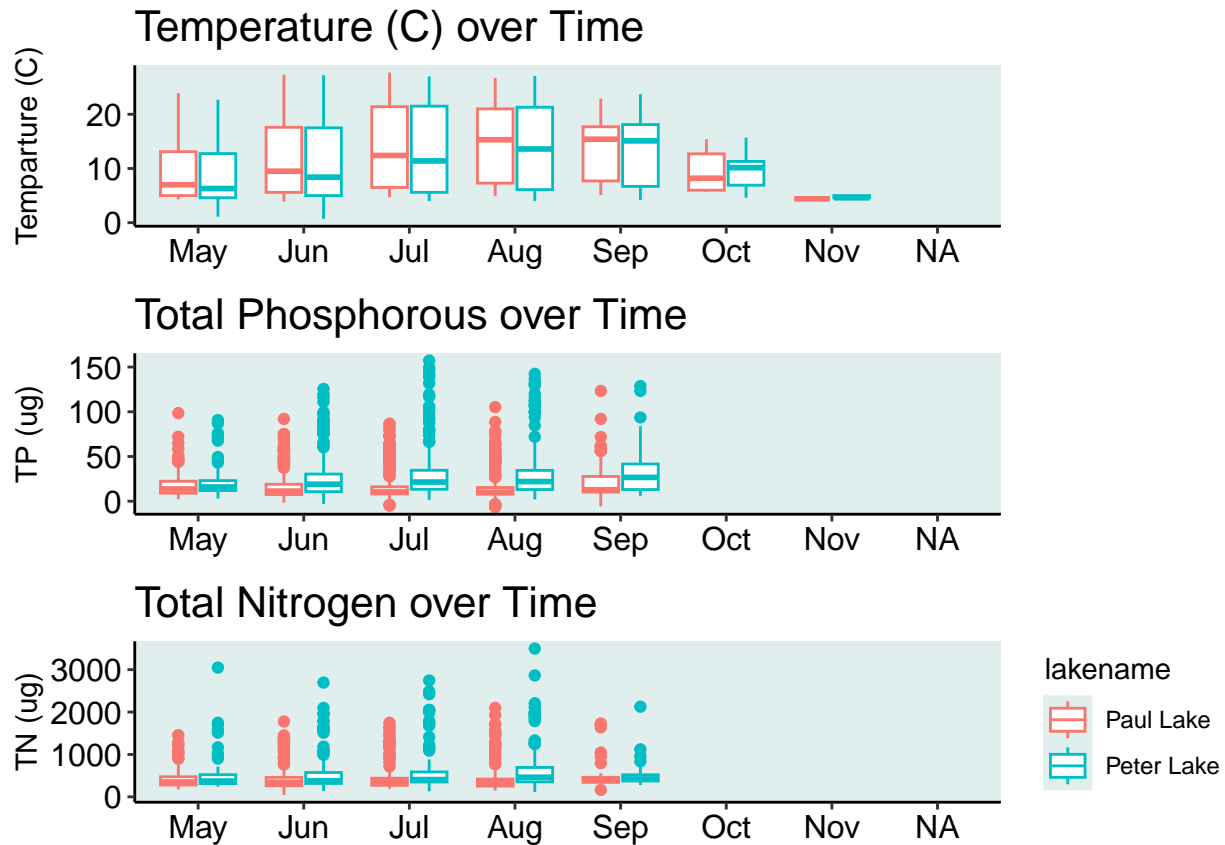
```

        y = tp_ug,
        color = lakename)))+
geom_boxplot()+
labs(title = "Total Phosphorous over Time",
      x = "Month",
      y = "TP (ug)")+
scale_x_discrete(drop=F, name = "Month")+
mytheme+
theme(axis.title.x = element_blank(),
      legend.position = "none")

box_tn <- peterpaul |>
ggplot(aes(x=factor(month, levels = 5:11, labels = month.abb[5:11]),
          y = tn_ug,
          color = lakename))+
geom_boxplot()+
labs(title = "Total Nitrogen over Time",
      x = "Month",
      y = "TN (ug)")+
scale_x_discrete(drop=F, name = "Month")+
mytheme+
theme(axis.title.x = element_blank())

plot_grid(
  box_temp + theme(legend.position = "none"),
  box_tp + theme(legend.position = "none"),
  box_tn,
  ncol = 1,
  align = "v",
  rel_heights = c(1, 1, 1)
)

```



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

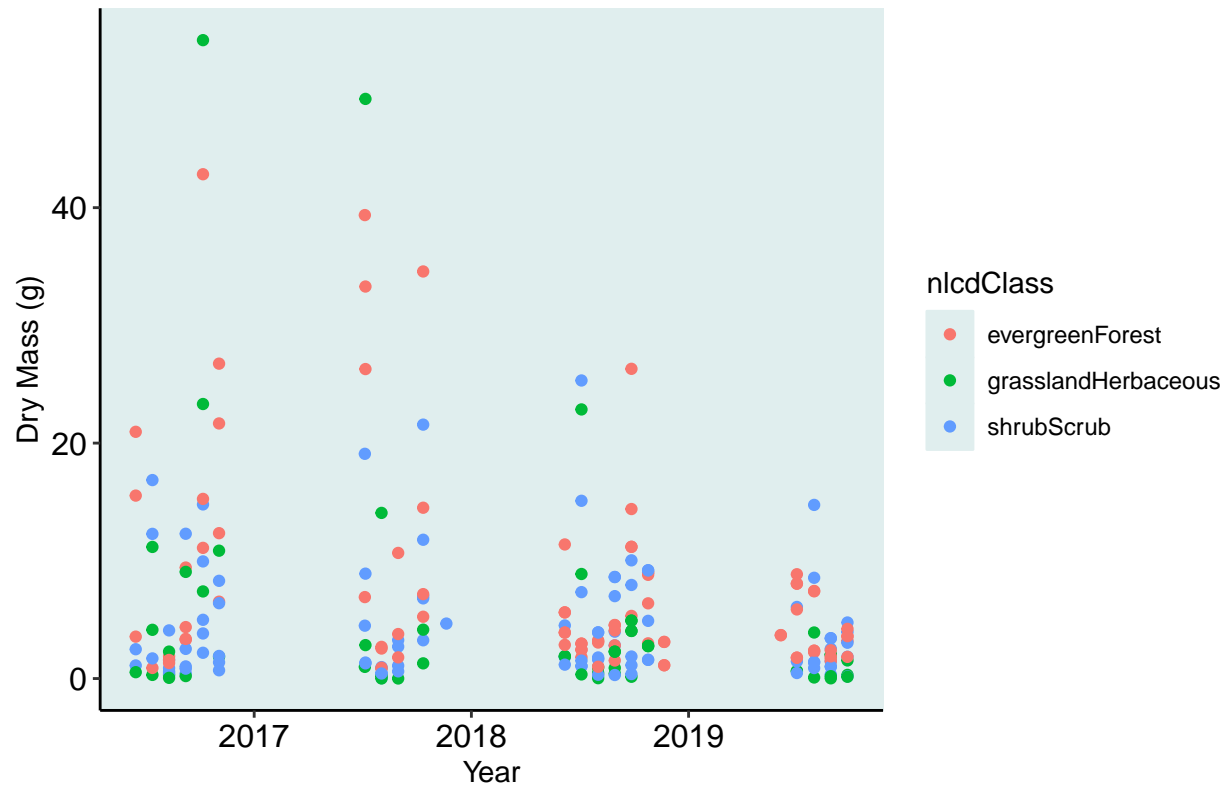
Answer: In the warmer months, extreme values of TN and TP increase in Peter lake, while they appear to remain somewhat consistent in Paul Lake. Overall, it appears Peter Lake has higher concentrations of TP and TN than Paul Lake.

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

```
#6
litter$functionalGroup <- as.factor(litter$functionalGroup)
litter$nlcdClass <- as.factor(litter$nlcdClass)

needles.plot1 <- litter |>
  filter(functionalGroup == "Needles") |>
  ggplot(aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point() +
  labs(title = "Needles Dry Mass over 2017, 2018, and 2019",
       x = "Year",
       y = "Dry Mass (g)") +
  theme
print(needles.plot1)
```

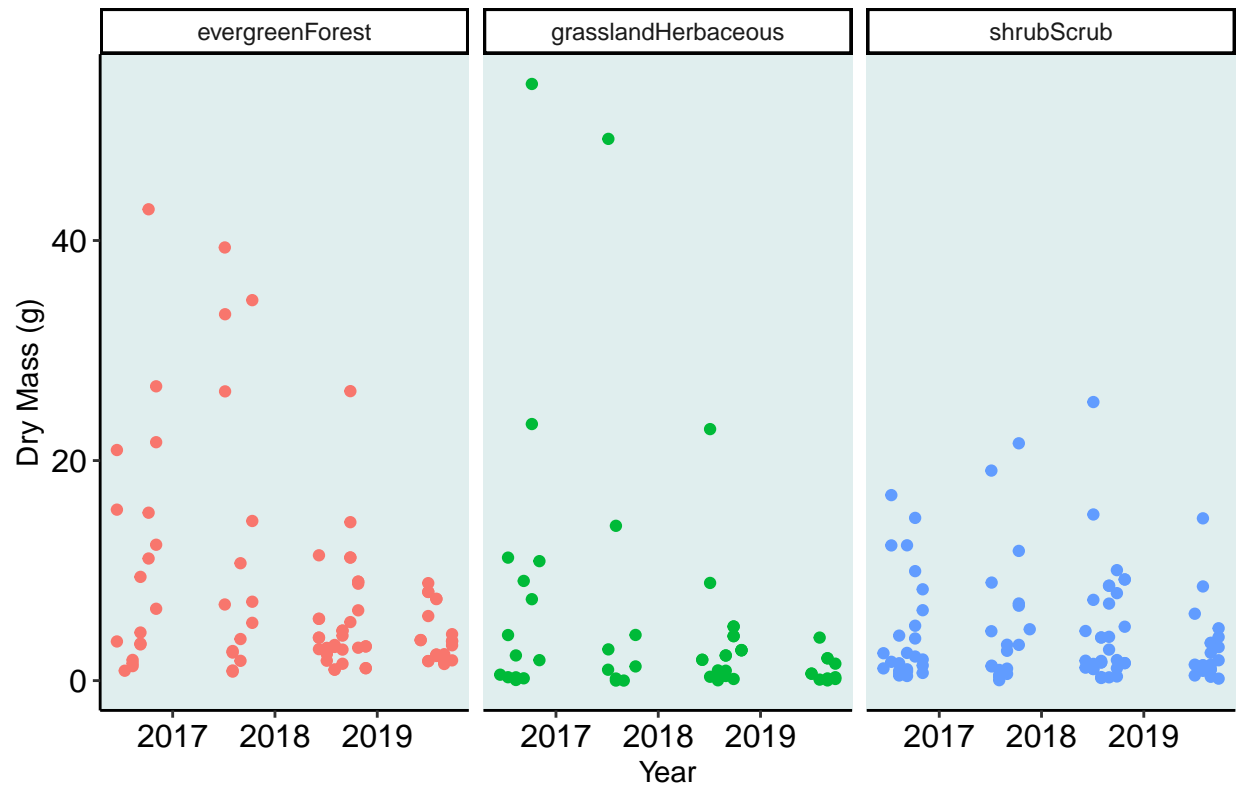
## Needles Dry Mass over 2017, 2018, and 2019



```
#7
needles.plot2 <- litter |>
  filter(functionalGroup == "Needles") |>
  ggplot(aes(x = collectDate, y = dryMass)) +
  geom_point(aes(color = nlcdClass)) +
  labs(title = "Needles Dry Mass over Time",
        x = "Year",
        y = "Dry Mass (g)") +
  facet_wrap(~ nlcdClass) +
  mytheme +
  theme(legend.position = "none")

print(needles.plot2)
```

## Needles Dry Mass over Time



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

Answer: Plot 7 is more effective at analyzing the different litter types because it displays the different nlcd classes in clear, comparable, plots. Plot 6 might be better for showing overall dry mass patterns overtime, however, it is a very busy graph. I prefer plot 7 overall because it separates the data into plots which are easier to read than in one big plot.