Assignment 5: Data Visualization

Hanna Bliska

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A02_CodingBasics.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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

The completed exercise is due on Friday, Oct 21st @ 5:00pm.

Set up your session

1. Set up your session. Verify your working directory and load the tidyverse, lubridate, & cowplot packages. Upload the NTL-LTER processed data files for
nutrients and chemistry/physics for Peter and Paul Lakes (use the tidy [`NTL-LTER_Lake_Chemistry_Nutrients_PeterP`
version) and the processed data file for the Niwot Ridge litter dataset (use the [`NEON_NIWO_Litter_mass_trap_Processe`
version).

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

```
# 1
getwd()
```

```
## [1] "/Users/hbliska/Desktop/EDA-Fall2022"
```

```
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   1.2.1      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()
```

```
library(lubridate)
```

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

```
library(cowplot)
```

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

```
Peter.Paul.Nutrients.Chem <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Pro
    stringsAsFactors = TRUE)
Niwot.Litter <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
    stringsAsFactors = TRUE)

# 2
Peter.Paul.Nutrients.Chem$sampledate <- as.Date(Peter.Paul.Nutrients.Chem$sampledate,
    format = "%Y-%m-%d")  #formatting date
Niwot.Litter$collectDate <- as.Date(Niwot.Litter$collectDate,
    format = "%Y-%m-%d")  #formatting date
```

Define your theme

3. Build a theme and set it as your default theme.

```
# 3
mytheme <- theme_classic(base_size = 12) +
    theme(axis.text = element_text(color = "black"),
        legend.position = "right")  #building my theme from the classic theme with black text, size 13
```

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.

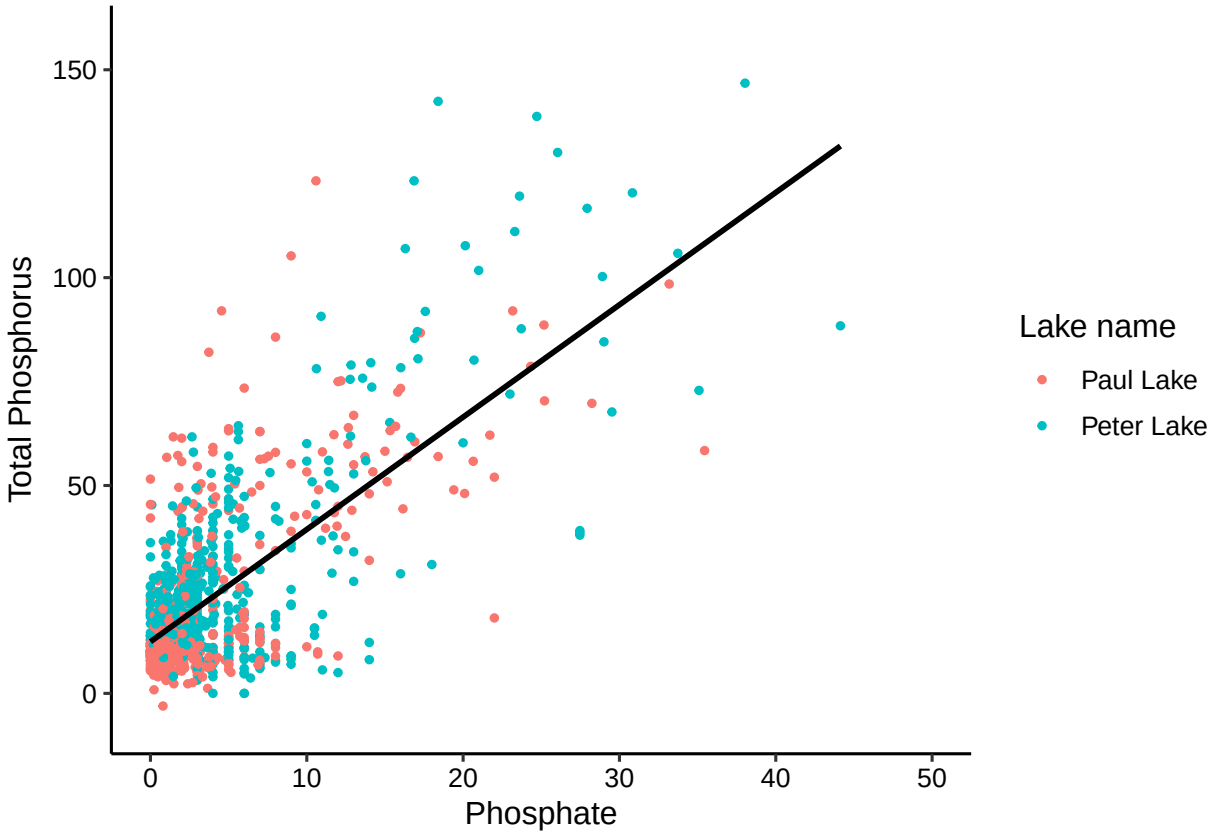
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```r
# 4
Plot_TotalP_Phosphate <- ggplot(Peter.Paul.Nutrients.Chem,
    aes(x = po4, y = tp_ug)) + geom_point(aes(color = lakename),
    size = 1) + xlim(0, 50) + geom_smooth(method = lm,
    se = FALSE, color = "black") + ylab(expression("Total Phosphorus")) +
    xlab(expression("Phosphate")) + labs(color = "Lake name") +
    mytheme
print(Plot_TotalP_Phosphate)
```

```
## `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).
```



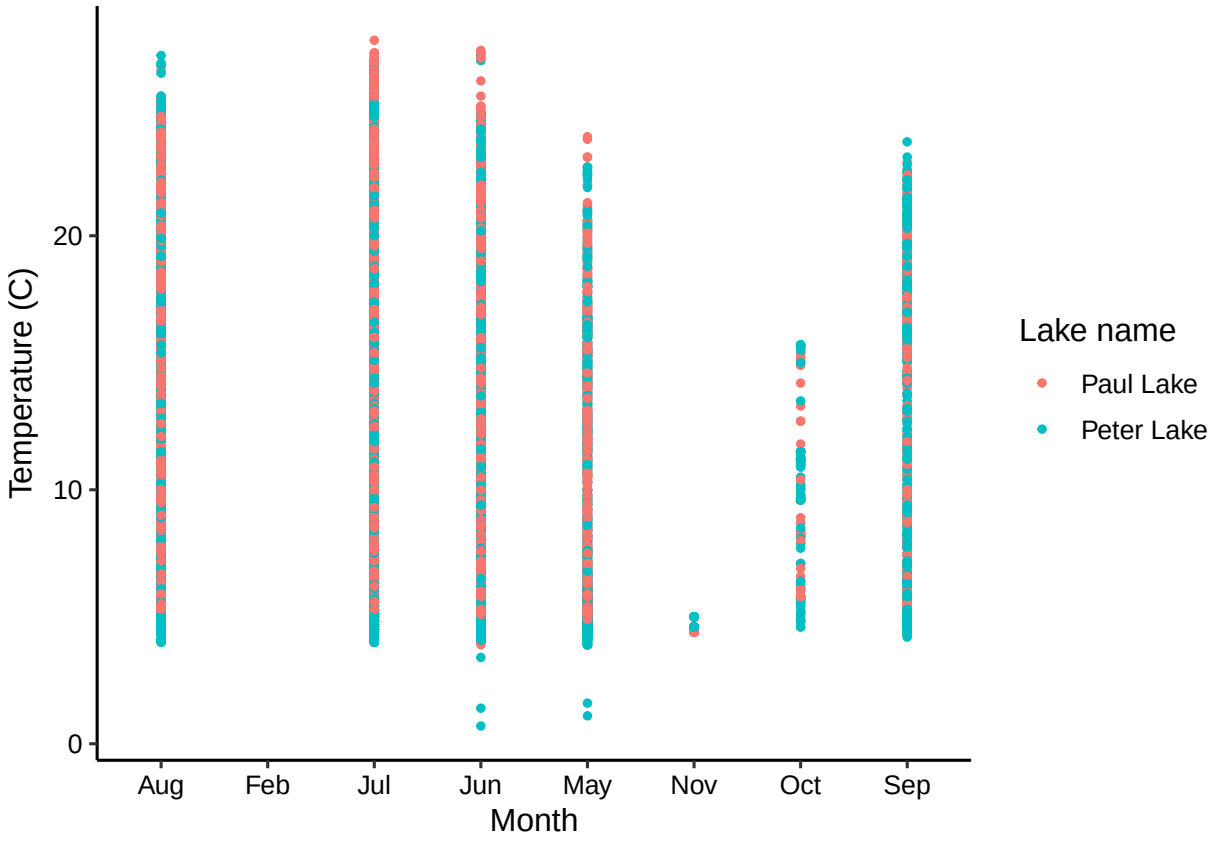
5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and

(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: R has a build in variable called `month.abb` that returns a list of months; see https://r-lang.com/month-abb-in-r-with-example

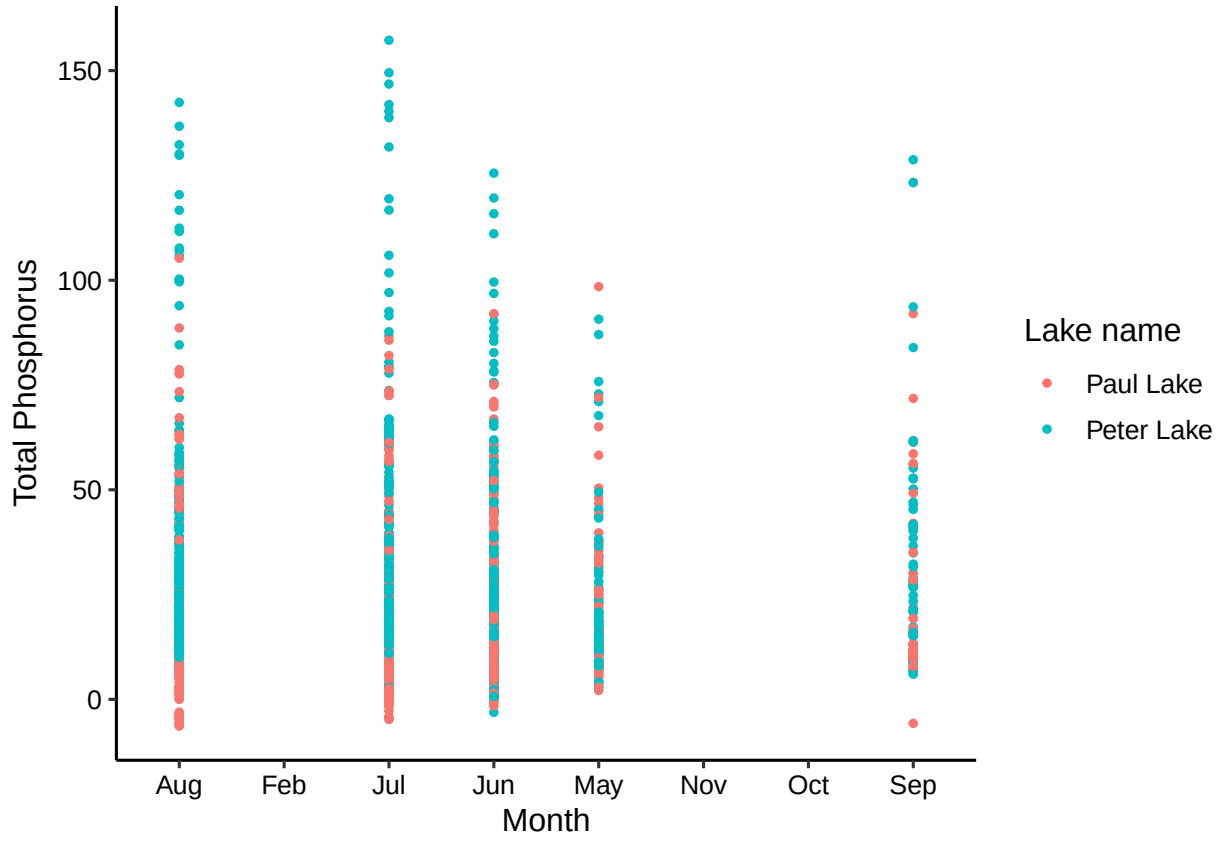
```
# 5
Plot_Temp <- ggplot(Peter.Paul.Nutrients.Chem,
    aes(x = month.abb[month], y = temperature_C)) +
    geom_point(aes(color = lakename), size = 1) +
    ylab(expression("Temperature (C)")) +
    xlab(expression("Month")) + labs(color = "Lake name") +
    mytheme
print(Plot_Temp)
```

```
## Warning: Removed 3566 rows containing missing values (geom_point).
```



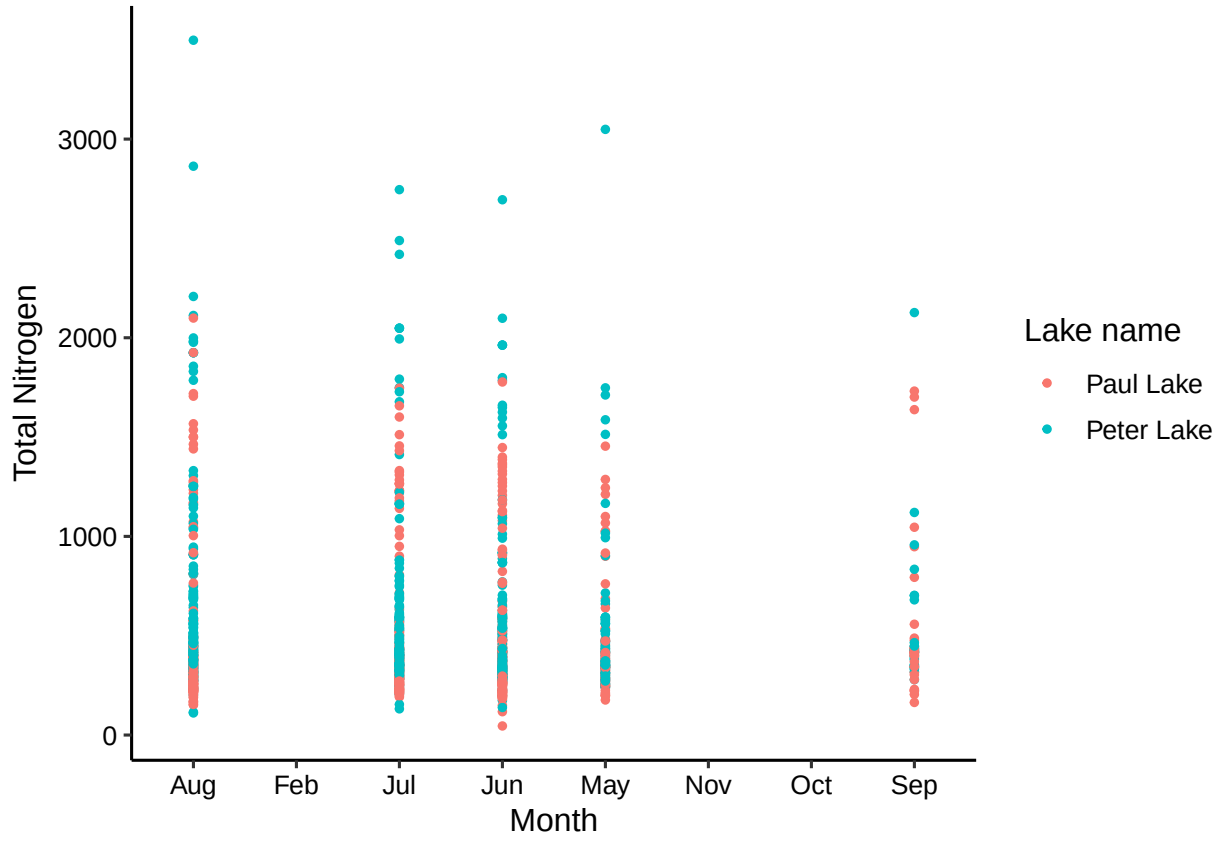

```
Plot_TP <- ggplot(Peter.Paul.Nutrients.Chem,
    aes(x = month.abb[month], y = tp_ug)) +
    geom_point(aes(color = lakename), size = 1) +
    ylab(expression("Total Phosphorus")) +
    xlab(expression("Month")) + labs(color = "Lake name") +
    mytheme
print(Plot_TP)
```

```
## Warning: Removed 20729 rows containing missing values (geom_point).
```




```
Plot_TN <- ggplot(Peter.Paul.Nutrients.Chem,
    aes(x = month.abb[month], y = tn_ug)) +
    geom_point(aes(color = lakename), size = 1) +
    ylab(expression("Total Nitrogen")) +
    xlab(expression("Month")) + labs(color = "Lake name") +
    mytheme
print(Plot_TN)
```

```
## Warning: Removed 21583 rows containing missing values (geom_point).
```




```
Lake_Name_Legend <- get_legend(Plot_TP +
    mytheme)
```

```
## Warning: Removed 20729 rows containing missing values (geom_point).
```

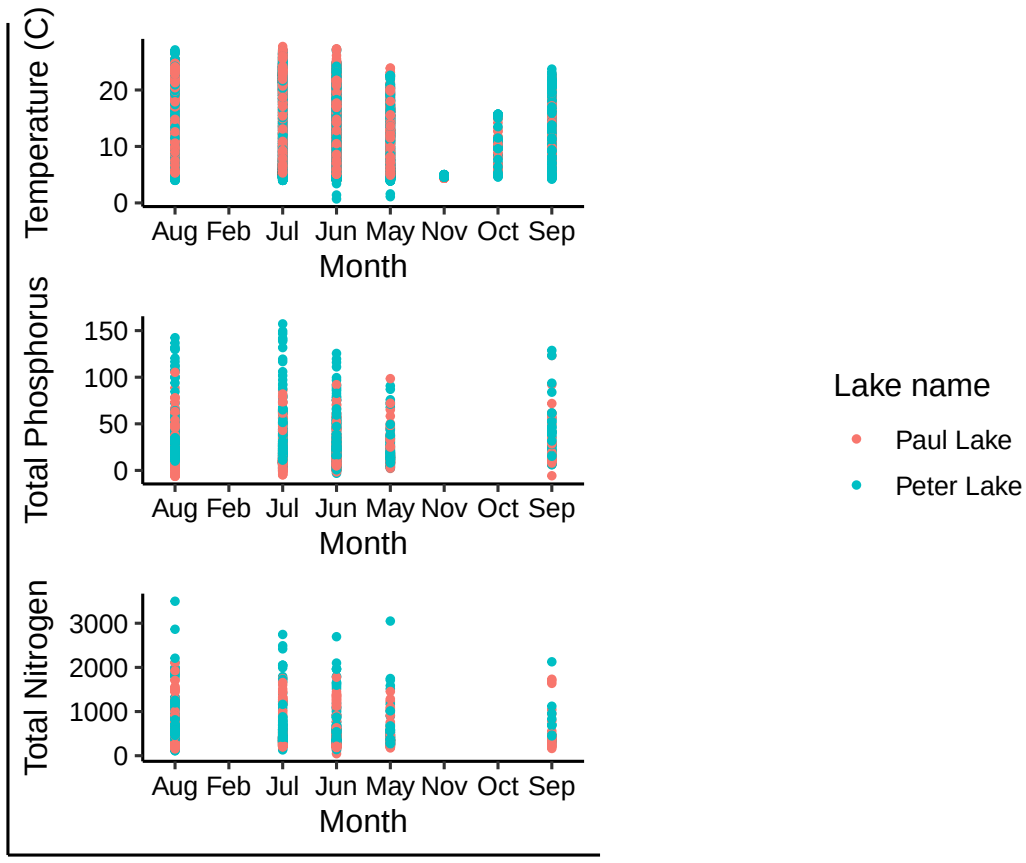
```
Plot_Temp_TP_TN <- plot_grid(Plot_Temp +
    theme(legend.position = "none"), Plot_TP +
    theme(legend.position = "none"), Plot_TN +
    theme(legend.position = "none"), ncol = 1,
    align = "hv", rel_widths = c(10, 10),
    rel_heights = c(10, 10)) + mytheme
```

```
## Warning: Removed 3566 rows containing missing values (geom_point).
```

```
## Warning: Removed 20729 rows containing missing values (geom_point).
```

```
## Warning: Removed 21583 rows containing missing values (geom_point).
```

```
Plot_Temp_TP_TN_Legend <- plot_grid(Plot_Temp_TP_TN,
    Lake_Name_Legend)
print(Plot_Temp_TP_TN_Legend)
```



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

Answer:

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

# 7
```

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

Answer: