Assignment 5: Data Visualization

Lehe Xu

OVERVIEW

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

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Fay_A05_DataVisualization.Rmd") prior to submission.

The completed exercise is due on Monday, February 14 at 7:00 pm.

Set up your session

1. Set up your session. Verify your working directory and load the tidyverse and 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_PeterPaul_Processed.csv`] version) and the processed data file for the Niwot Ridge litter dataset (use the [`NEON_NIWO_Litter_mass_trap_Processed.csv`] version).

```
#1
getwd()
```

```
## [1] "D:/Documents/Environmental_Data_Analytics_2022"
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.6.3
```

```
## -- Attaching packages ---------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.1     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 3.6.3
```

```
## Warning: package 'tidyr' was built under R version 3.6.3
```

```
## Warning: package 'readr' was built under R version 3.6.3
```

```
## Warning: package 'purrr' was built under R version 3.6.3
```

```
## Warning: package 'dplyr' was built under R version 3.6.3
```

```
## Warning: package 'forcats' was built under R version 3.6.3
```

```
## -- Conflicts ------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(ggridges)
```

```
## Warning: package 'ggridges' was built under R version 3.6.3
```

```r
NTL <-
  read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
           stringsAsFactors = TRUE)
NEON <-
  read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv", stringsAsFactors = TRUE)
```

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

```r
#2
NTL$sampledate <- as.Date(NTL$sampledate, format = "%Y-%m-%d")
NEON$collectDate <- as.Date(NEON$collectDate, format = "%Y-%m-%d")
```

Define your theme

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

```r
#3
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "darkblue"),
        legend.position = "bottom")

theme_set(mytheme)
```

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.

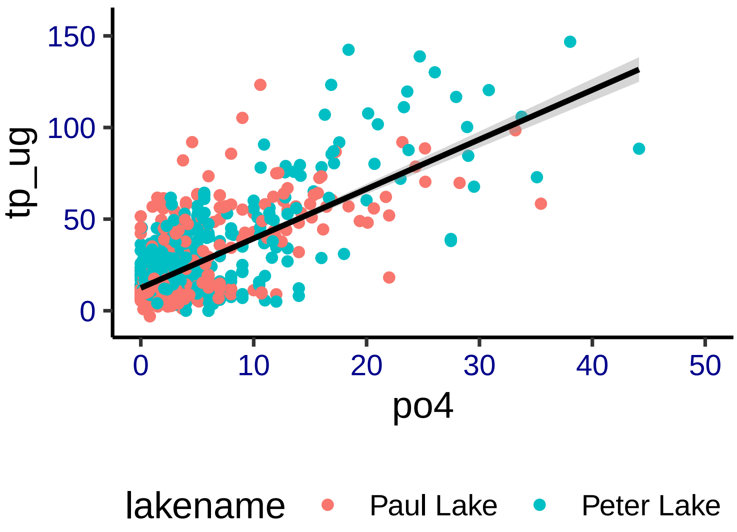
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and `ylim()`).

```r
#4
tp.po4 <-
  ggplot(NTL, aes(x = po4, y = tp_ug, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, color="black")+
  xlim(0, 50)
  ylim(0, 150)
```

```
## <ScaleContinuousPosition>
##  Range:
##  Limits:    0 --  150
```

```r
print(tp.po4)
```

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



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.

```
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 3.6.3
```

```
#5
NTL$month <- as.character(NTL$month)
#(a)
TEMM <-
  ggplot(NTL, aes(x = month, y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  theme(legend.position = "none")
#(b)
TPM <-
  ggplot(NTL, aes(x = month, y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  theme(legend.position = "none")
#(c)
TNM <-
  ggplot(NTL, aes(x = month, y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  theme(legend.position = "none")

legend_a <- get_legend(TEMM + theme(legend.position="bottom"))
```

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

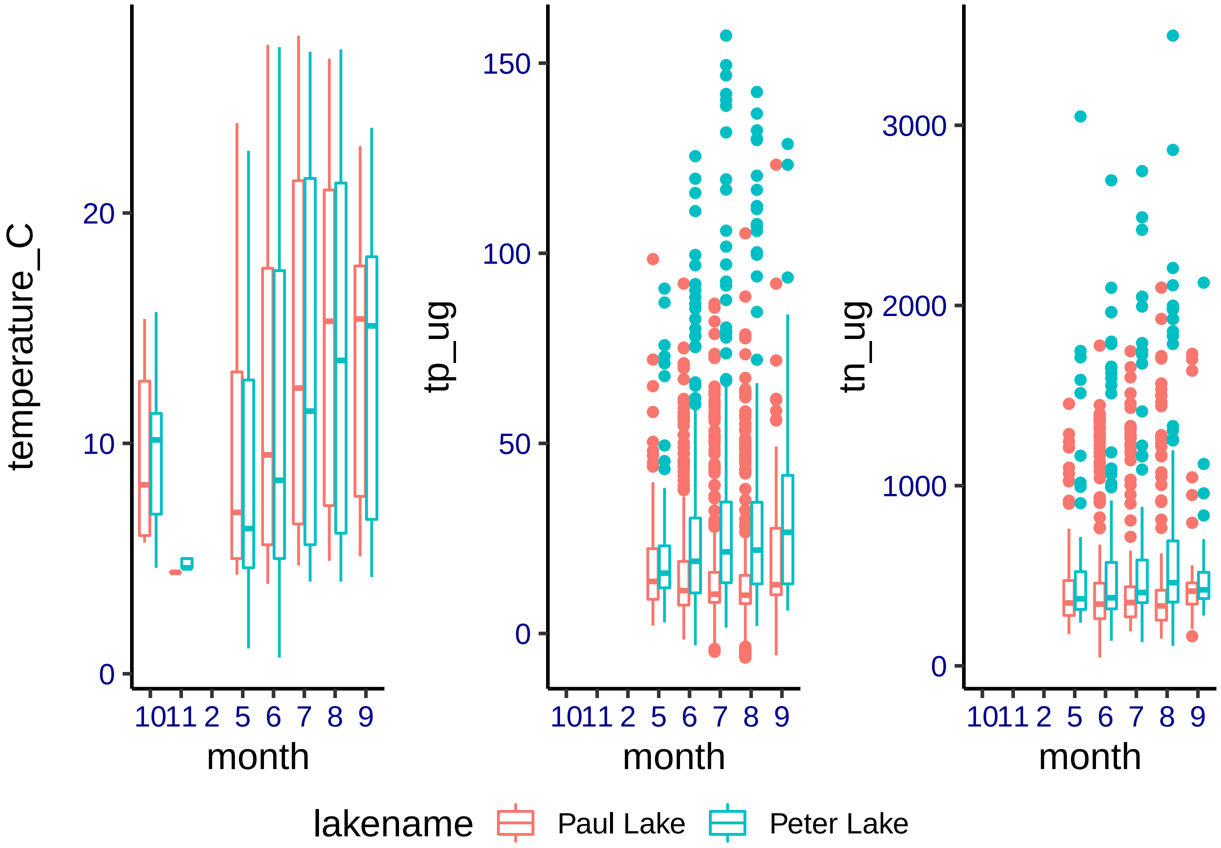
```
complot <- plot_grid(TEMM, TPM, TNM, align = 'vh', nrow = 1)
```

```
## 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).
```

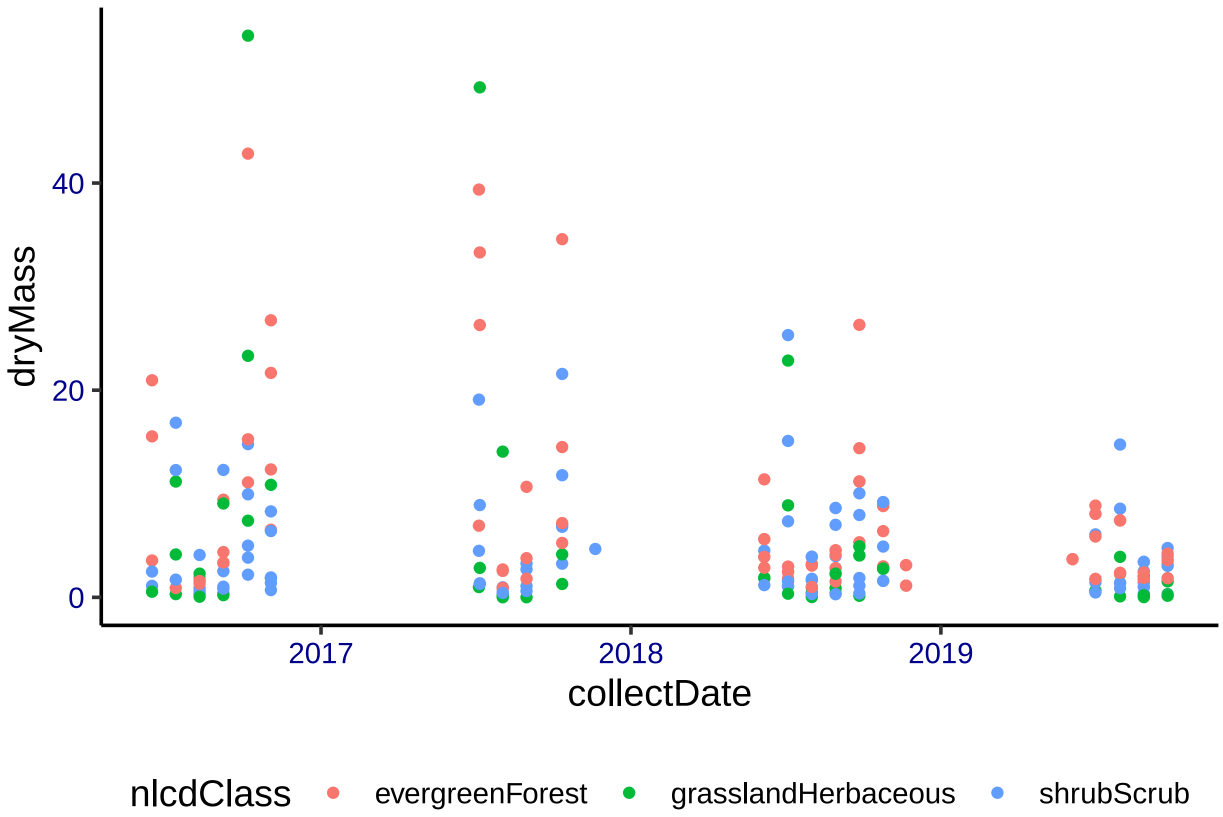
```
complot1 <- plot_grid(complot, legend_a , ncol = 1, rel_heights = c(3,.2))
print(complot1)
```



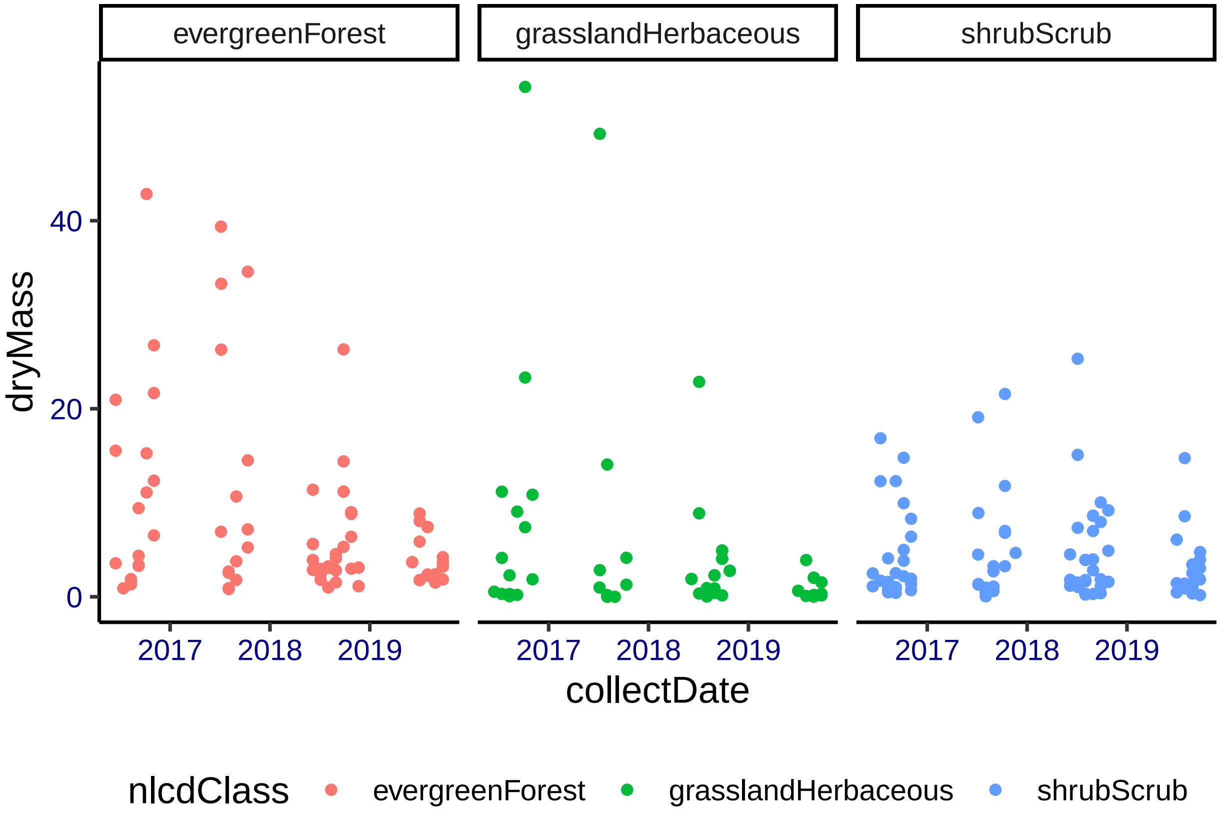
Question: What do you observe about the variables of interest over seasons and between lakes? > Answer: #For temeprature, lakes have the highest temperature in summer, and lowest temperature in winter. The temepratures in spring and autumn are mild. The two lakes have very similar temeprature, but ususally Paul lake has higher average temeprature than Peter Lake. #For total phosphorus(tp_ug), in Peter Lake, it is higher in summer and lower in spring. BUt, in Paul Lake, it is consistent; in summer, there is only a slightly lower TP. Peter lake has higher tp_ug than Paul lake. #For total nutrients (tn_ug), it is generally consistent for every lake. Peter lake may have a sligtly higher nutrient level in August. And, Peter lake has higher total nutrients 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
Needles1 <-
  ggplot(subset(NEON, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color=nlcdClass)) +
  geom_point()
print(Needles1)
```



```
#7
Needles2 <-
  ggplot(subset(NEON, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color=nlcdClass)) +
  geom_point() +
  facet_wrap(vars(nlcdClass), nrow = 1)
print(Needles2)
```



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

> Answer: I think the plot in #7 is more effective, because all the dots in the same class can be clearly observed. However, the plot #6 many dots in different classes overlapped with each other, which makes the trend hard to observe.