Assignment 5: Data Visualization

Israel Golden

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).

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

```
#1
getwd()
```

```
## [1] "/Users/israelgolden/Desktop/School/MEM/Semester 4/ENV 872/GitHub Repos/Environmental_Data_Analy
```

```
setwd("/Users/israelgolden/Desktop/School/MEM/Semester 4/ENV 872/GitHub Repos/Environmental_Data_Analyti
#Load packages
library(tidyverse)
```

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

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

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

```
library(cowplot)
#load data
PeterPaul.chem.nutrients <- read.csv("../Environmental_Data_Analytics_2022/Data/Processed/NTL-LTER_Lake_
NEON.litter <- read.csv("../Environmental_Data_Analytics_2022/Data/Processed/NEON_NIWO_Litter_mass_trap_

#2
PeterPaul.chem.nutrients$sampledate <- as.Date(PeterPaul.chem.nutrients$sampledate, format = "%Y-%m-%d")
class(PeterPaul.chem.nutrients$sampledate)
```

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

```
NEON.litter$collectDate <- as.Date(NEON.litter$collectDate, format = "%Y-%m-%d")
class(NEON.litter$collectDate)
```

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

Define your theme

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

```
#3
mytheme <- theme_bw(base_size = 10) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top") #alternative: legend.position + legend.justification

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.

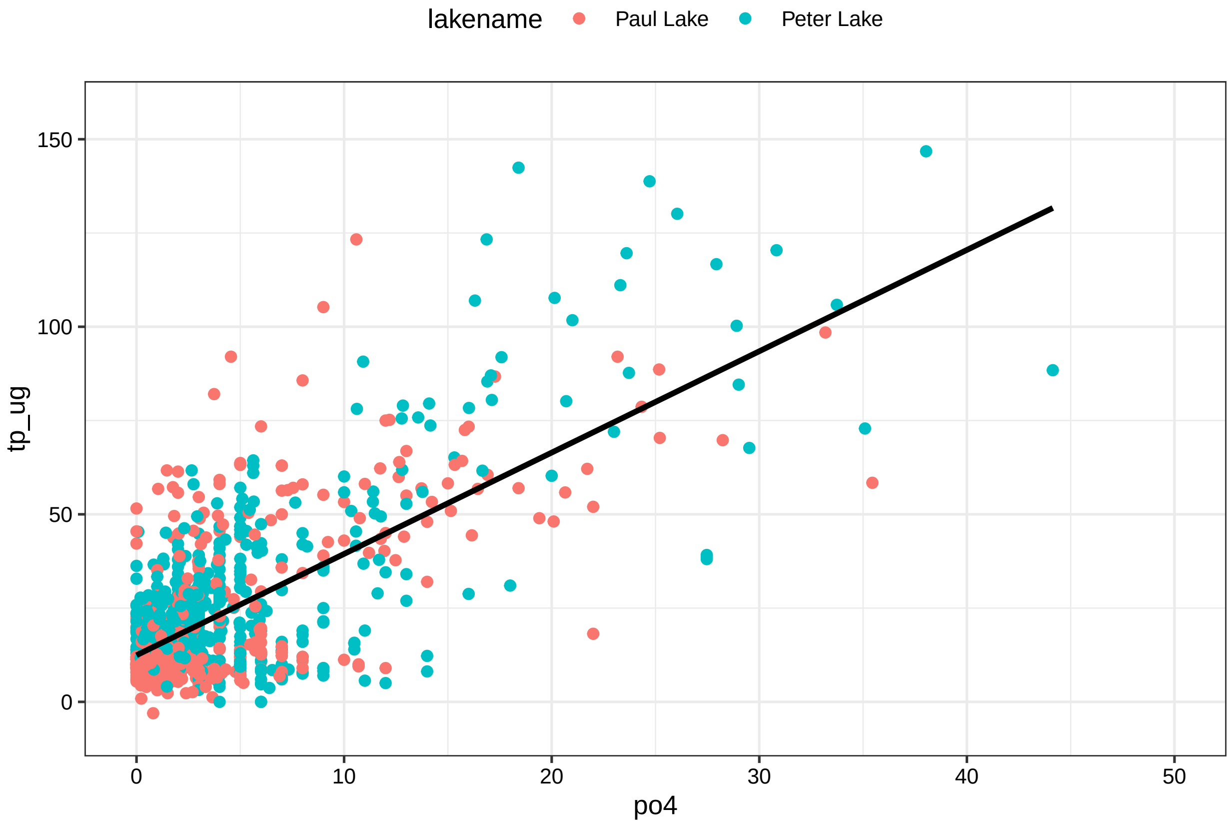
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and `ylim()`).

```
#4
ggplot(PeterPaul.chem.nutrients, aes(x = po4, y = tp_ug, col = lakename)) +
  geom_point() +
  xlim(0,50) +
  geom_smooth(method = lm, col = "black", se = FALSE)
```

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

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

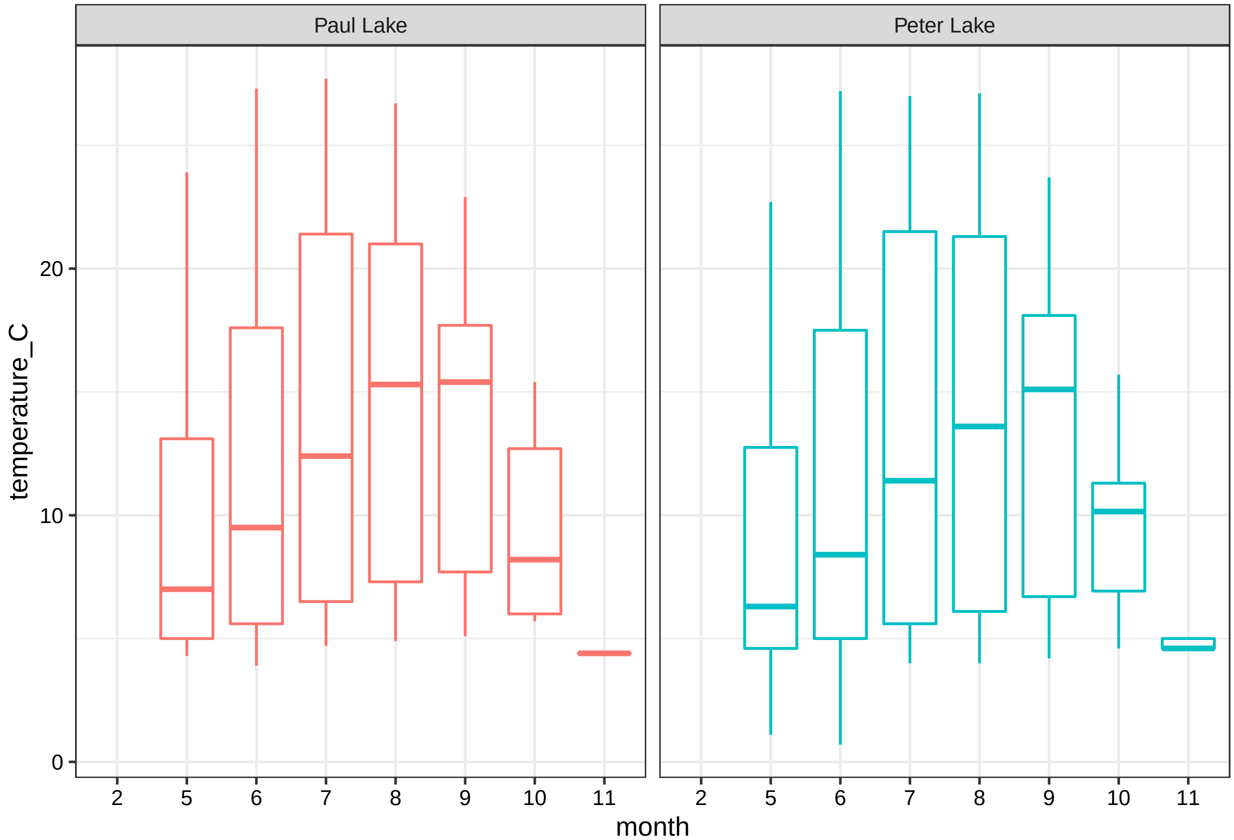
```
## Warning: Removed 21947 rows containing missing values (geom_point).
```



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.

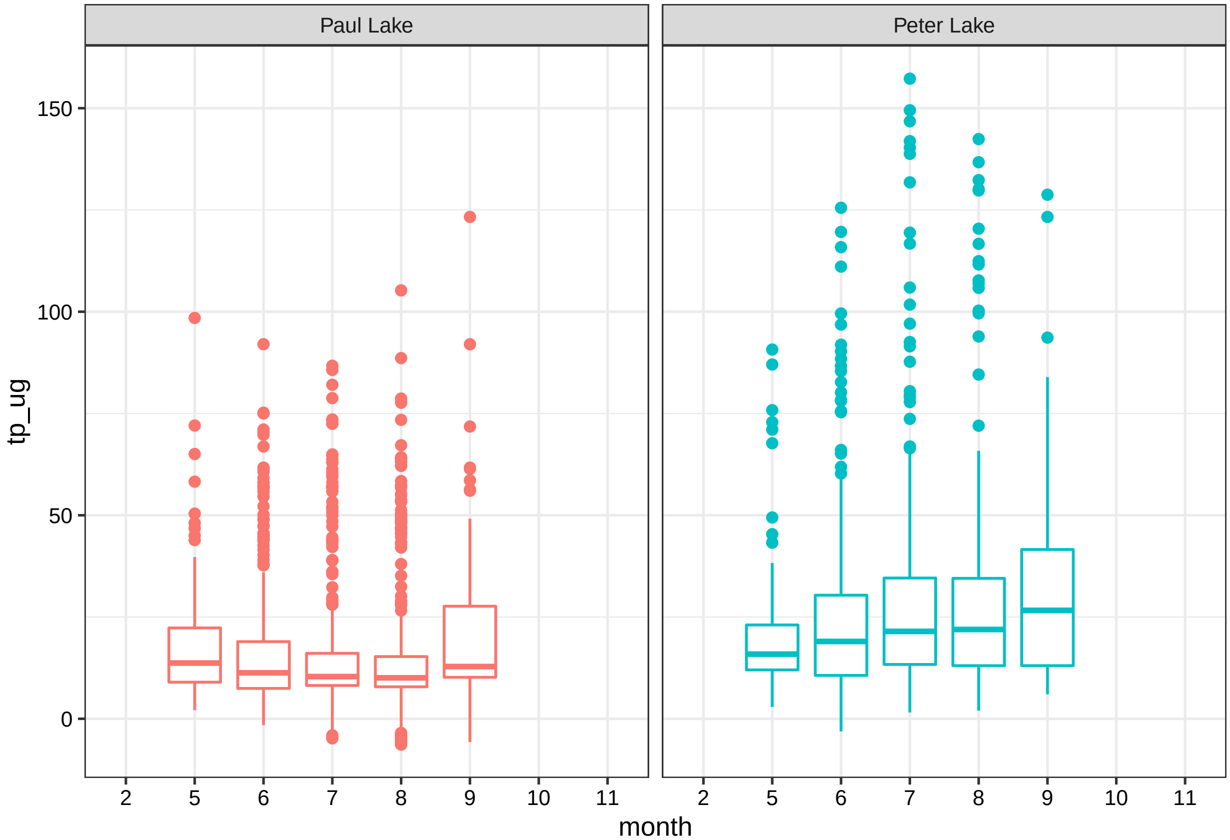
```
#5
PeterPaul.chem.nutrients$month <- as.factor(PeterPaul.chem.nutrients$month)

temp_box <- ggplot(PeterPaul.chem.nutrients, aes(x = month, y = temperature_C, col = lakename)) +
  geom_boxplot() +
  facet_wrap("lakename") +
  theme(legend.position = "none")
print(temp_box)
```

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



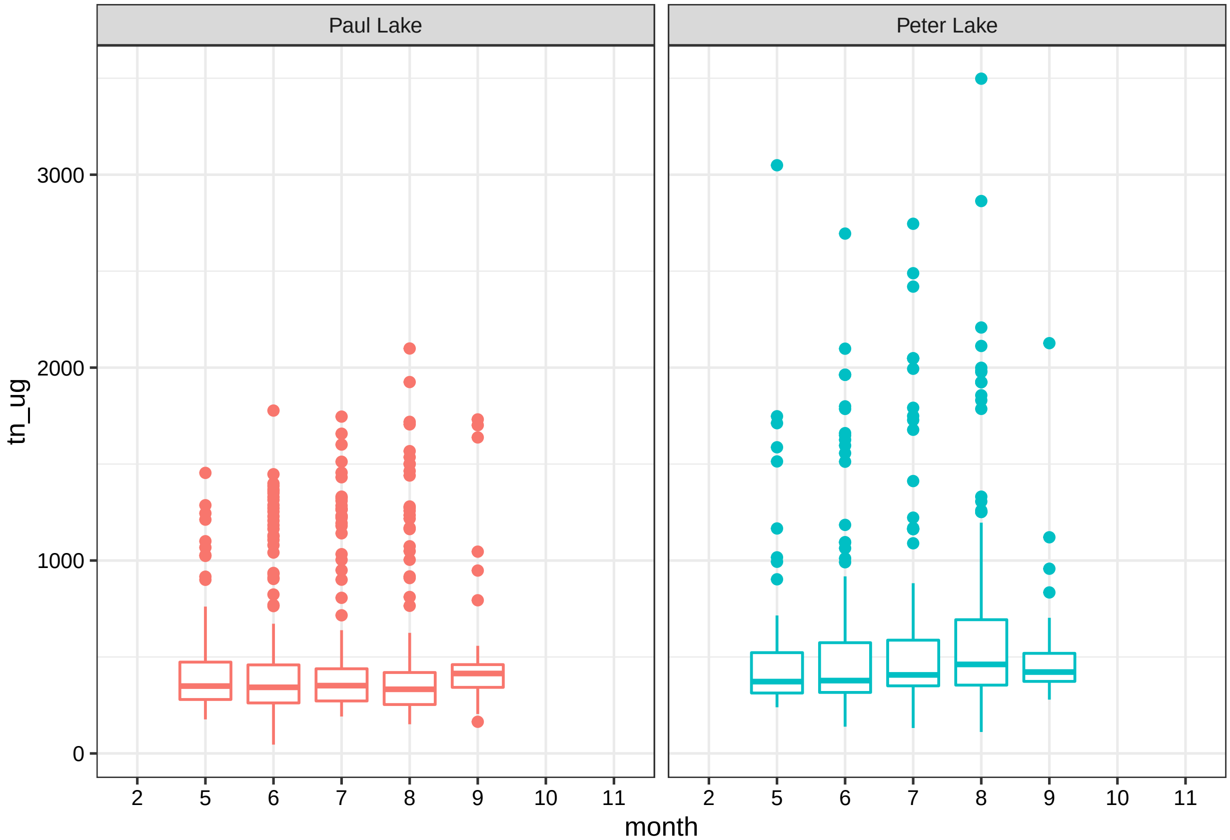

```
TP_box <- ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  facet_wrap("lakename") +
  theme(legend.position = "none")
print(TP_box)
```

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




```
TN_box <- ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  facet_wrap("lakename") +
  theme(legend.position = "none")
print(TN_box)
```

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

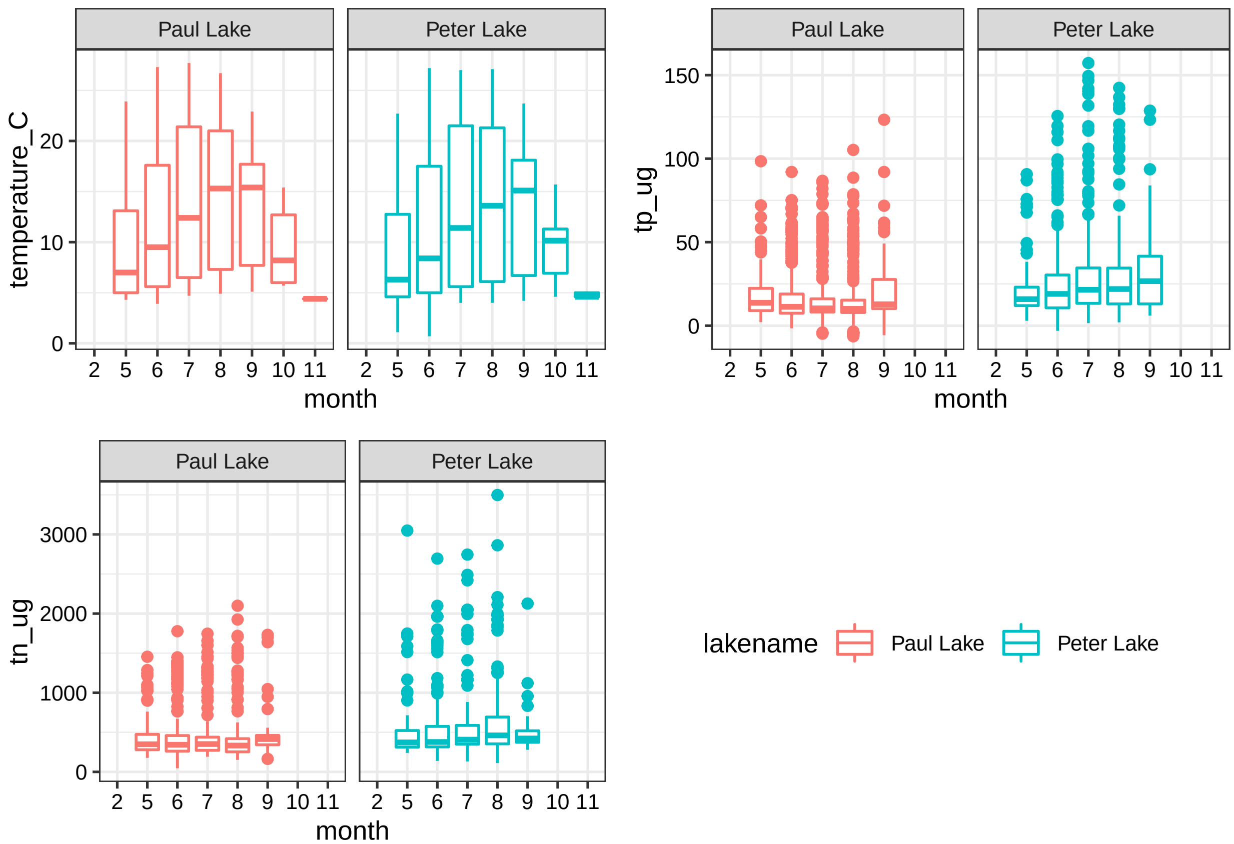



```
temp_box2 <- ggplot(PeterPaul.chem.nutrients, aes(x = month, y = temperature_C, col = lakename)) +
  geom_boxplot() +
  facet_wrap("lakename")
mylegend <- get_legend(temp_box2)
```

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

```
library(cowplot)
plot_grid(temp_box, TP_box, TN_box, mylegend, nrow = 2)
```

```
## 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).
```

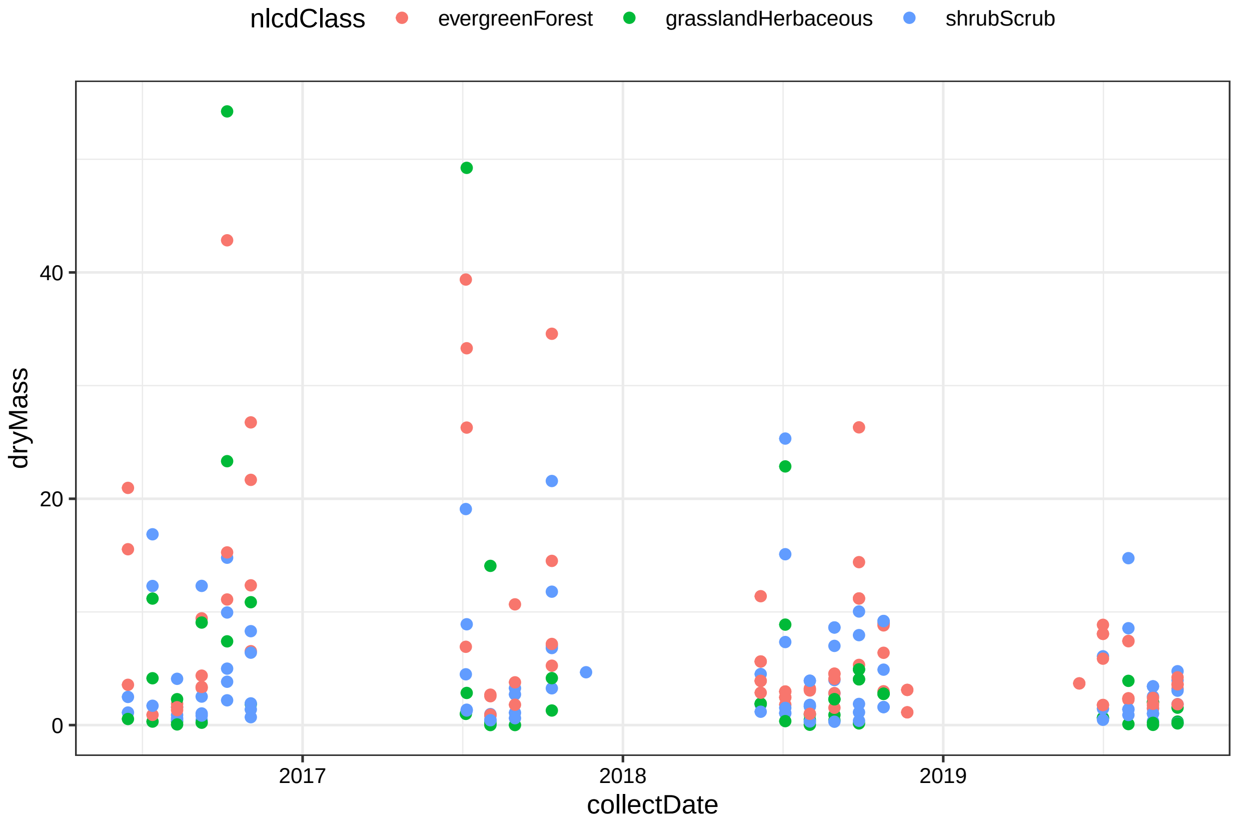


Question: What do you observe about the variables of interest over seasons and between lakes?

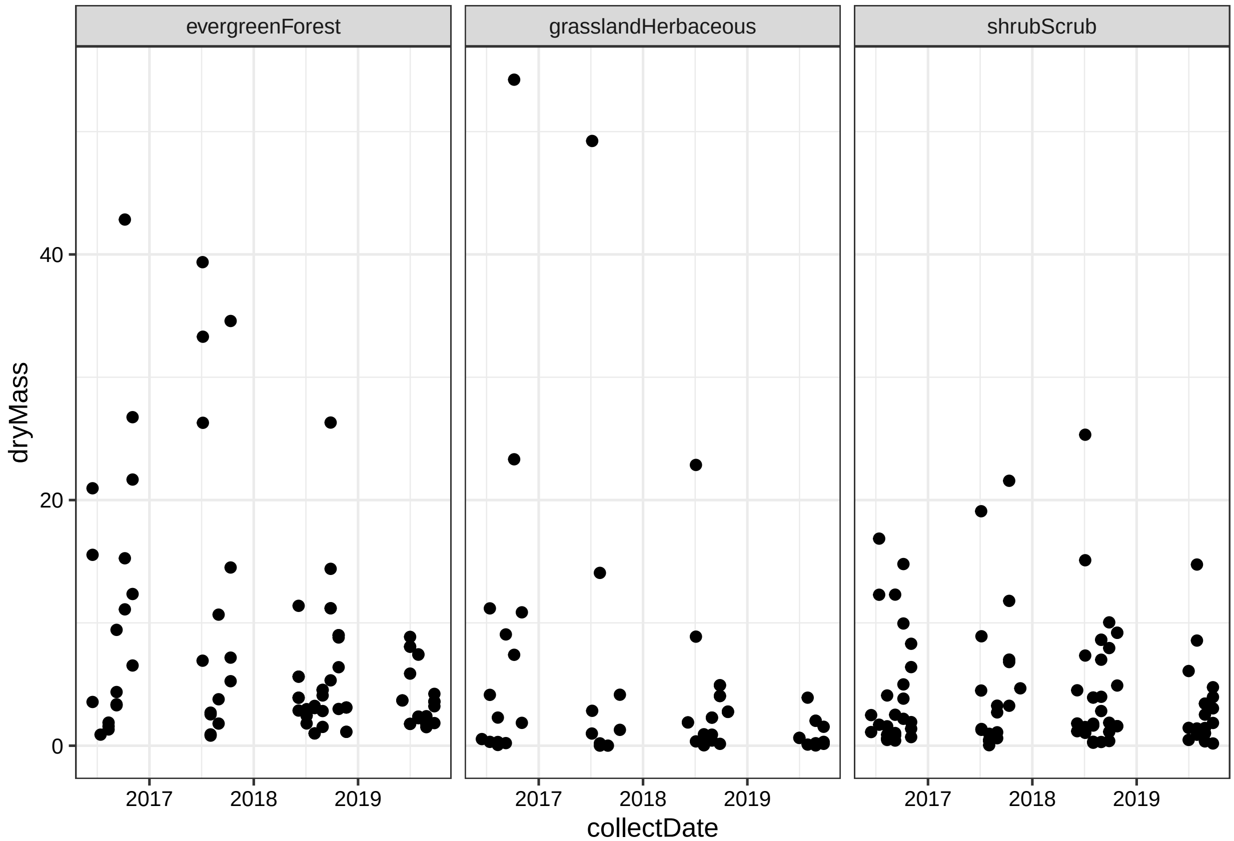
> Answer: I notice that temperature behaves just as I would expect it to with higher temperatures in the Summer months and lower temperatures in the Winter months for both lakes. ‘tp_ug’ appears to marginally decrease during Summer months in Paul Lake and rise again during Winter months. This trend does not hold for Peter Lake’s ‘tp_ug’ - here the mean tp_ug rises with each month’s data. However, there are quite a few outliers in Peter Lake that seem to suggest ‘tp_ug’ rise during warm months (peaking in July) and decrease thereafter. There does not seem to be much movement in ‘tn_ug’ in Paul Lake between months whereas in Peter Lake ‘tn_ug’ rises modestly between May and August and declines somewhat in September.

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
ggplot(filter(NEON.litter, functionalGroup == "Needles")) +
  geom_point(aes(x=collectDate, y=dryMass, color = nlcdClass))
```




```
#7
ggplot(filter(NEON.litter, functionalGroup == "Needles")) +
  geom_point(aes(x=collectDate, y=dryMass)) +
  facet_wrap('nlcdClass')
```



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

Answer: I think that plot 7 is the more effective plot because it is difficult to assess annual trends in dry mass by functional group when all points (representing different functional groups) are bunched together. Even though it is less colorful, graph 7 displays the same information but more clearly by separating annual trends in dry mass by functional group.