Assignment 5: Data Visualization

Amanda Braun

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., "Salk_A05_DataVisualization.Rmd") prior to submission.

The completed exercise is due on Tuesday, February 11 at 1:00 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 (tidy and gathered) and the processed data file for the Niwot Ridge litter dataset.
2. Make sure R is reading dates as date format; if not change the format to date.

```
#1
getwd()
```

```
## [1] "/Users/amandabraun/Documents/Classes Spring 2020/Data Analytics/Environmental_Data_Analytics_202
```

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------ tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

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

```
library(cowplot)
```

```
##
## ********************************************************

## Note: As of version 1.0.0, cowplot does not change the

##   default ggplot2 theme anymore. To recover the previous
```

```
##   behavior, execute:
##   theme_set(theme_cowplot())
```

```
## ********************************************************
```

```
NT_LTER_Chemistry_PPLakes <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Proc

NT_LTER_Nutrients_PPLake <- read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaulGathered_Process

NiwotRidge_Litter_Processed <-read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")

#2
class(NT_LTER_Chemistry_PPLakes$sampledate)
```

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

```
NT_LTER_Chemistry_PPLakes$sampledate <- as.Date(NT_LTER_Chemistry_PPLakes$sampledate, format = "%Y-%m-%
class(NT_LTER_Chemistry_PPLakes$sampledate)
```

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

```
class(NT_LTER_Nutrients_PPLake$sampledate)
```

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

```
NT_LTER_Nutrients_PPLake$sampledate <- as.Date(NT_LTER_Nutrients_PPLake$sampledate, format = "%Y-%m-%d")
class(NT_LTER_Nutrients_PPLake$sampledate)
```

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

```
class(NiwotRidge_Litter_Processed$collectDate)
```

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

```
NiwotRidge_Litter_Processed$collectDate <- as.Date(NiwotRidge_Litter_Processed$collectDate, format = "%Y
class(NiwotRidge_Litter_Processed$collectDate)
```

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

Define your theme

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

```
amandastheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "blue"), legend.position = "top")
theme_set(amandastheme)
```

Create graphs

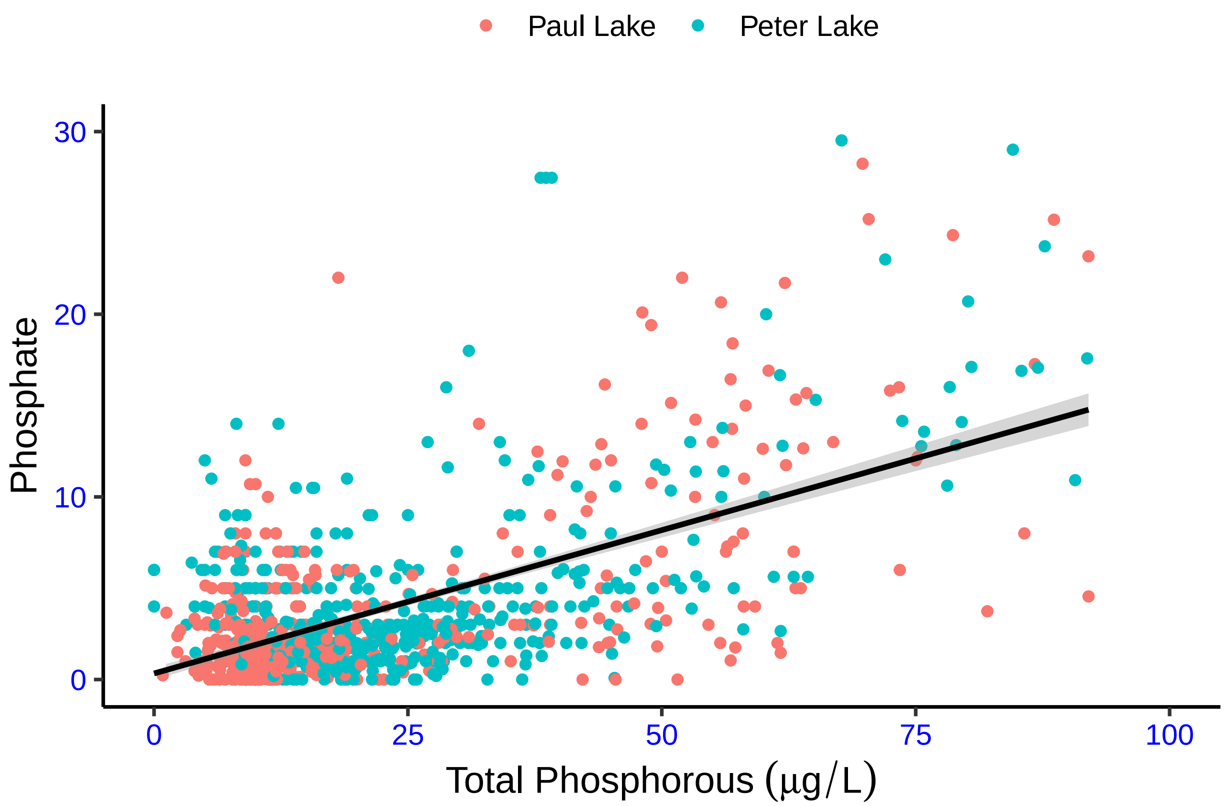
For numbers 4-7, create ggplot graphs and adjust aesthetics to follow best practices for data visualization. Ensure your theme, color palettes, axes, and additional aesthetics are edited accordingly.

4. [NTL-LTER] Plot total phosphorus by phosphate, 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.

```
NT_LTER_Chemistry_PPLakes <- drop_na(NT_LTER_Chemistry_PPLakes, po4, tp_ug)

plot1 <-ggplot(NT_LTER_Chemistry_PPLakes, aes(x = tp_ug, y = po4, color = lakename))+
  geom_point() +
  xlim(0,100) +
```

```
  ylim(0, 30)+
  geom_smooth(method = lm, color = "black")+
  labs(x = expression("Total Phosphorous" ~ (mu*g / L)), y = "Phosphate", color = "")
print(plot1)
```

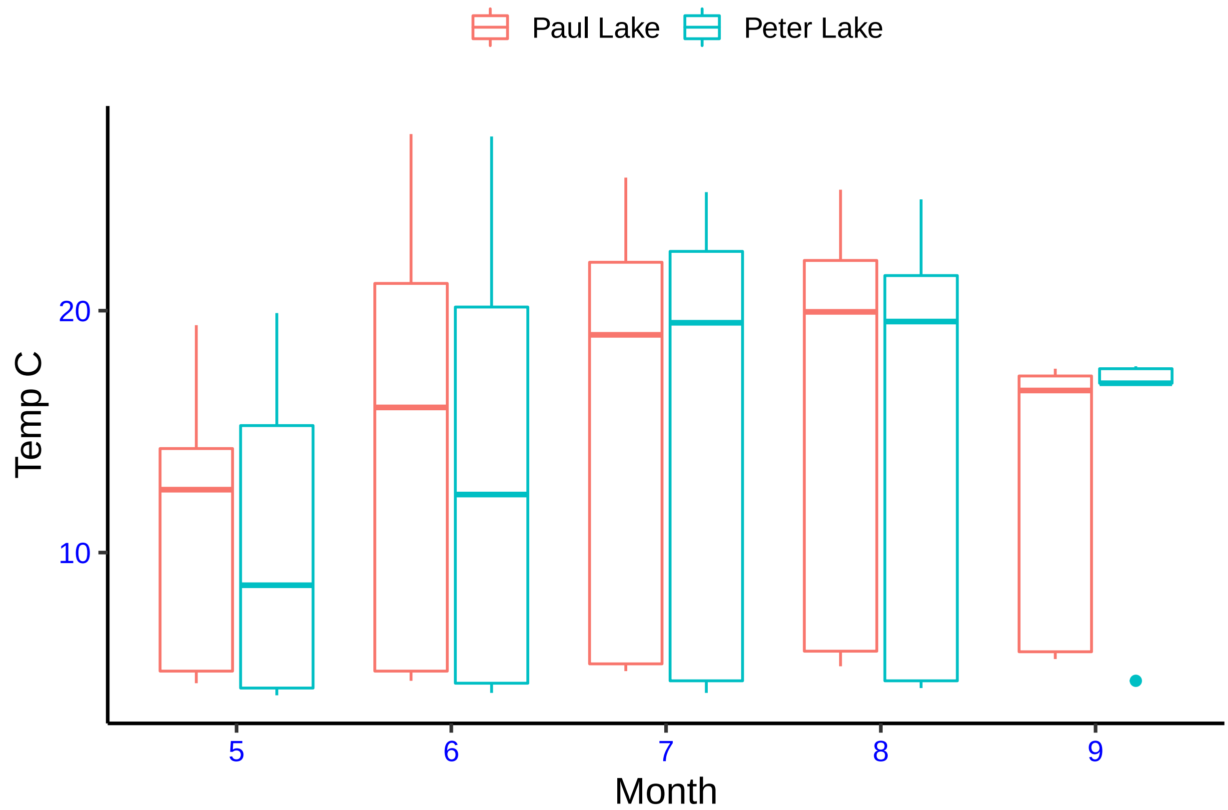
```
## Warning: Removed 22 rows containing non-finite values (stat_smooth).

## Warning: Removed 22 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.

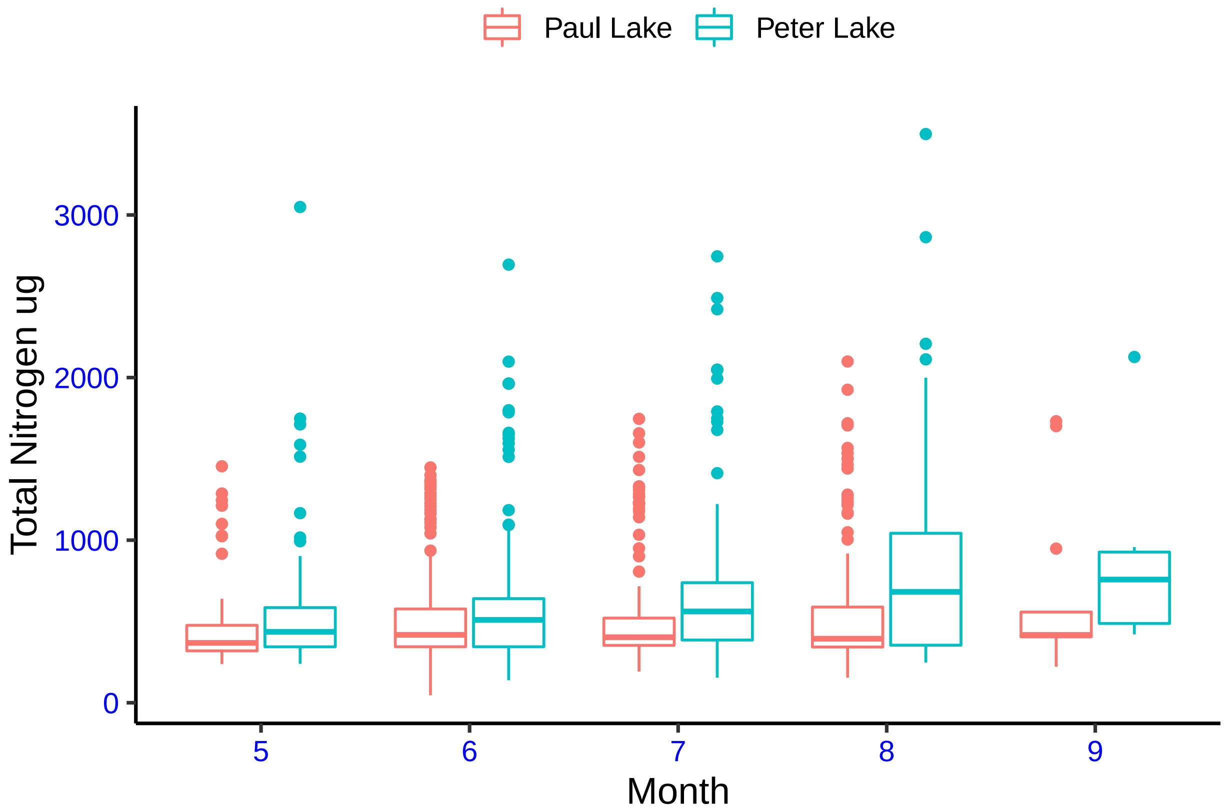
```
tempplot <- ggplot(NT_LTER_Chemistry_PPLakes, aes(x = as.factor(month), y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  labs(x = "Month", y = "Temp C", color = "")
print(tempplot)
```

```
## Warning: Removed 601 rows containing non-finite values (stat_boxplot).
```

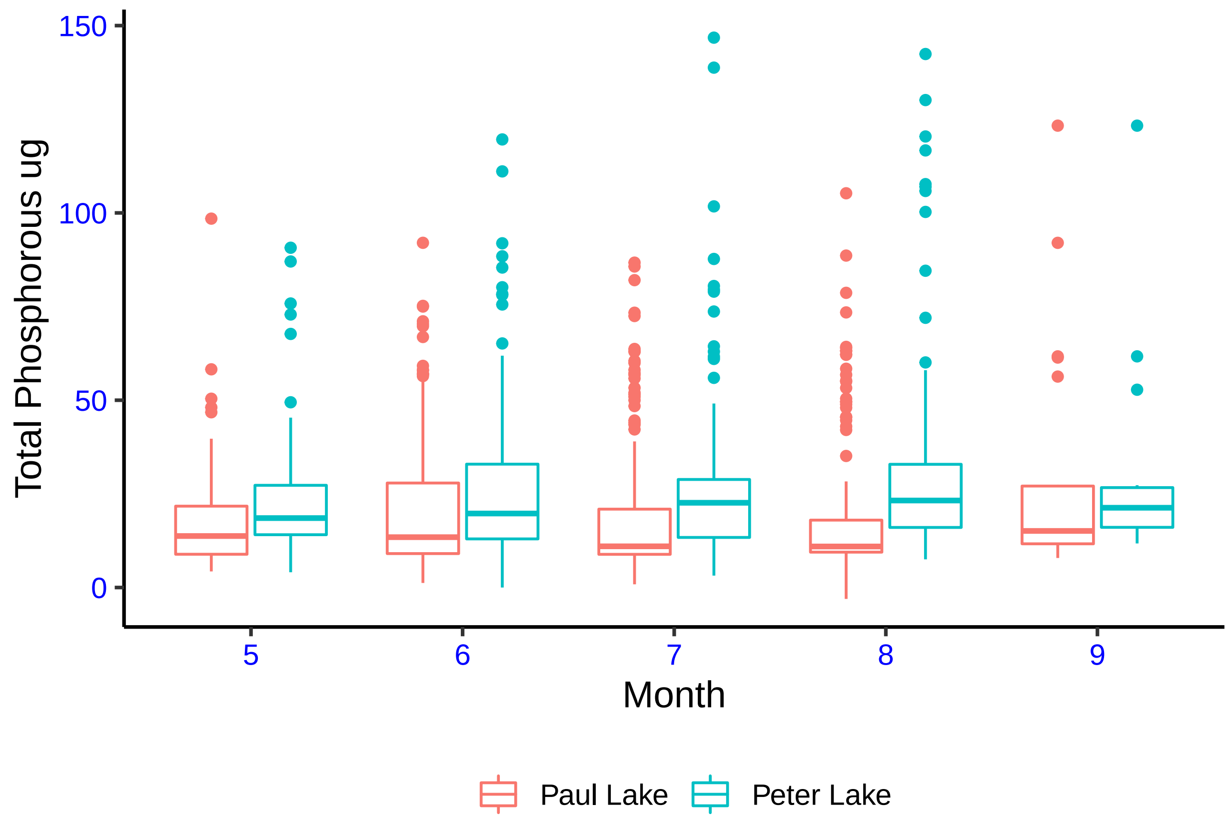



```
TNplot <- ggplot(NT_LTER_Chemistry_PPLakes, aes(x = as.factor(month), y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
    labs(x = "Month", y = "Total Nitrogen ug", color = "")
print(TNplot)
```

```
## Warning: Removed 283 rows containing non-finite values (stat_boxplot).
```



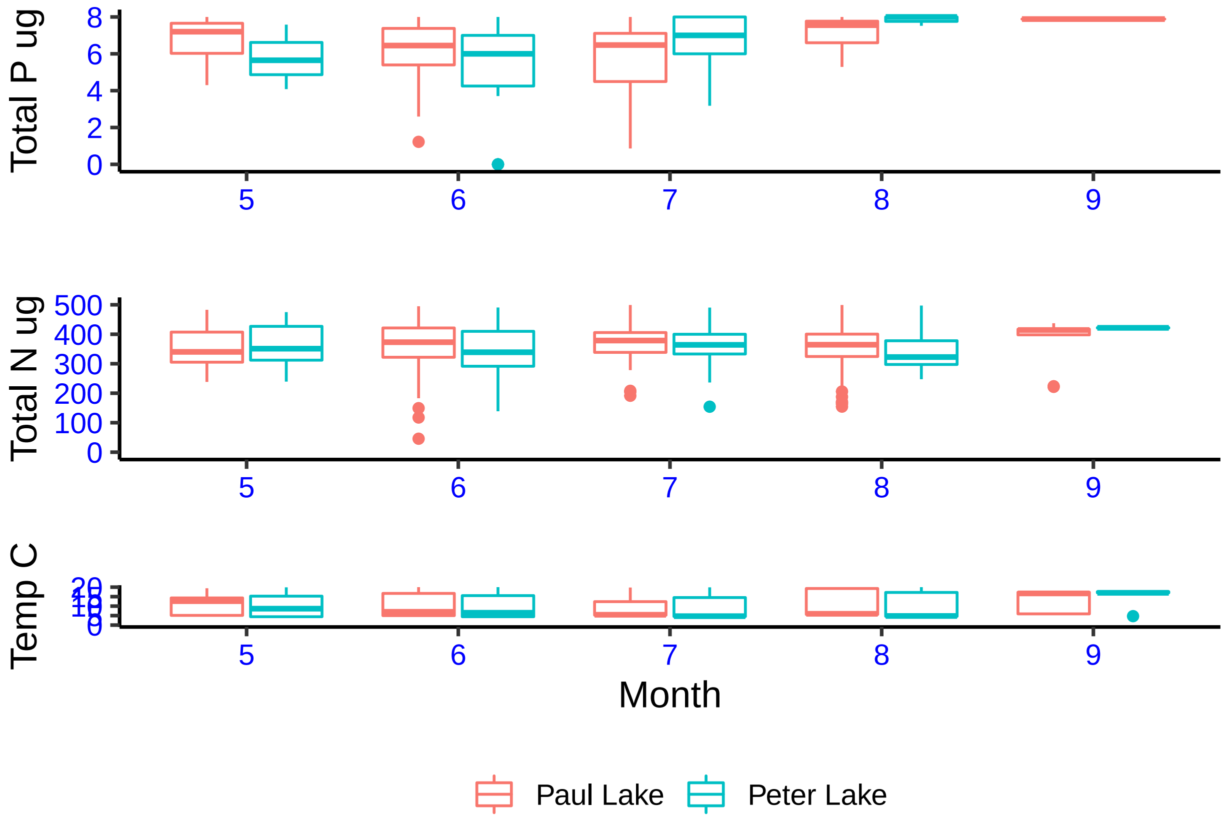

```
TPplot <- ggplot(NT_LTER_Chemistry_PPLakes, aes(x = as.factor(month), y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  theme(legend.position = "bottom") +
      labs(x = "Month", y = "Total Phosphorous ug", color = "")
print(TPplot)
```




```
tempplot2 <- ggplot(NT_LTER_Chemistry_PPLakes, aes(x = as.factor(month), y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  labs(x = "Month", y = "Temp C", color = "")+
  theme(legend.position = "bottom") +
ylim(0,20)

TNplot2 <- ggplot(NT_LTER_Chemistry_PPLakes, aes(x = as.factor(month), y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  theme(legend.position="none") +
    labs(x = "", y = "Total N ug", color = "") +
  ylim(0, 500)

TPplot2 <- ggplot(NT_LTER_Chemistry_PPLakes, aes (x = as.factor(month), y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
      theme(legend.position = "none") +
      labs(x = "", y = "Total P ug", color = "") +
        ylim(0, 8)

TempTPTNCowplot <- plot_grid(TPplot2, TNplot2, tempplot2, align = 'v', ncol = 1)
```

```
## Warning: Removed 947 rows containing non-finite values (stat_boxplot).

## Warning: Removed 600 rows containing non-finite values (stat_boxplot).

## Warning: Removed 758 rows containing non-finite values (stat_boxplot).
```

```
print(TempTPTNCowplot)
```



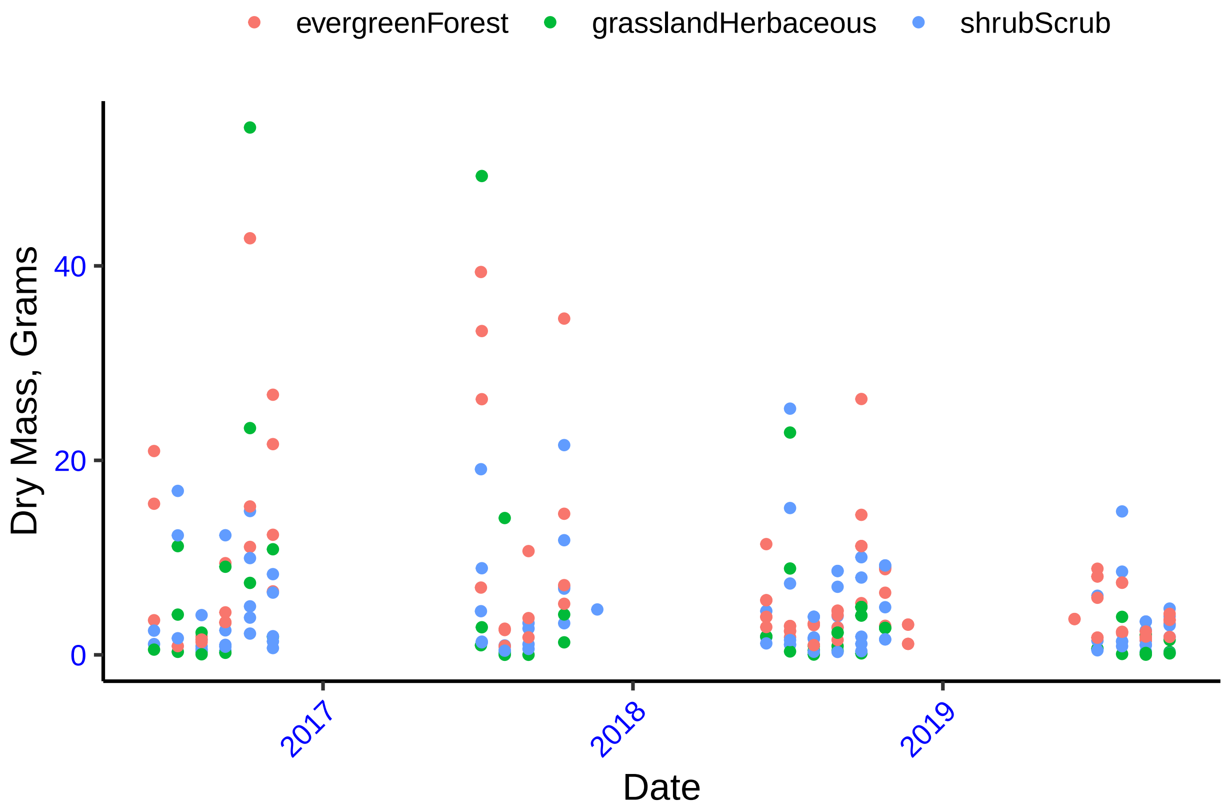
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Paul Lake experiences Nitrogen and Phosphorous levels are only observed during the months from May to October when the lake is not frozen. Peter Lake experiences higher slightly total nitrogen levels than Paul Lake. Paul Lake generally experiences slightly higher total phosphorous levels than Peter Lake.

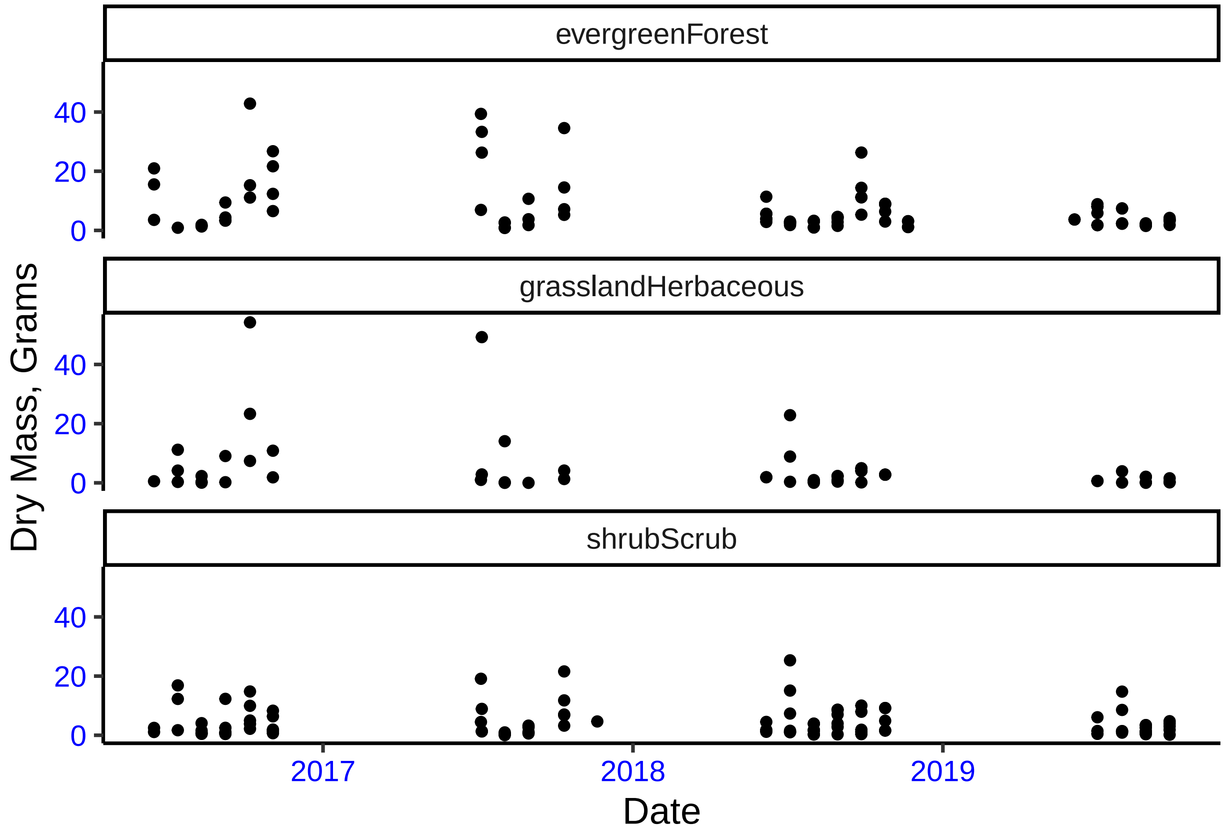
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
NeedlesPlot <-
  ggplot(subset(NiwotRidge_Litter_Processed, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  labs(x = "Date", y = "Dry Mass, Grams", color = "") +
    theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  geom_point()
print(NeedlesPlot)
```




```
NeedlesPlotFaceted <-
  ggplot(subset(NiwotRidge_Litter_Processed, functionalGroup == "Needles")) +
    aes(x = collectDate, y = dryMass) +
  labs(x = "Date", y = "Dry Mass, Grams ", color = "") +
  geom_point() +
 facet_wrap(vars(nlcdClass), nrow = 3)
print(NeedlesPlotFaceted)
```



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

Answer: I think that plot 7 is a more effective visualization of needle dry mass of different functional groups. It's eaiser to compare relative grouping weights and and spread between the functonal groups in the faceted plot than in 7 than in 6. Because the data points from the three different functional groups share many of the same values, it is hard to pull trends from plot 6.