Assignment 5: Data Visualization

Student Name

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() #checking working directory
```

```
## [1] "/Users/Marisa/Documents/Environmental data analytics/Environmental_Data_Analytics_2022/Assignme
```

```
library(tidyverse) #loading 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()
```

```r
library(cowplot) #loading cowplot
#uploading both data sets

PeterPaul.chem.nutrients <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Proc

Niwotridge<- read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",  stringsAsFactors =

#2
#checking date class
class(PeterPaul.chem.nutrients$sampledate)
```

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

```r
class(Niwotridge$collectDate)
```

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

```r
#changing date from a factor to date
View(Niwotridge)
PeterPaul.chem.nutrients$sampledate <- as.Date(PeterPaul.chem.nutrients$sampledate, format = "%Y-%m-%d"
Niwotridge$collectDate<- as.Date(Niwotridge$collectDate, format = "%Y-%m-%d")
#checking class
class(PeterPaul.chem.nutrients$sampledate)
```

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

```r
class(Niwotridge$collectDate)
```

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

Define your theme

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

```r
#3

#building my theme
mytheme <- theme_bw(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
```

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.

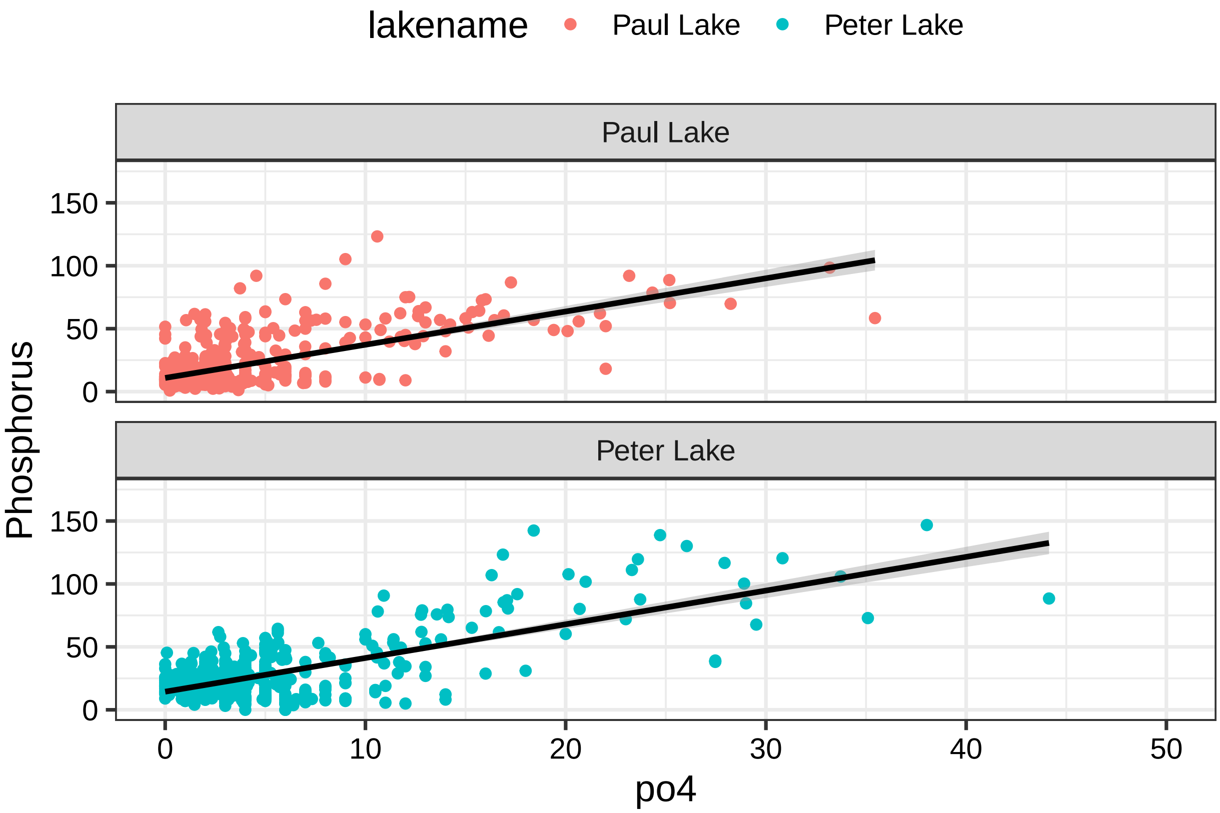
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and `ylim()`).

```
#4
View(PeterPaul.chem.nutrients)
PeterPaulplot.4<-
  ggplot(PeterPaul.chem.nutrients, aes(x = po4, y = tp_ug, color = lakename)) + #plotting phosphorus by
    geom_point()+ #making graph a scatterplot
   geom_smooth(method=lm, color = "black")+ #adding black line
  ylab("Phosphorus") + #Labelling y axis
ylim(0,175)+ #setting y and x limits
xlim(0,50)+
  facet_wrap(vars(lakename), nrow = 2)+ #separating by lakename
 mytheme #adding my theme
print(PeterPaulplot.4)
```

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

## Warning: Removed 21948 rows containing non-finite values (stat_smooth).

## Warning: Removed 21948 rows containing missing values (geom_point).
```

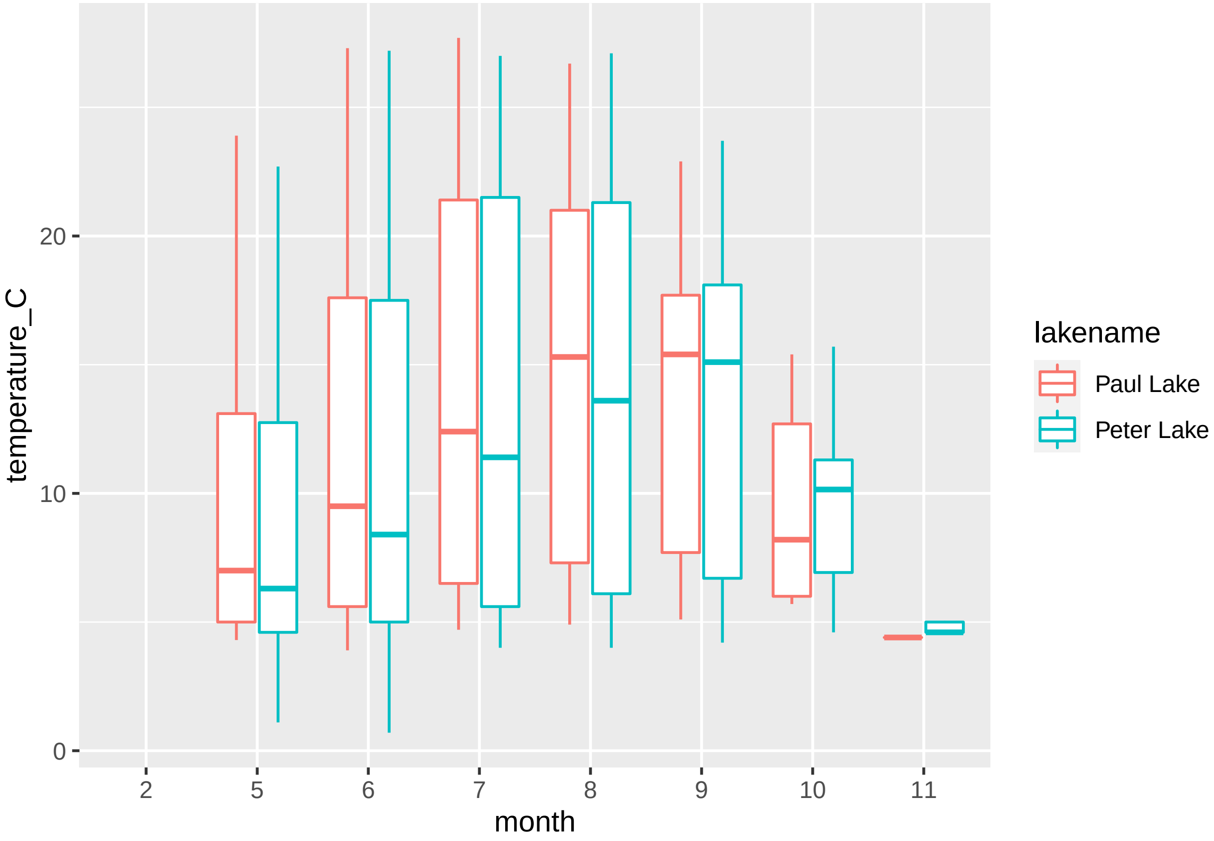


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

```
#5
#creating temperature box plot
```

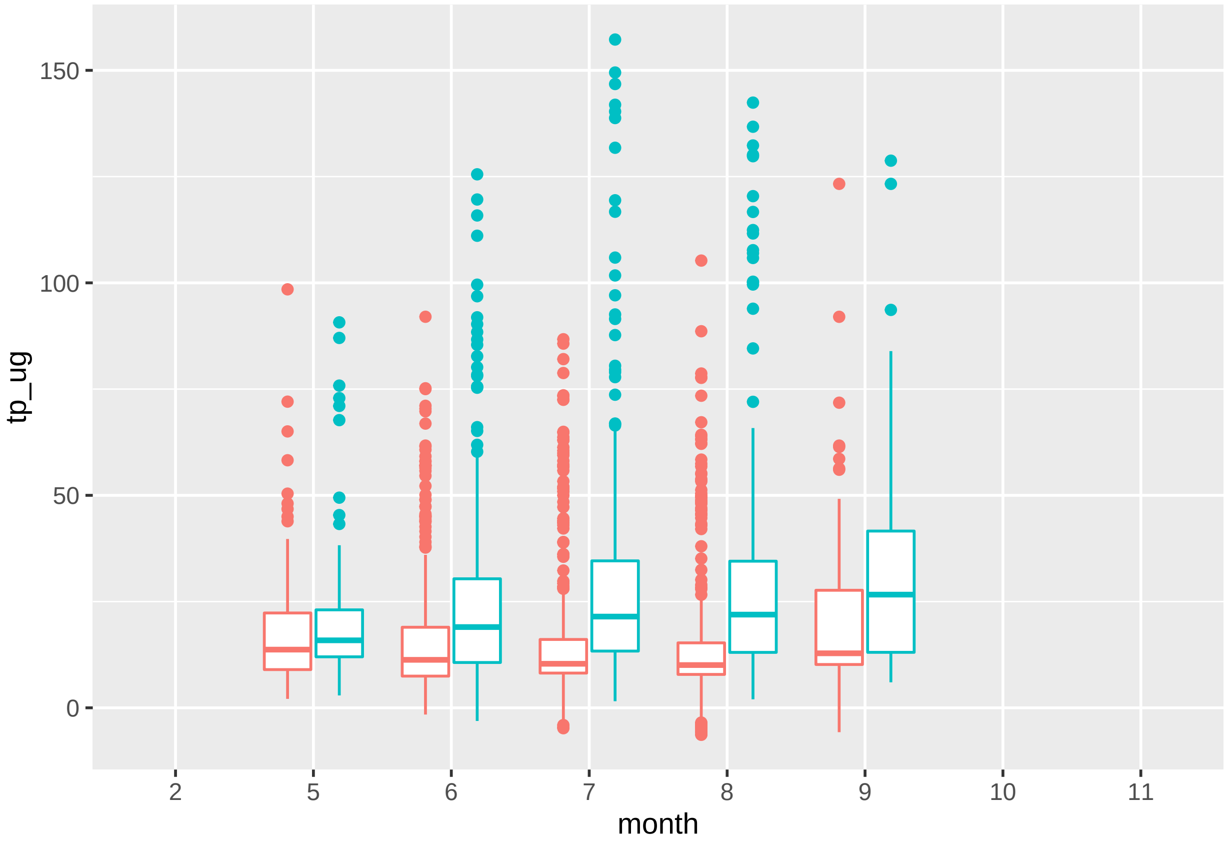
```
PeterPaulplot.5a <-
  ggplot(PeterPaul.chem.nutrients, aes(x = as.factor(month), y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  xlab("month")
print(PeterPaulplot.5a)
```

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



```
#creating TP box plot
PeterPaulplot.5b <-
  ggplot(PeterPaul.chem.nutrients, aes(x = as.factor(month), y = tp_ug)) +
  geom_boxplot(aes(color = lakename))+
  xlab("month")+
 theme(legend.position = "none")
print(PeterPaulplot.5b)
```

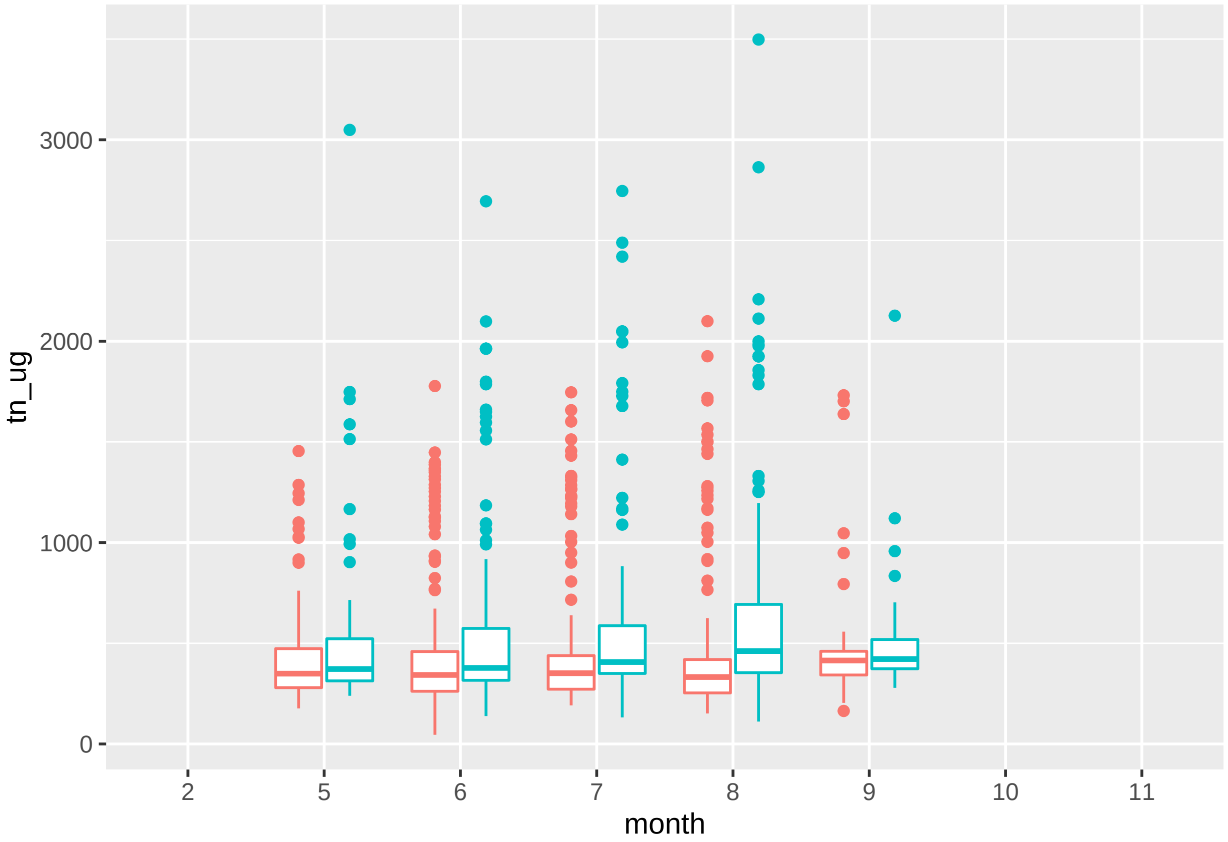
```
## Warning: Removed 20729 rows containing non-finite values (stat_boxplot).
```



```
#creating TN box plot
PeterPaulplot.5c <-
  ggplot(PeterPaul.chem.nutrients, aes(x = as.factor(month), y = tn_ug)) +
  geom_boxplot(aes(color = lakename))+
  xlab("month")+
  theme(legend.position = "none")

print(PeterPaulplot.5c)
```

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



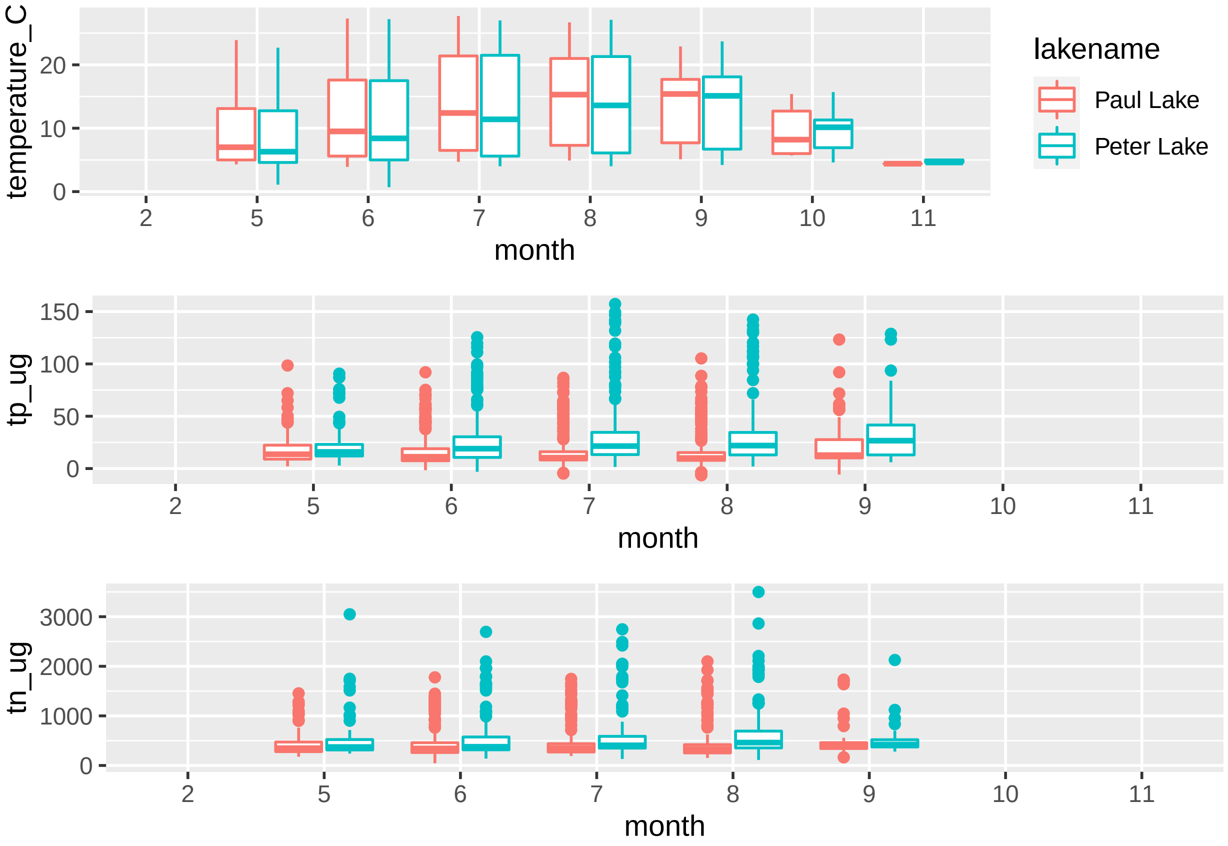
```
#combining plots

plot_grid(PeterPaulplot.5a, PeterPaulplot.5b, PeterPaulplot.5c, nrow = 3, align = 'h')
```

```
## 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).
```



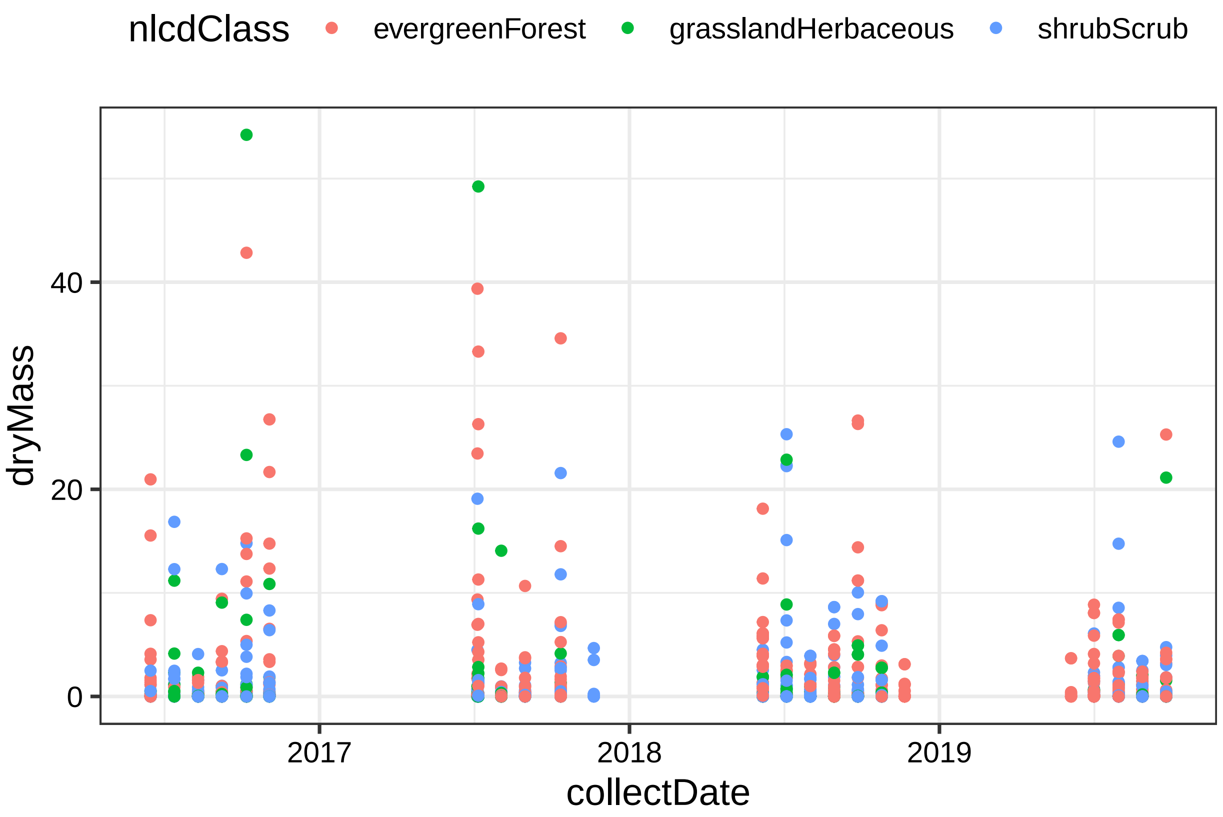
Question: What do you observe about the variables of interest over seasons and between lakes?

Answer:

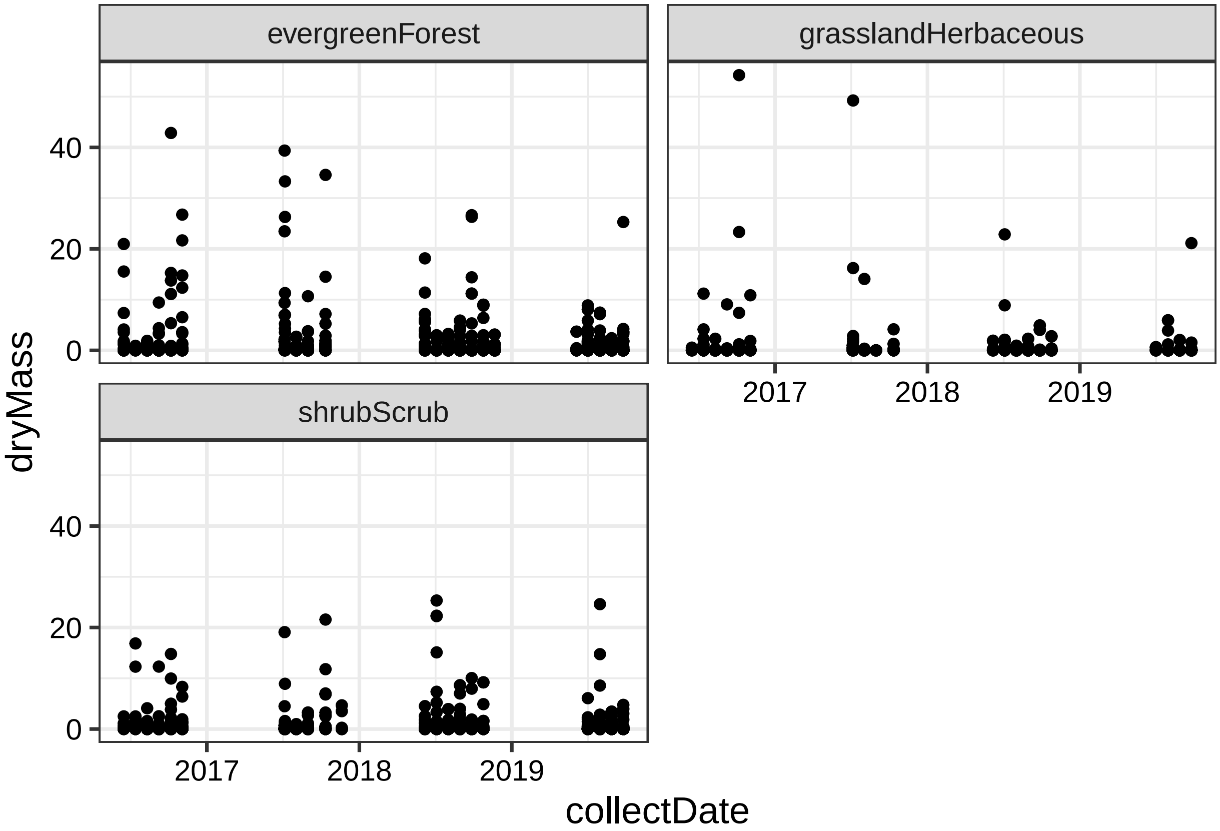
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
#6
#filtering for only needles functional group
Niwotridge.needles<- filter(Niwotridge, functionalGroup == "Needles")
#creating scatterplot of dry mass by date
Niwotridgeplot.6<- ggplot(Niwotridge, aes(x = collectDate, y = dryMass, color = nlcdClass)) +
geom_point()+
mytheme
print(Niwotridgeplot.6)
```



```
#7
Niwotridgeplot.7<- ggplot(Niwotridge, aes(x = collectDate, y = dryMass))+
facet_wrap(vars(nlcdClass), nrow = 2) +
geom_point()+
  mytheme
print(Niwotridgeplot.7)
```



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

> Answer:Plot 77 is more effective. In plot 6, the points are too close together to distinguish the different colors well. With plot 7, we can see differences each of the ncld classes more clearly since their points are not clustered together.