Assignment 5: Data Visualization

Elise Boos

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
#check working directory
getwd()
```

```
## [1] "/Users/elise/Desktop/Data_Analytics/Environmental_Data_Analytics_2022/Assignments"
```

```
#load packages
require(tidyverse)
require(cowplot)

#load data
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
#check date
class(NTL$sampledate)
```

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

```
class(NEON$collectDate)
```

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

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

#check date again
class(NTL$sampledate)
```

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

```
class(NEON$collectDate)
```

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

Define your theme

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

```
#3
#default theme
mytheme <- theme_classic(base_size = 12) + #increase font size
  theme(axis.text = element_text(color = "black"), #change color of text to black
        legend.position = "top")
#set as default
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(NTL, aes(x = tp_ug, y = po4, color = lakename)) +
ylab("Phosphate")+ #rename axis
xlab("Total Phosphorus")+
```

```
geom_point(aes(),alpha = 0.5)+
xlim(0,75)+ #hide extremes by setting limits
ylim(0,20)+
geom_smooth(method = lm, color = "black") #line of best fit
```

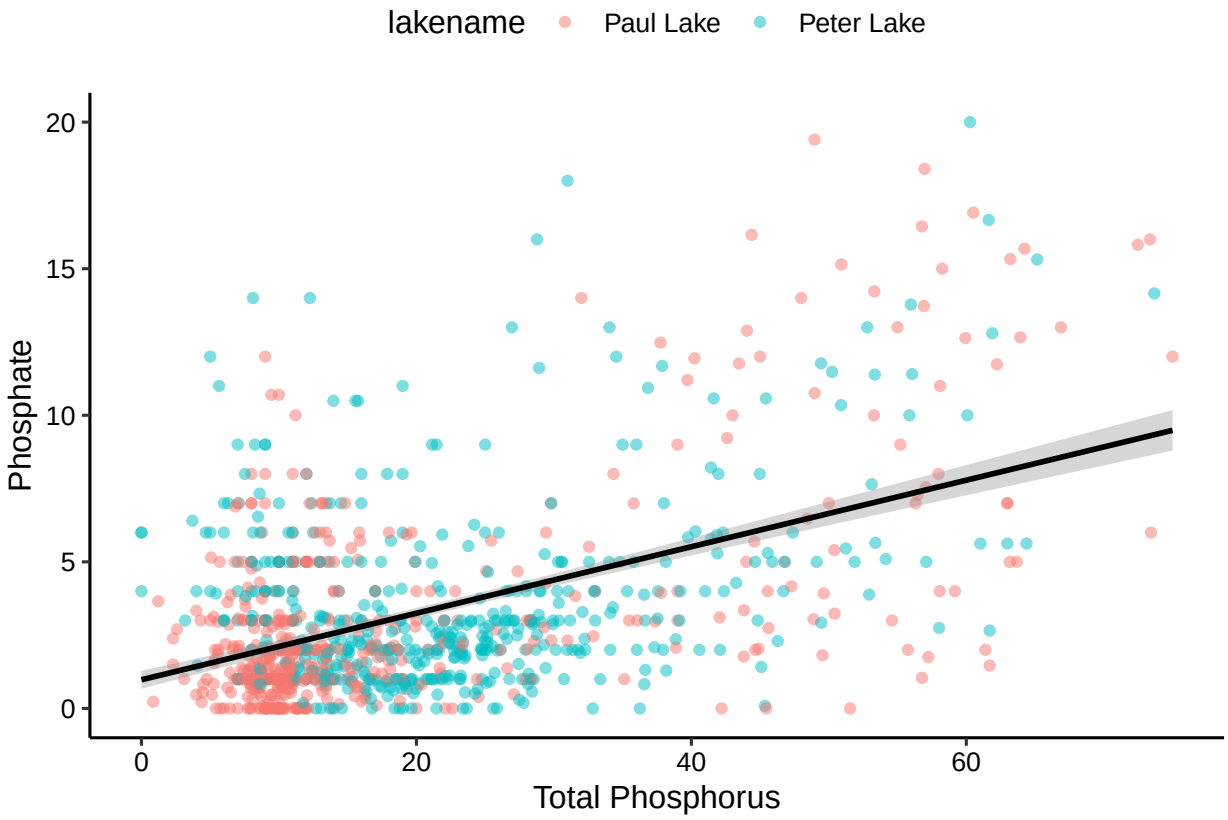
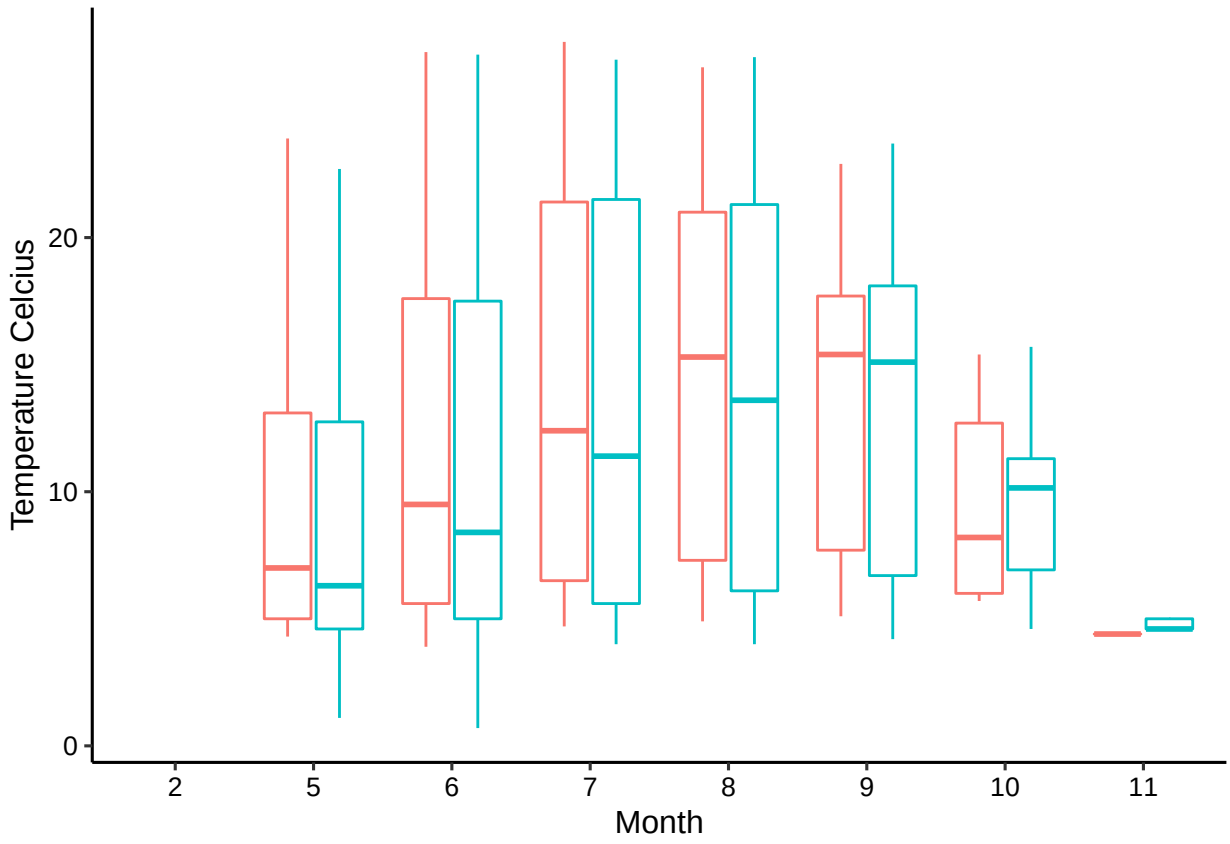


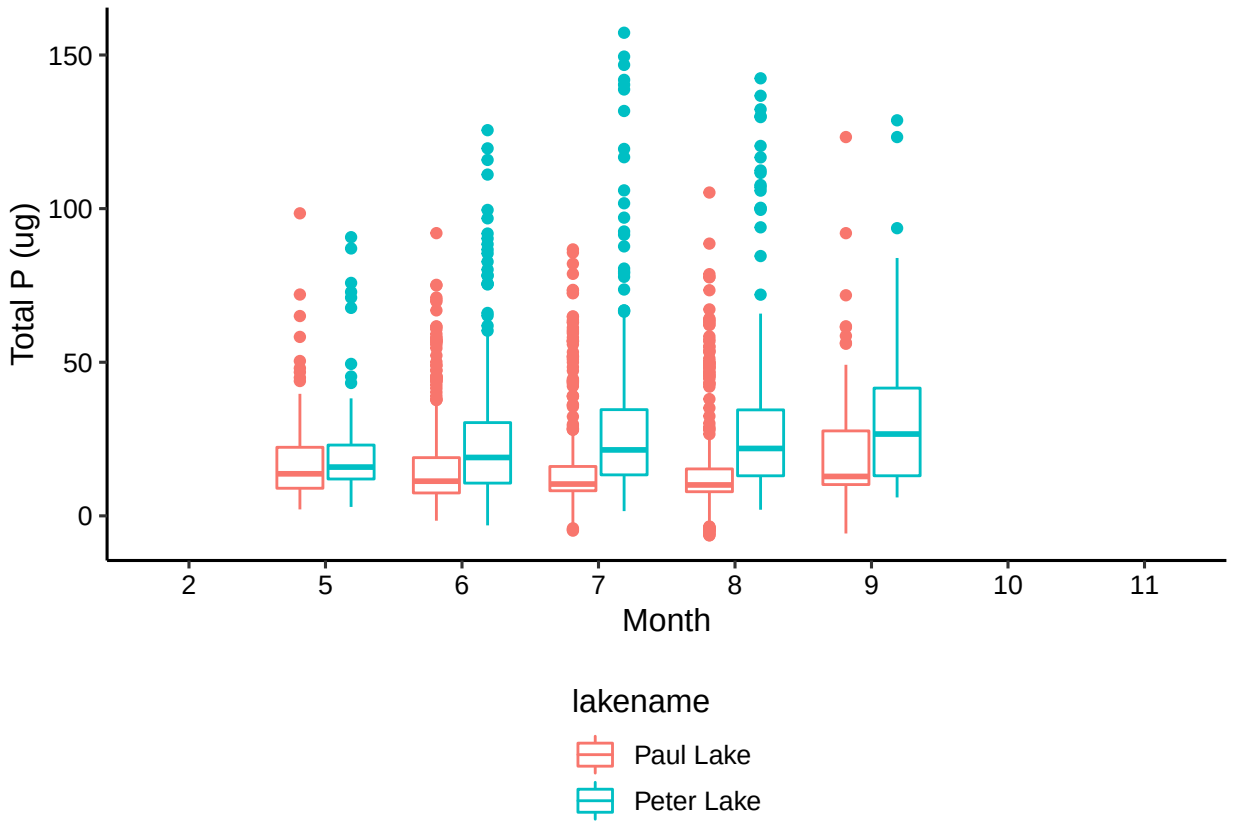
Figure 1: Total phosphorus by phosphate in Peter and Paul lakes

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.

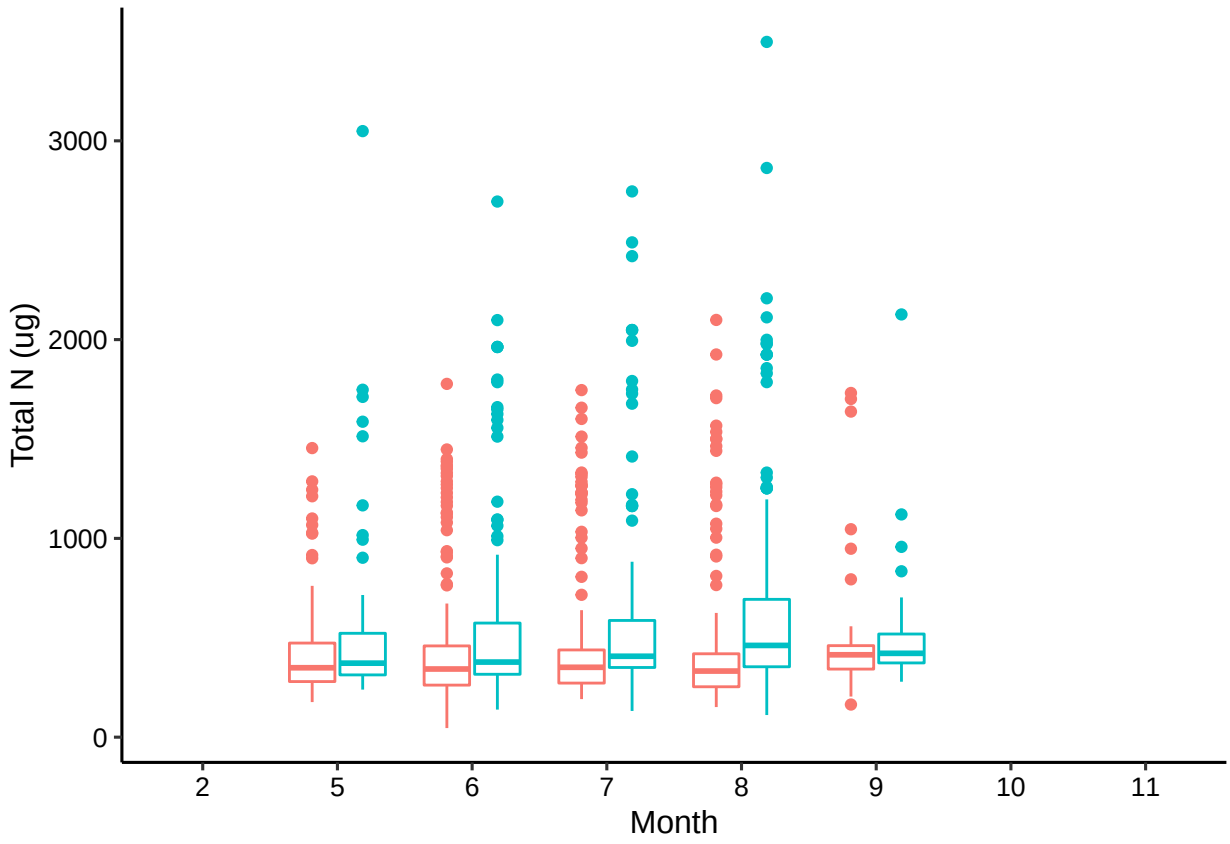
```
#5
#creating the components of the cowplot, temperature, TP and TN
#temp
NTL_temp <-
  ggplot(NTL, aes(x = as.factor(month), y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  ylab("Temperature Celcius")+
  xlab("Month")+
  theme(legend.position = "none") #remove legend so only one in final graph
print(NTL_temp)
```



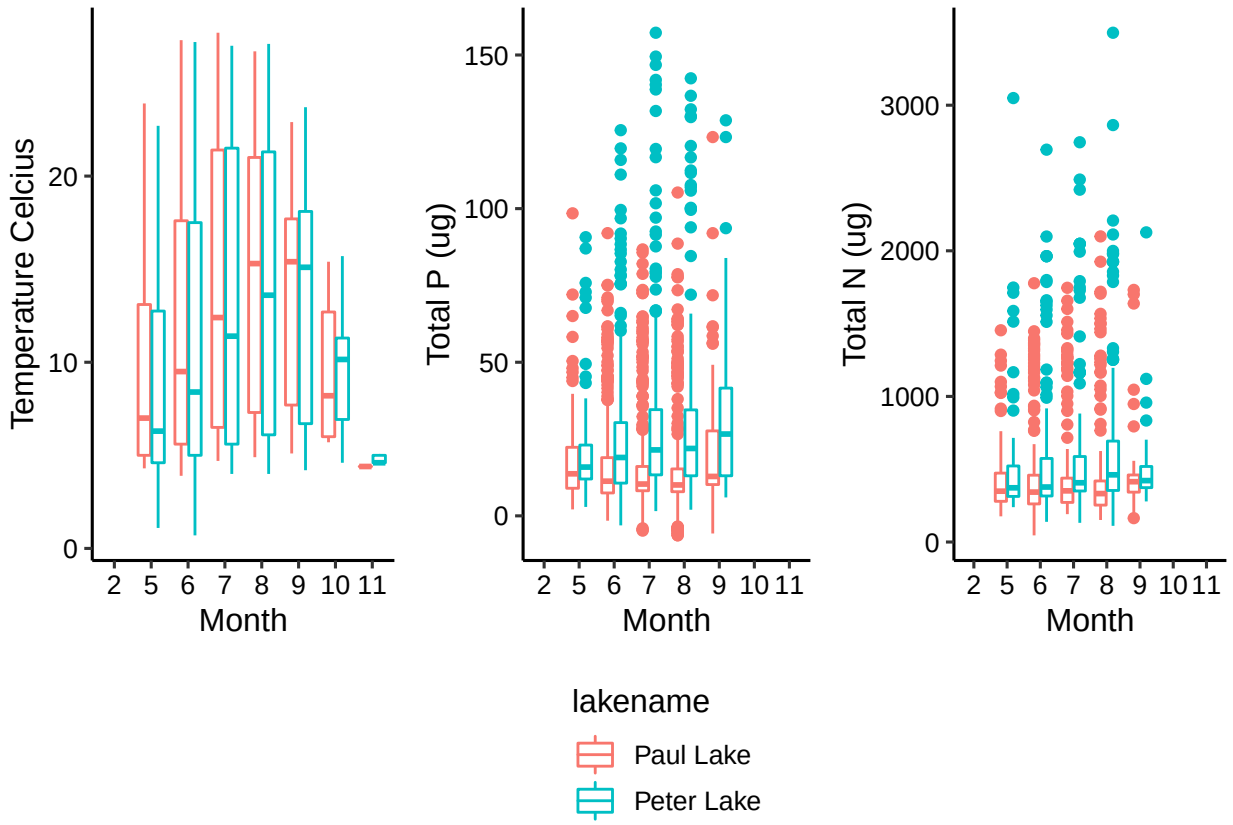

```
#TP
NTL_TP <-
  ggplot(NTL, aes(x = as.factor(month), y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  ylab("Total P (ug)")+
  xlab("Month") +
  theme(legend.position = "bottom", legend.direction = "vertical")
#have middle graph be the only one with legend
print(NTL_TP)
```




```
#TN
NTL_TN <-
  ggplot(NTL, aes(x = as.factor(month), y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  ylab("Total N (ug)")+
  xlab("Month")+
  theme(legend.position = "none") #remove legend so only one final graph
print(NTL_TN)
```




```
#put all together
plot_grid(NTL_temp, NTL_TP, NTL_TN,  nrow = 1, align = 'h',
          rel_heights = c(1, 1, 1), axis = "b")
```



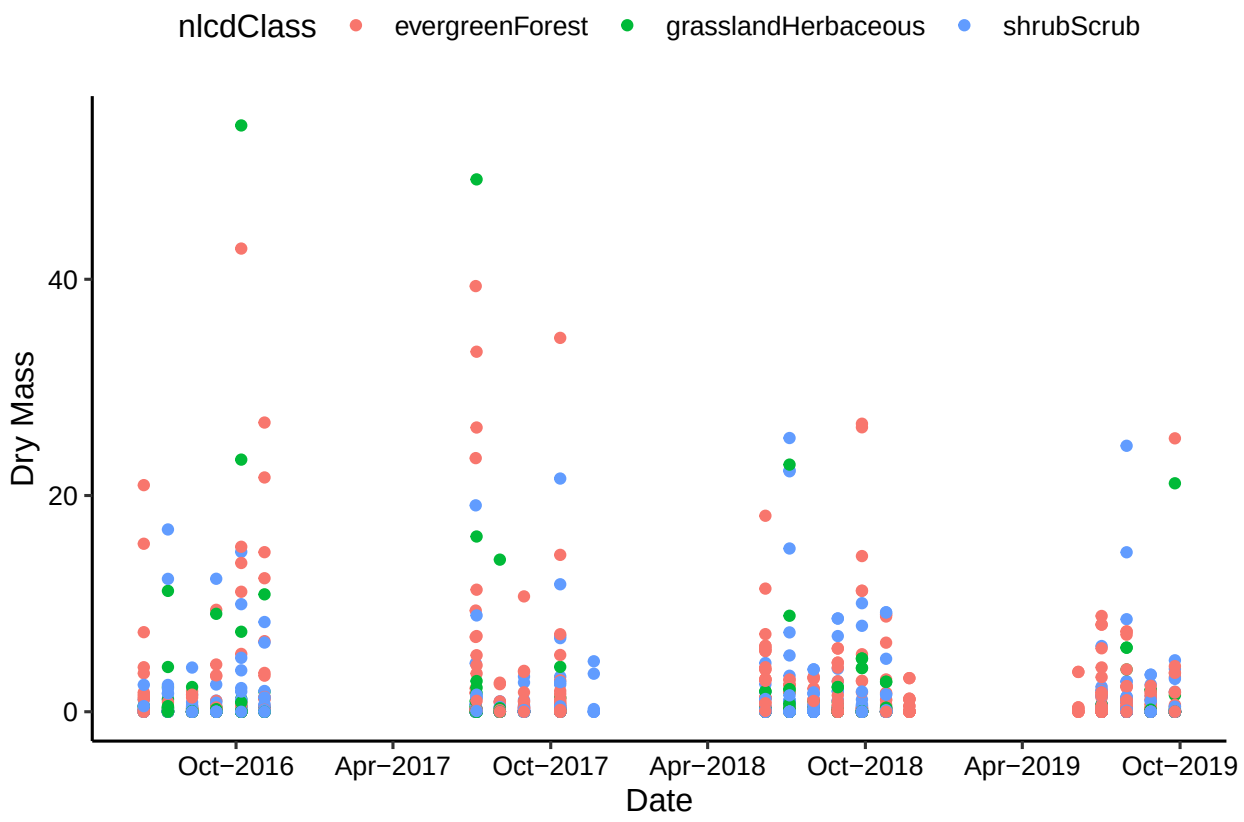
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: There is higher total phosporous and nitrogen in Peter lake compared to Paul Lake throughout the months. Summer months have higher temperatures and higher concentrations of nutrients in both lakes.

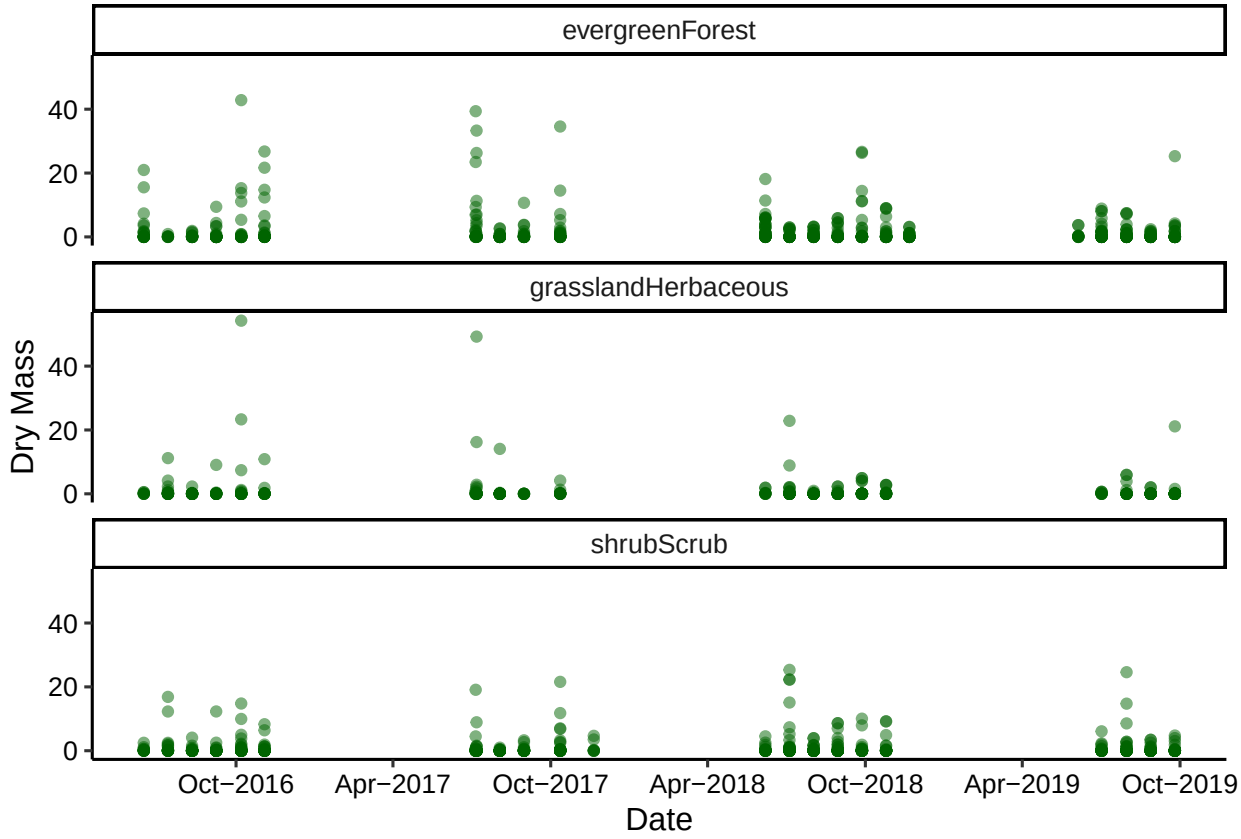
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
#6
#First graph using color to differentiate
ggplot(subset(NEON, functionalGroup = "Needles"), aes(x = collectDate, y = dryMass)) +
geom_point(aes(color = nlcdClass)) +
ylab("Dry Mass")+
xlab("Date")+
scale_x_date(date_breaks = "6 months", date_labels = "%b-%Y")
```




```
#7
#second graph using a facet to separate
ggplot(subset(NEON, functionalGroup = "Needles"), aes(x = collectDate, y = dryMass)) +
geom_point(color = "dark green", alpha = 0.5) +
facet_wrap(vars(nlcdClass), nrow = 3) +
ylab("Dry Mass")+
xlab("Date")+
scale_x_date(date_breaks = "6 months", date_labels = "%b-%Y")
```



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

Answer: Plot 6 where the different classes are colored differently because its easier to visualize the comparisons between each class on a given day.