Assignment 5: Data Visualization

Stephanie Kinser

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] "C:/Users/skins/Documents/EDA/Environmental_Data_Analytics_2022/Assignments"
```

```
#install.packages("tidyverse")
#install.packages("cowplot")
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)

NTL_LTER <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")

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

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

Litter$collectDate <- as.Date(Litter$collectDate, format = "%Y-%m-%d")

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

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

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

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

Define your theme

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

```
#3
A5_theme <- theme_light(base_size = 12)+
   theme(axis.text = element_text(color = "black"),
         legend.position = "right", panel.grid.minor = element_blank())

theme_set(A5_theme)
```

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.

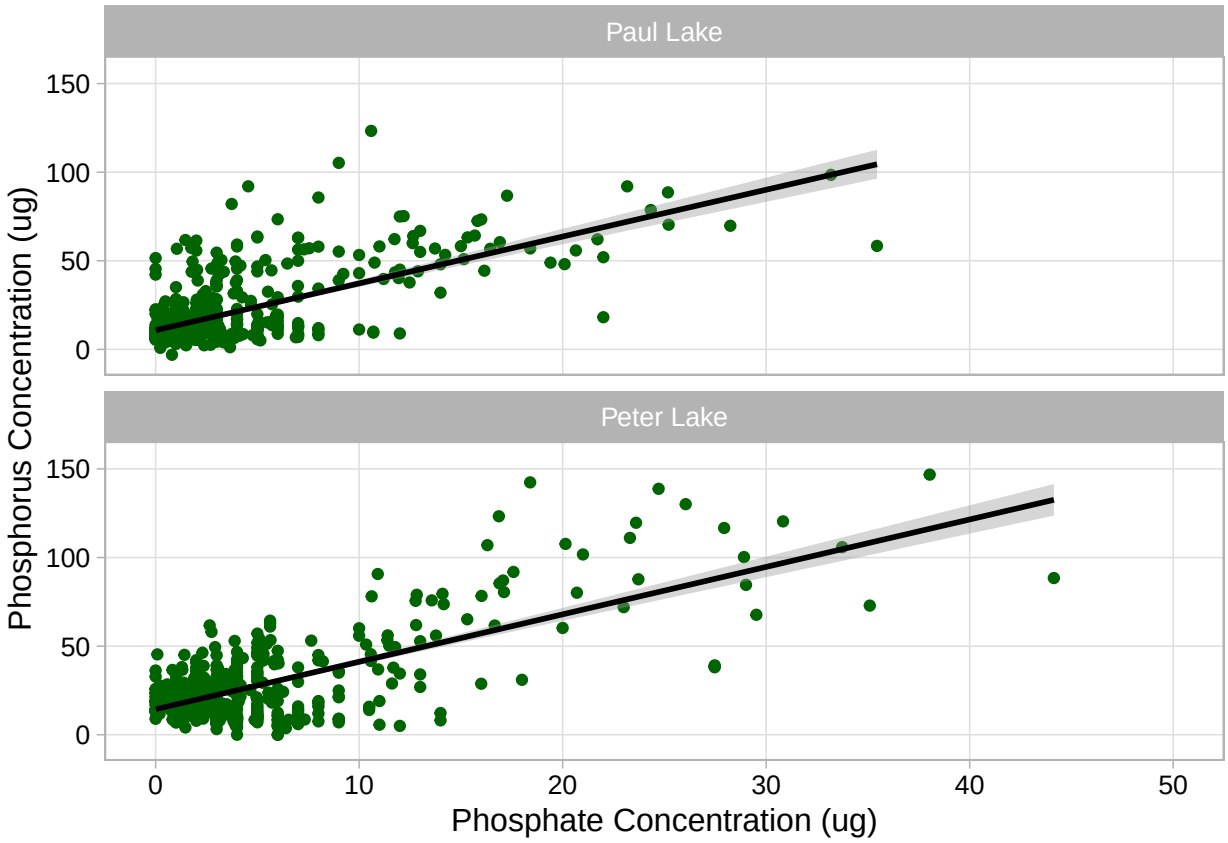
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and `ylim()`).

```
#4
#install.packages("ggridges")
library(ggridges)

plot4 <- ggplot(NTL_LTER, aes(x = po4, y = tp_ug)) +
```

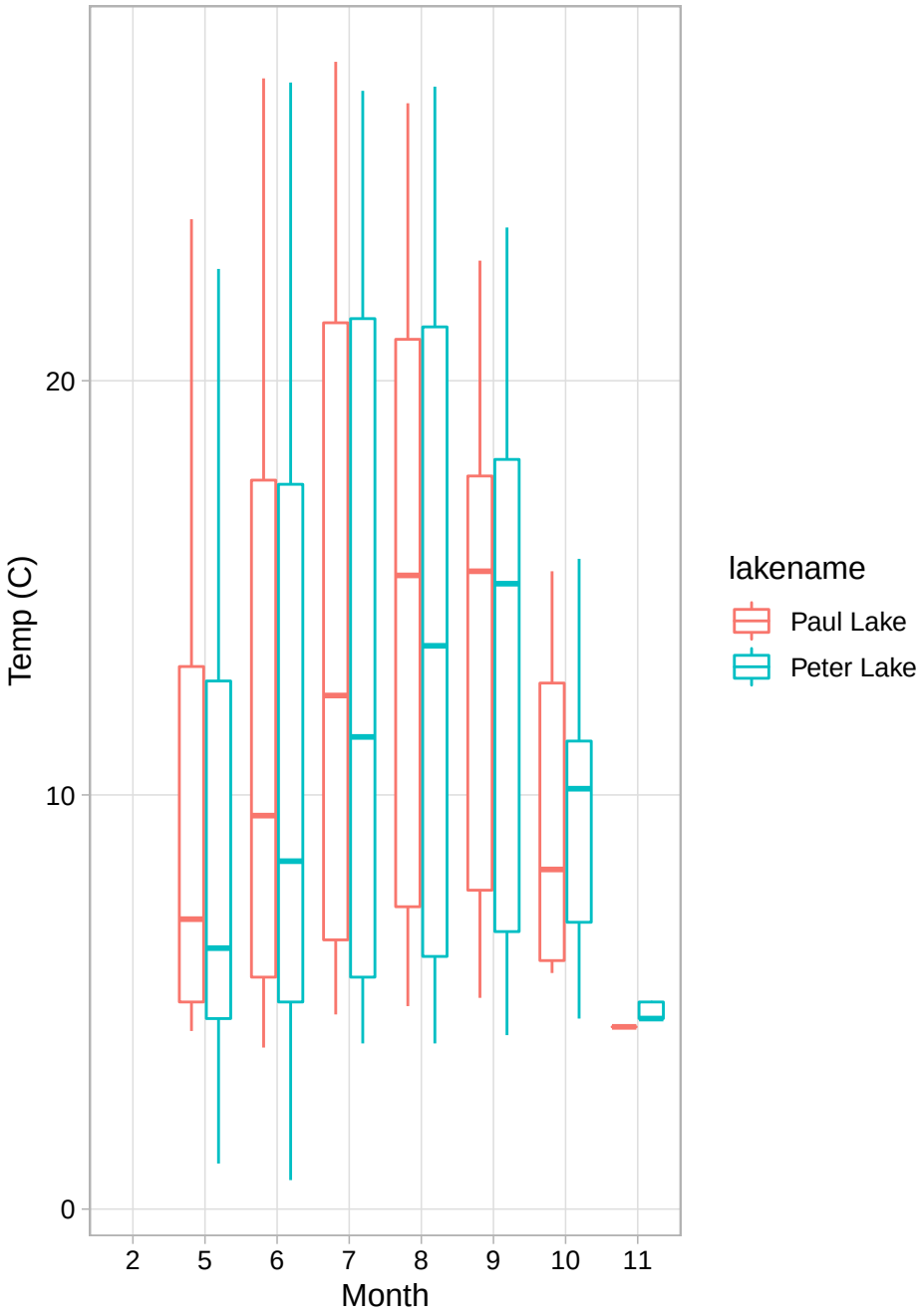
```
  geom_point(color = "darkgreen")+
  facet_wrap(vars(lakename), nrow = 2)+
  xlim(0,50)+
  geom_smooth(method = lm, color = "black")+
  labs(x = "Phosphate Concentration (ug)", y = "Phosphorus Concentration (ug)")
print(plot4)
```

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

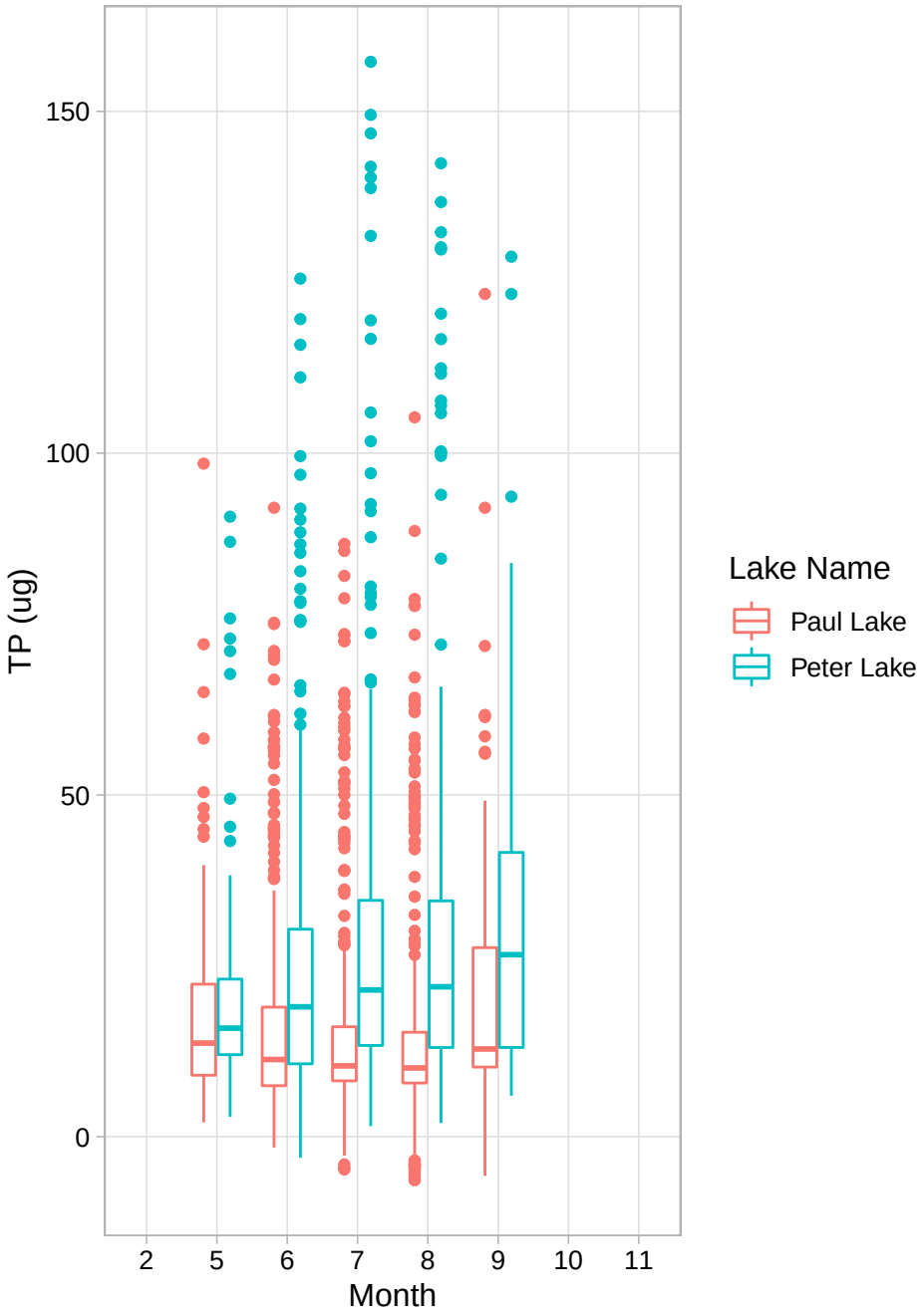


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

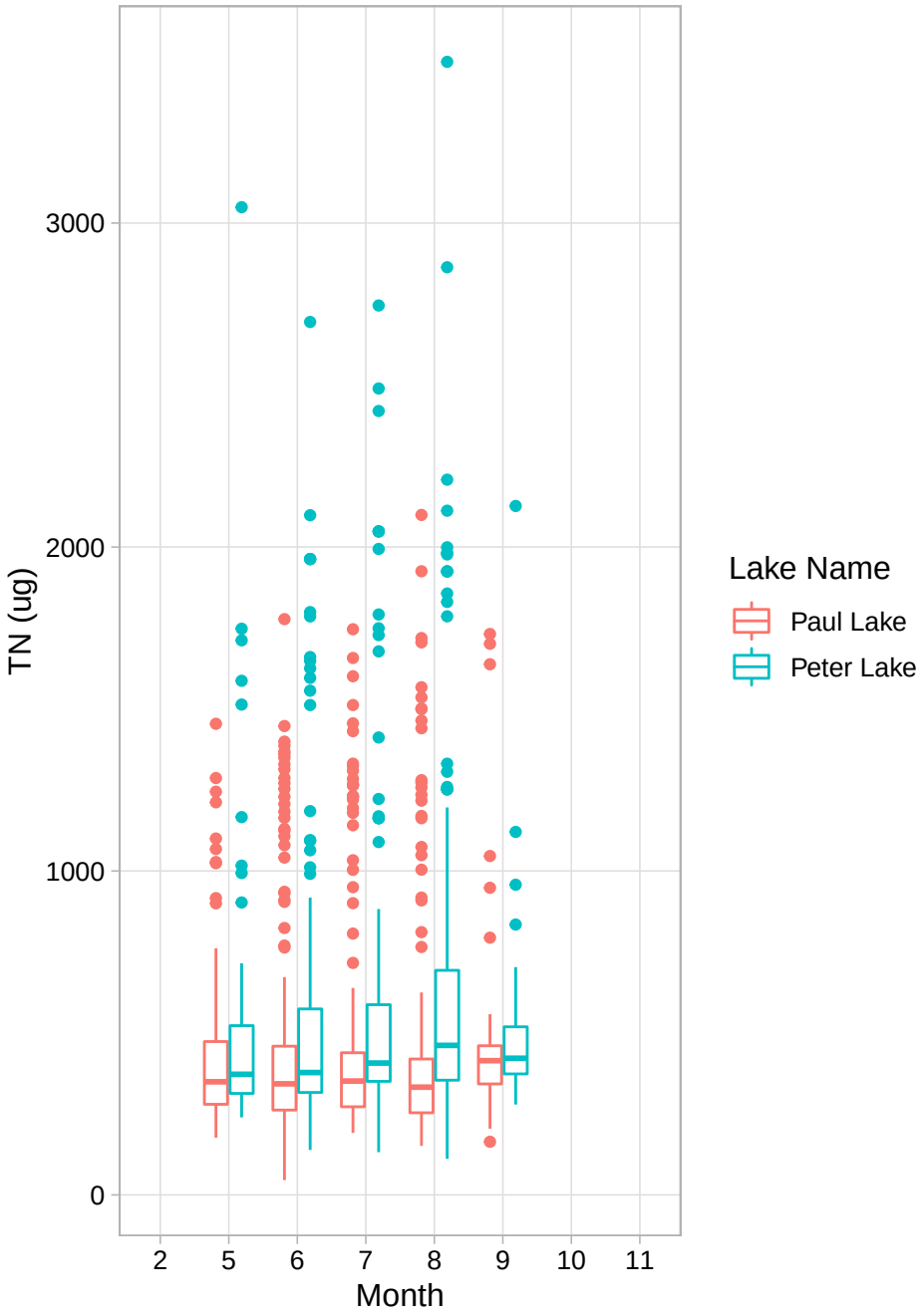
```
#5
NTL_temp <- ggplot(NTL_LTER)+
  geom_boxplot(aes(x = as.factor(month), y = temperature_C, color = lakename))+
  labs(x = "Month", y = "Temp (C)")
print(NTL_temp)
```




```
#What is TP and TN
NTL_TP <- ggplot(NTL_LTER)+
  geom_boxplot(aes(x = as.factor(month), y = tp_ug, color = lakename))+
  labs(x = "Month", y = "TP (ug)", color = "Lake Name")
print(NTL_TP)
```



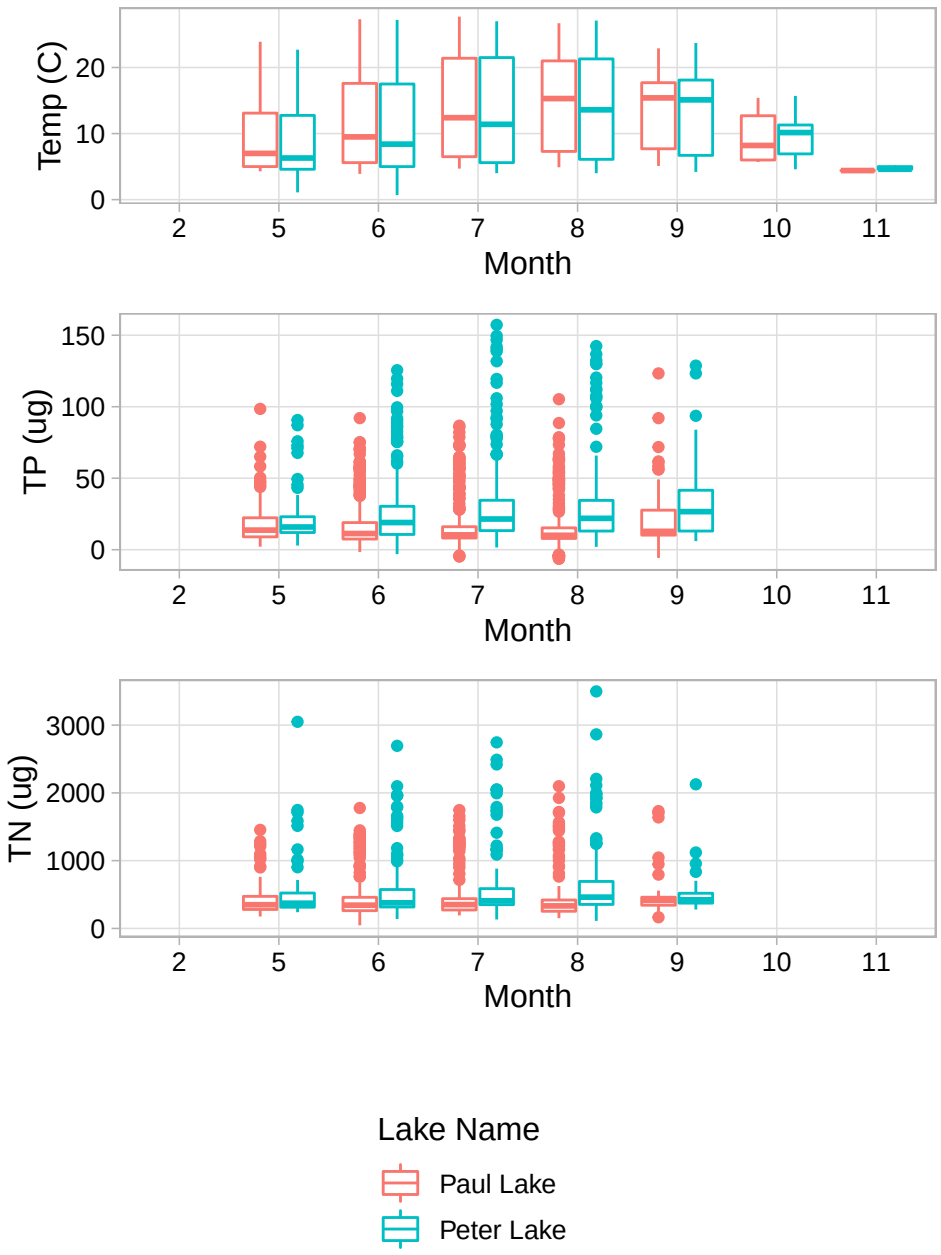

```
NTL_TN <- ggplot(NTL_LTER)+
  geom_boxplot(aes(x = as.factor(month), y = tn_ug, color = lakename))+
  labs(x = "Month", y = "TN (ug)", color = "Lake Name")
print(NTL_TN)
```




```
legend <- get_legend(NTL_TN) #pulls legend from plot to add to plot_grid


NTL_grid <- plot_grid(NTL_temp + theme(legend.position = "none"),
      NTL_TP + theme(legend.position = "none"),
      NTL_TN + theme(legend.position = "none"),
      legend,
      nrow = 4, #num of rows
      align = 'v', #align vertically
      axis = 'l',
      rel_heights = c(1.25, 1.5, 1.5)) #align along left axis
```

```
print(NTL_grid)
```



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

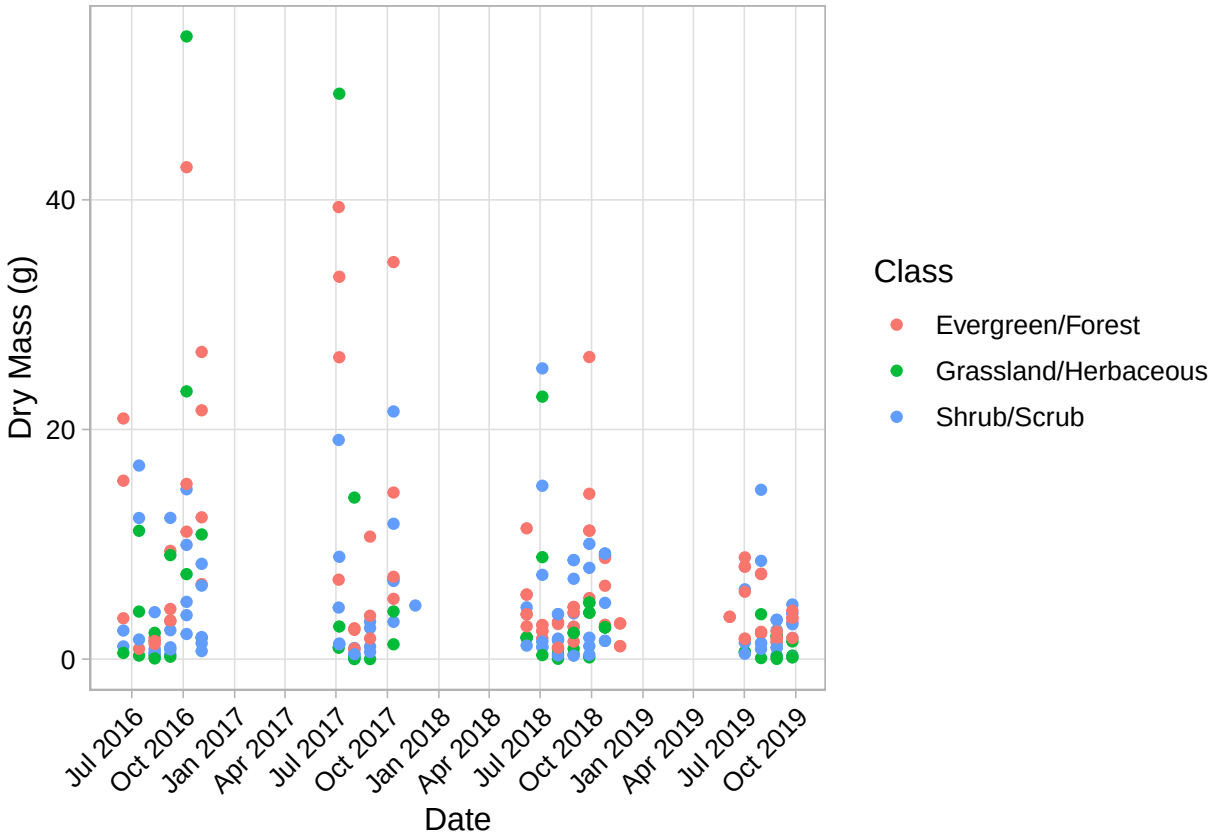
> Answer: For each month that has data, nitrogen and phosphorous concentrations are higher for Peter Lake during all seasons. The phosphorous and nitrogen concentrations tend to have more outliers in the peak summer months. Temperature is the most variable across seasons, and temperature at Peter Lake tends to be more variable than for Paul Lake across seasons.

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.

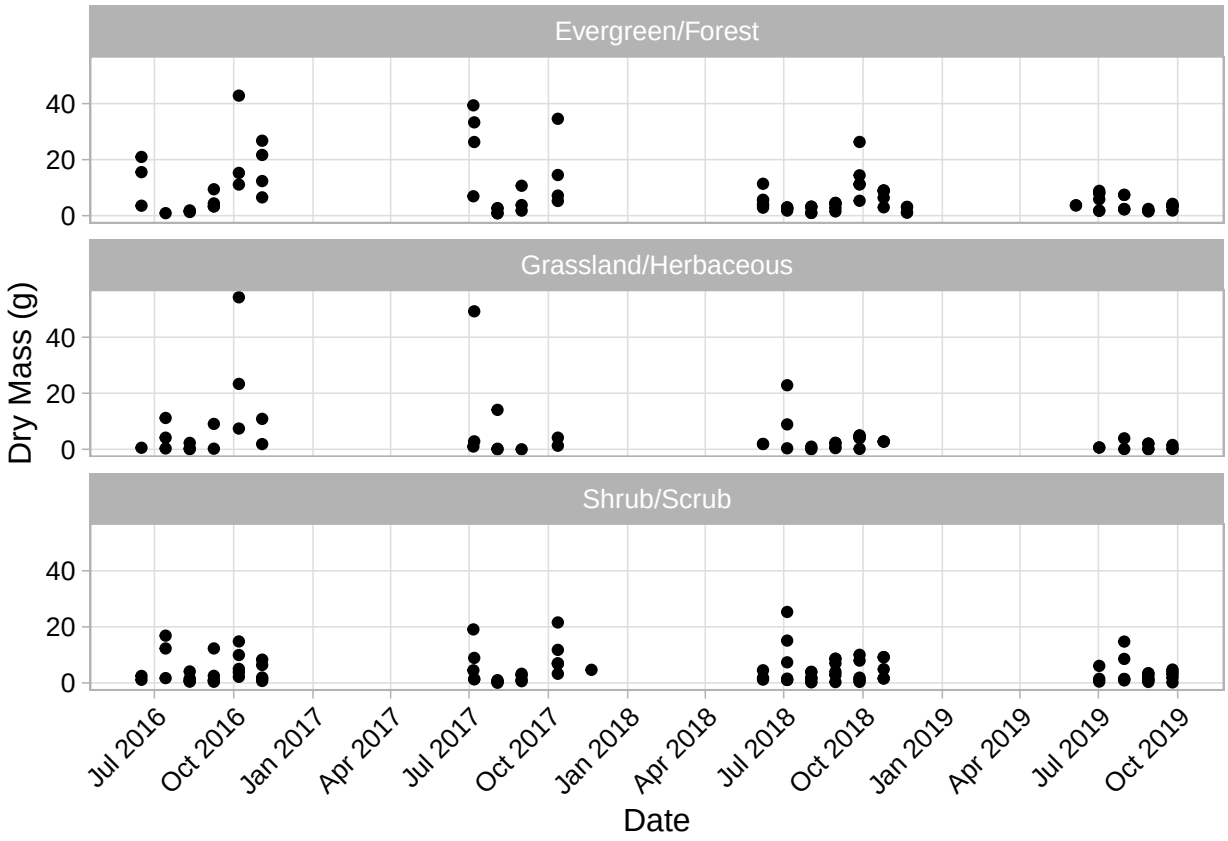
```
#6
Needles <- ggplot(subset(Litter, functionalGroup =="Needles"))+
  geom_point(aes(x = collectDate, y = dryMass, color = nlcdClass))+
  scale_x_date(date_breaks = "3 months", date_labels = "%b %Y")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))+
  labs(x = "Date", y = "Dry Mass (g)", color = "Class")+
  scale_color_hue(labels = c("Evergreen/Forest", "Grassland/Herbaceous", "Shrub/Scrub")) #adjusting lab
print(Needles)
```




```
#7
#creates labels for facets - applied with labeller
ncld_labels <- as_labeller(c(evergreenForest = "Evergreen/Forest",
          grasslandHerbaceous = "Grassland/Herbaceous",
          shrubScrub = "Shrub/Scrub"))

Needles_facet <- ggplot(subset(Litter, functionalGroup =="Needles"))+
  geom_point(aes(x = collectDate, y = dryMass))+
  facet_wrap(vars(nlcdClass), nrow = 3, labeller = ncld_labels)+
  scale_x_date(date_breaks = "3 months", date_labels = "%b %Y")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))+
```

```
  labs(x = "Date", y = "Dry Mass (g)")
print(Needles_facet)
```



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

Answer: In this case, Plot 7 is more effective because you can more clearly see the individual points for each class and, thereby, compare the differences in classes more easily. In Plot 6, the points overlap in some cases making it difficult to see the range of points for a given class. Because the plots can are aligned in Plot 7, this format still makes it easy to compare across classes for a given date.