Assignment 5: Data Visualization

Courtney Horn

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 Tuesday, February 23 at 11:59 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 (both the tidy [`NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv`] and the gathered [`NTL-LTER_Lake_Nutrients_PeterPaulGathered_Processed.csv`] versions) 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.

```
library(tidyverse)
library(cowplot)
library(lubridate)
library(viridis)
library(RColorBrewer)
library(colormap)
getwd()
```

```
## [1] "/Users/courtneyannehorn/Desktop/EDA/EDAfin/Assignments"
```

```
lakenut <-read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv", string
lakenut_gath <- read.csv("../Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaulGathered_Processed.csv", s
Niwot_litter <-read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv", stringsAsFactors =



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

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

```
#gathered
#colnames(lakenut_gath)
#class(lakenut_gath$sampledate)
#head(lakenut_gath$sampledate)
lakenut_gath$sampledate <- as.Date(lakenut_gath$sampledate, format = "%Y-%m-%d")
class(lakenut_gath$sampledate)
```

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

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

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

Define your theme

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

```
mytheme <- theme_classic(base_size = 10) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")


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.

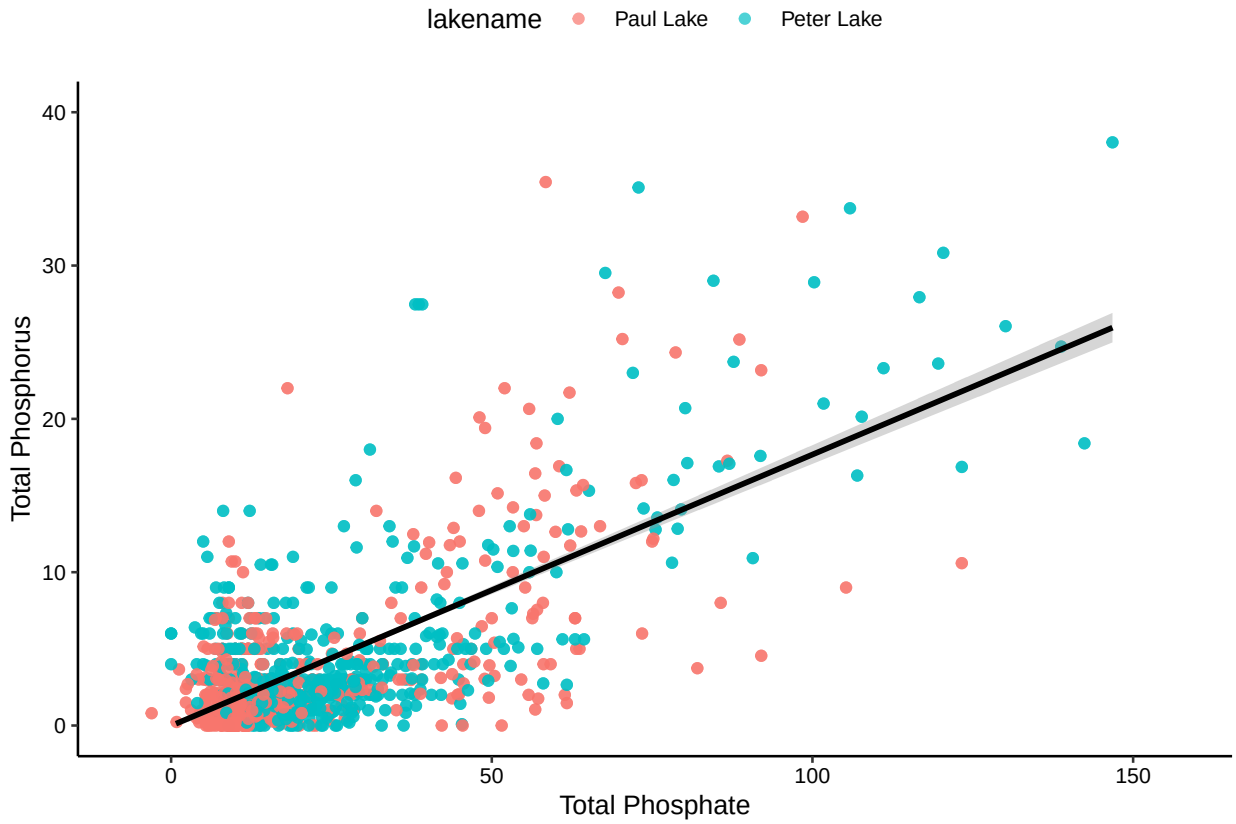
```
#colnames(lakenut)
#dim(lakenut)
#colnames(lakenut_gath)
#dim(lakenut_gath)



lakenutmod <- full_join(lakenut, lakenut_gath)
lakenutmodsub <- filter(lakenutmod, nutrient == "tp_ug" | nutrient == "po4" )
#View(lakenutmodsub)




tp_ugvspo4 <-
  ggplot(lakenutmodsub, aes(x = tp_ug, y = po4, color = lakename)) +
  geom_point(alpha = 0.7) +
    geom_smooth(method = lm, color = "black") +
```

```
  xlab("Total Phosphate") + ylab("Total Phosphorus") +
  ylim(0, 40)
  #scale_color_distiller(palette = "Blues", direction = 1)
 #scale_color_viridis(option = "magma", direction = -1)
print(tp_ugvspo4)
```

```
## Warning: Removed 1345 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1345 rows containing missing values (geom_point).
```

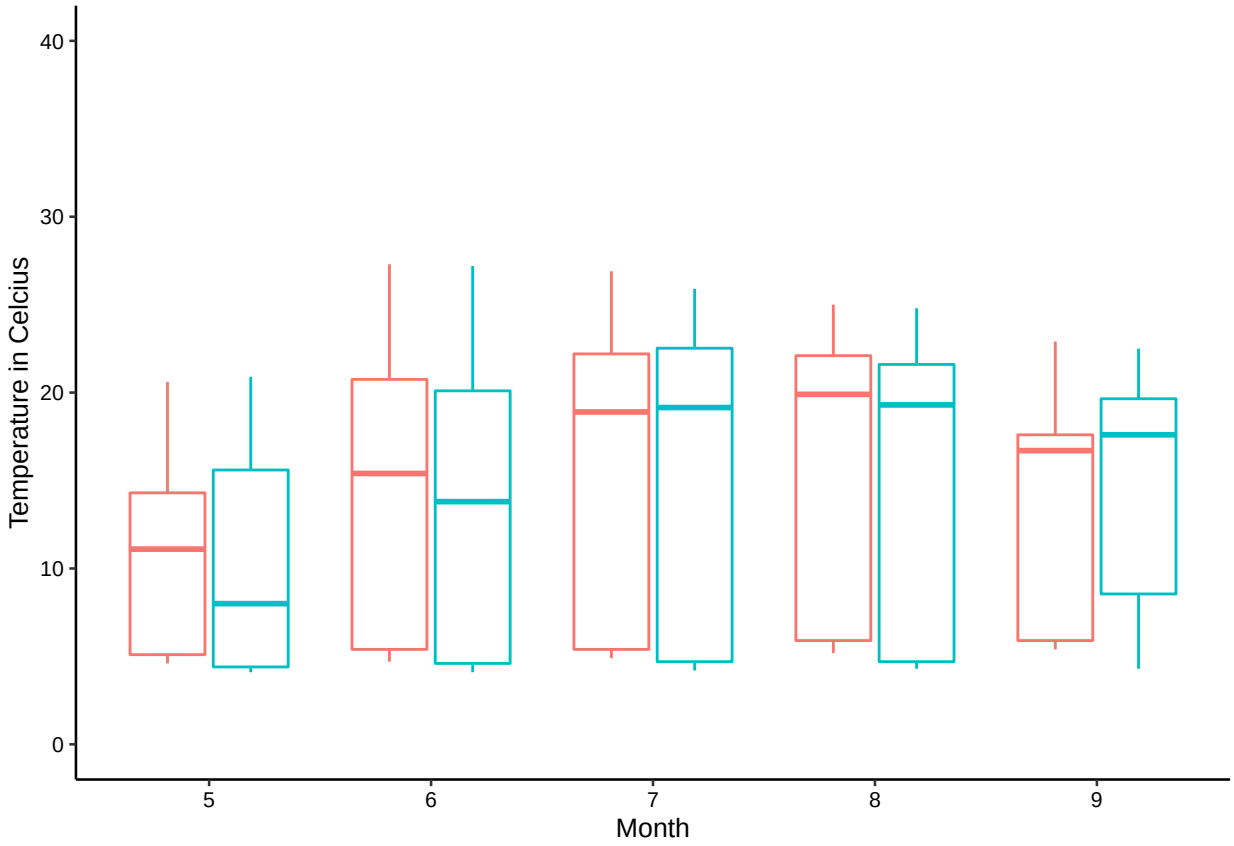
```
## Warning: Removed 2 rows containing missing values (geom_smooth).
```



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.

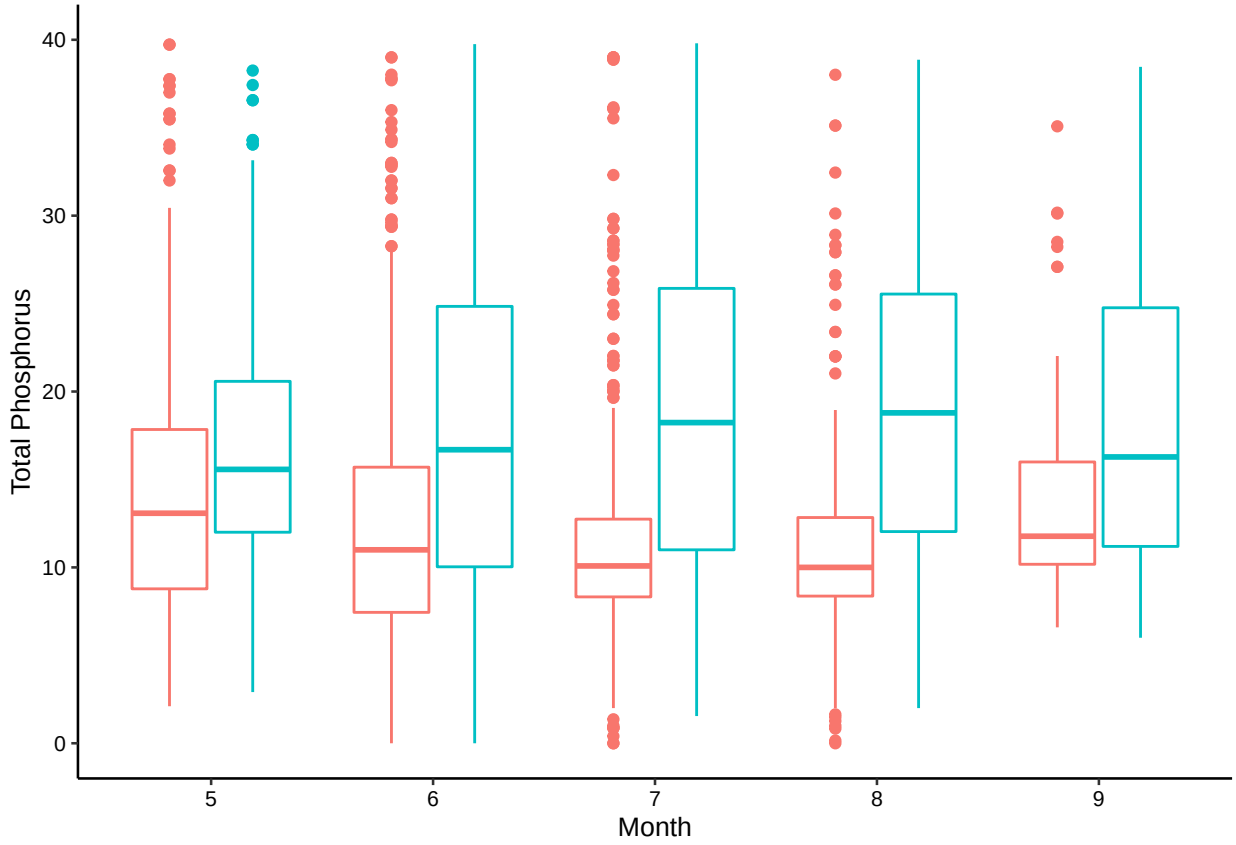
```
#colnames(lakenutmodsub)
lakenutmodsub$month <-as.factor(lakenutmodsub$month)
#class(lakenutmodsub$month)

tempplot <-
  ggplot(lakenutmodsub, aes(x = month, y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  ylab("Temperature in Celcius") + xlab("Month") +
  theme(legend.position = "none") +
   #theme_classic(base_size = 10) +
    ylim(0, 40)
print(tempplot)
```

```
## Warning: Removed 2027 rows containing non-finite values (stat_boxplot).
```



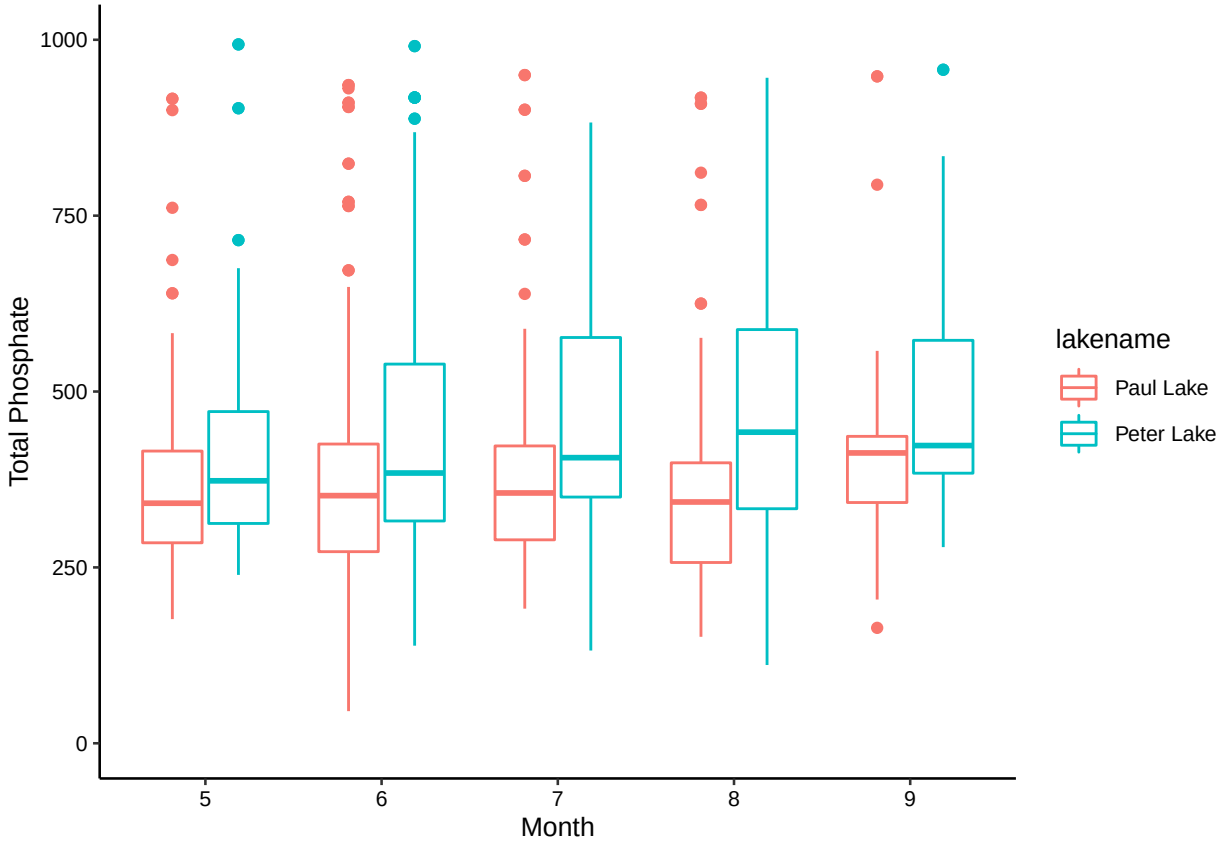

```
#needed to make month into a factor. Before it was a date object



#b) TP plot
TPplot <-
  ggplot(lakenutmodsub, aes(x = month, y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  ylab("Total Phosphorus") + xlab("Month") +
  theme(legend.position = "none") +
  #theme_classic(base_size = 10) +
    ylim(0, 40)
print(TPplot)
```

```
## Warning: Removed 678 rows containing non-finite values (stat_boxplot).
```




```
#c) TN plot
TNplot <-
  ggplot(lakenutmodsub, aes(x = month, y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  ylim(0, 1000) +
  ylab("Total Phosphate") + xlab("Month") +
  theme(legend.position = "right")
print(TNplot)
```

```
## Warning: Removed 1545 rows containing non-finite values (stat_boxplot).
```

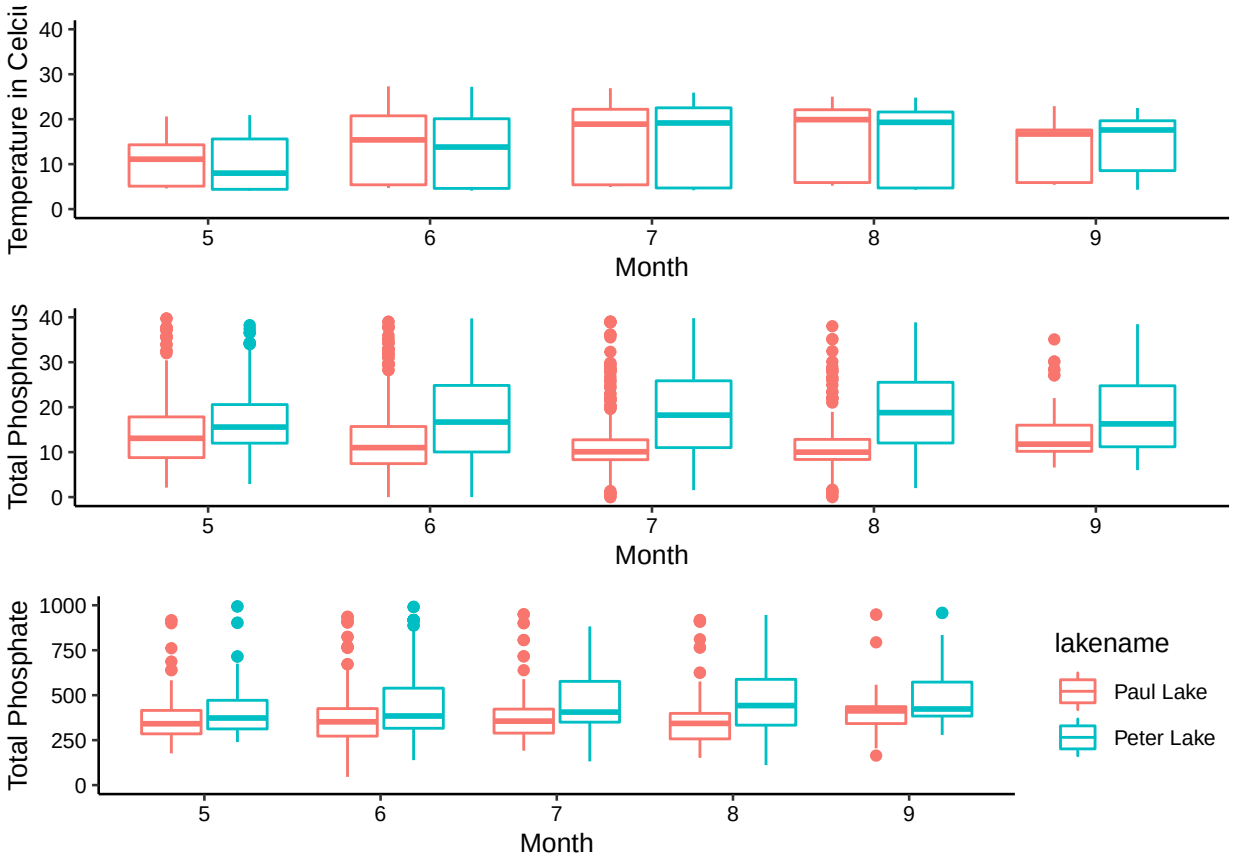



```
#d) cowplot combining the graphs
plot_grid(tempplot, TPplot, TNplot, nrow = 3, align = 'h', rel_heights = c(1, 1, 1))
```

```
## Warning: Removed 2027 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 678 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 1545 rows containing non-finite values (stat_boxplot).
```



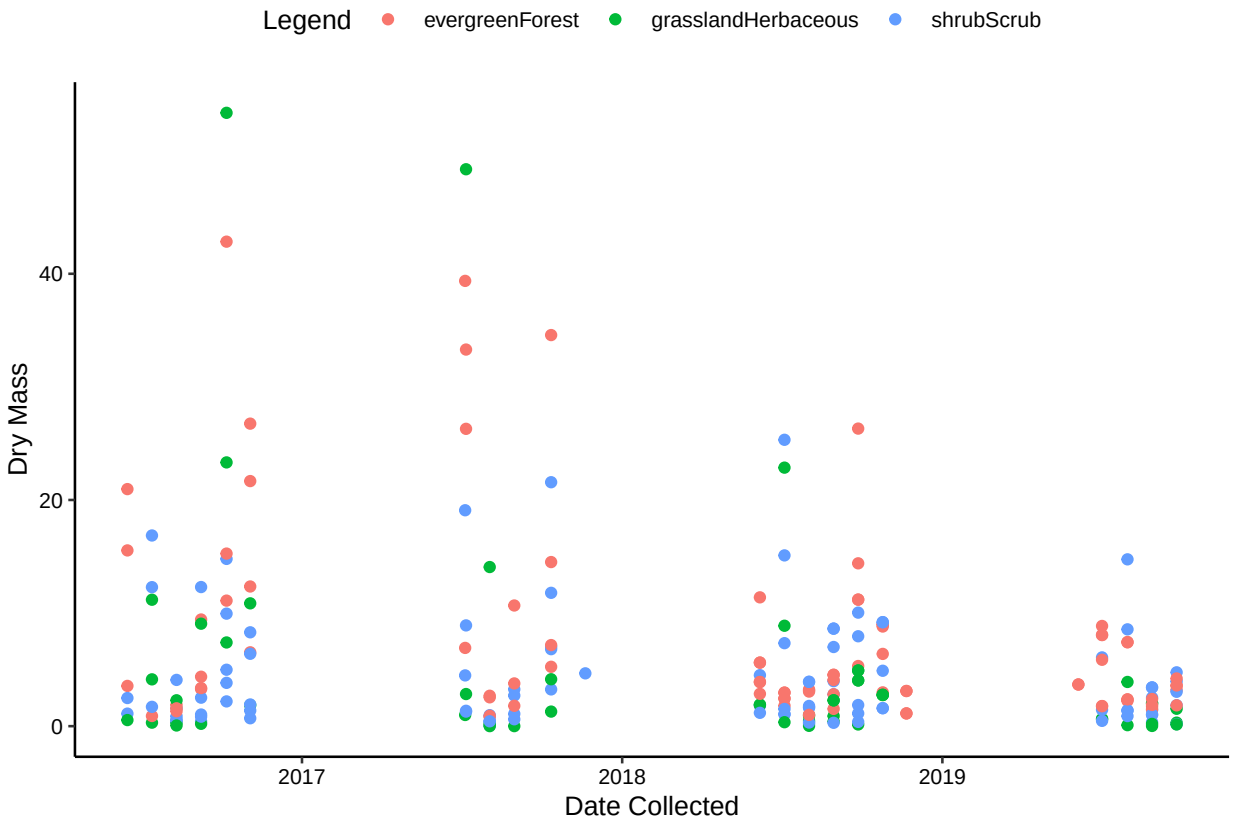
Question: What do you observe about the variables of interest over seasons and between lakes?

Answer: Temperature: The temperature is higher in Paul Lake than Peter Lake during the early months of the summer (May, June). During July, the temperatures are about the same. Total Phosphorus: The phosphorus is higher higher in Peter Lake than Paul Lake during each month. Total Phosphate: The total phosphate is higher in Peter Lake than Paul Lake during each month.

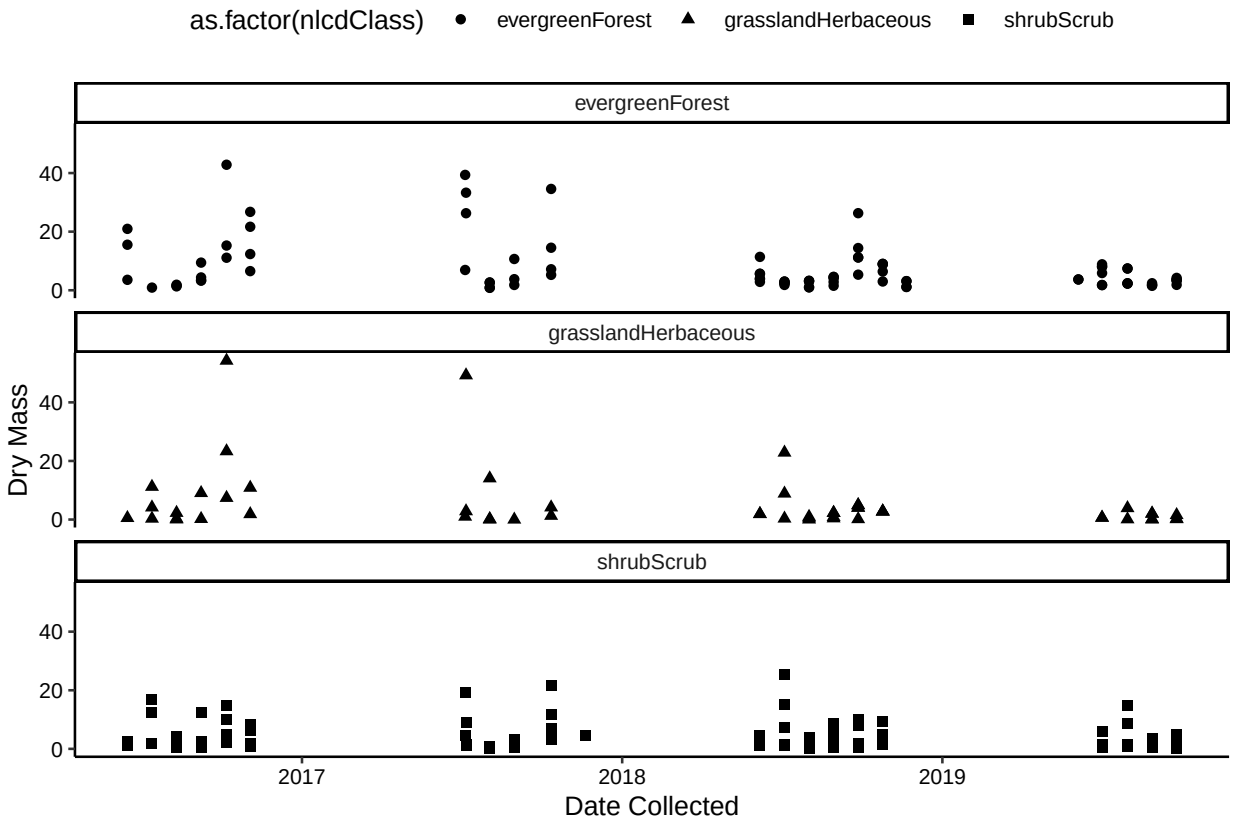
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
        ####6
#what are the NLCD classes??
#View(Niwot_litter)
#colnames(Niwot_litter)
#unique(Niwot_litter$nlcdClass)
#unique(Niwot_litter$functionalGroup)
needles <- filter(Niwot_litter, functionalGroup == "Needles")
#colnames(Niwot_litter)

needleplot <- ggplot(needles, aes(x = collectDate, y = dryMass)) +
  geom_point(aes(color = nlcdClass)) +
  ylab("Dry Mass") + xlab("Date Collected") +
  scale_color_discrete(name = "Legend")
print(needleplot)
```




```
#here, we are using a color plot
#it is discrete, because we have three seperate entities

        ######7

needleplot.faceted <-
  ggplot(needles, aes(x = collectDate, y = dryMass, shape = as.factor(nlcdClass))) +
  geom_point() +
   ylab("Dry Mass") + xlab("Date Collected") +
  facet_wrap(vars(nlcdClass), nrow = 3)
#ylab("Dry Mass") + xlab("Date Collected")

print(needleplot.faceted)
```



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

Answer: Plot 7 is more effective, because it is much easier to identifiy which nlcd classes are putting out relatively more or less dry mass.