Assignment 5: Data Visualization

Matthew Vining

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Visualization

Directions

1. Rename this file `<FirstLast>_A02_CodingBasics.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

The completed exercise is due on Friday, Oct 14th @ 5:00pm.

Set up your session

1. Set up your session. Verify your working directory and load the tidyverse, lubridate, & 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_PeterPau` version) and the processed data file for the Niwot Ridge litter dataset (use the [`NEON_NIWO_Litter_mass_trap_Processed` version).
2. Make sure R is reading dates as date format; if not change the format to date.

```
# 1
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
# install.packages('cowplot')
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```r
Nutrients <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
    stringsAsFactors = TRUE)
Chemistry_physics <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
    stringsAsFactors = TRUE)
# 2
class(Nutrients$sampledate)
```

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

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

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

```r
class(Chemistry_physics$collectDate)
```

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

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

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

Define your theme

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

```r
# 3
A05theme <- theme_classic(base_size = 14) + theme(axis.text = element_text(color = "black"),
    legend.position = "top")  #stick with classic theme

theme_set(A05theme)
```

Create graphs

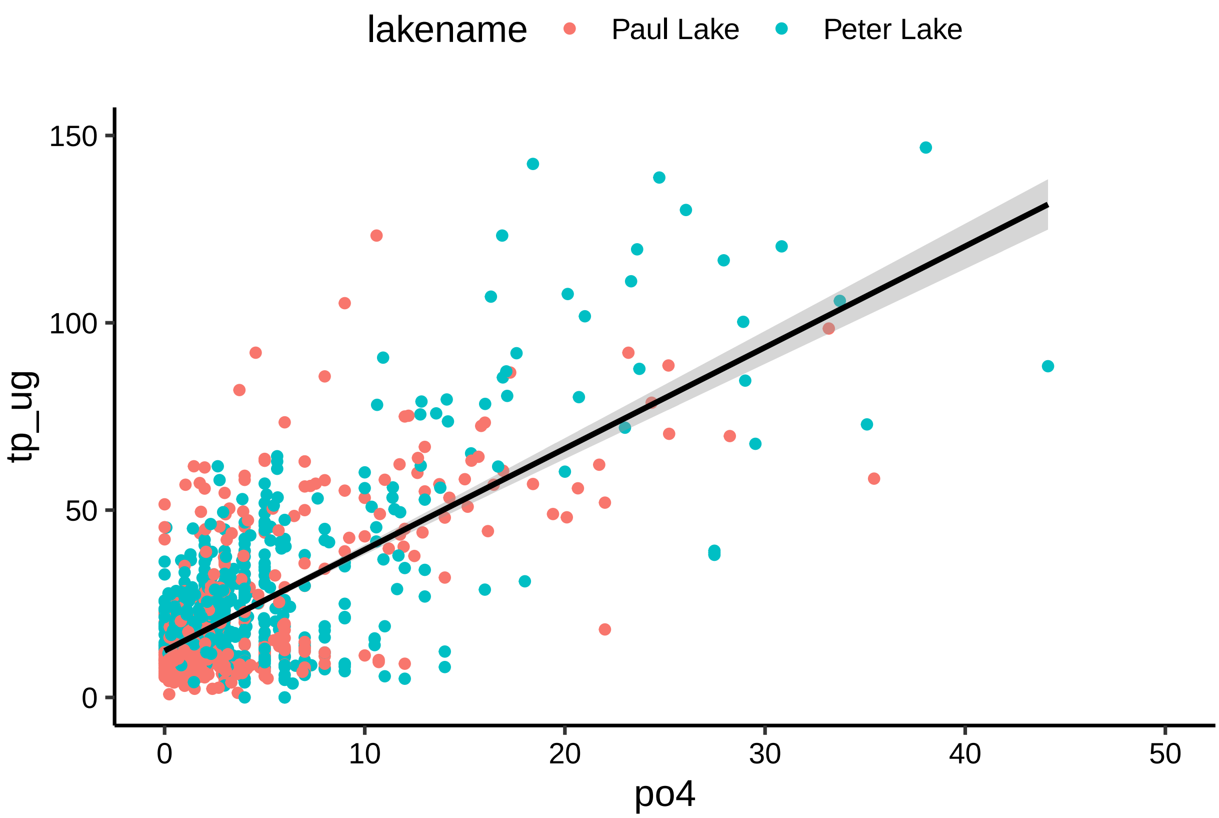
For numbers 4-7, create ggplot graphs and adjust aesthetics to follow best practices for data visualization. Ensure your theme, color palettes, axes, and additional aesthetics are edited accordingly.

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/or `ylim()`).

```r
# 4
Plot_P4 <- ggplot(Nutrients, aes(x = po4, y = tp_ug,
    color = lakename)) + geom_point() + geom_smooth(method = lm,
```

```
    col = "black") + ylim(0, 150) + xlim(0, 50)
print(Plot_P4)
```

```
## `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).
```



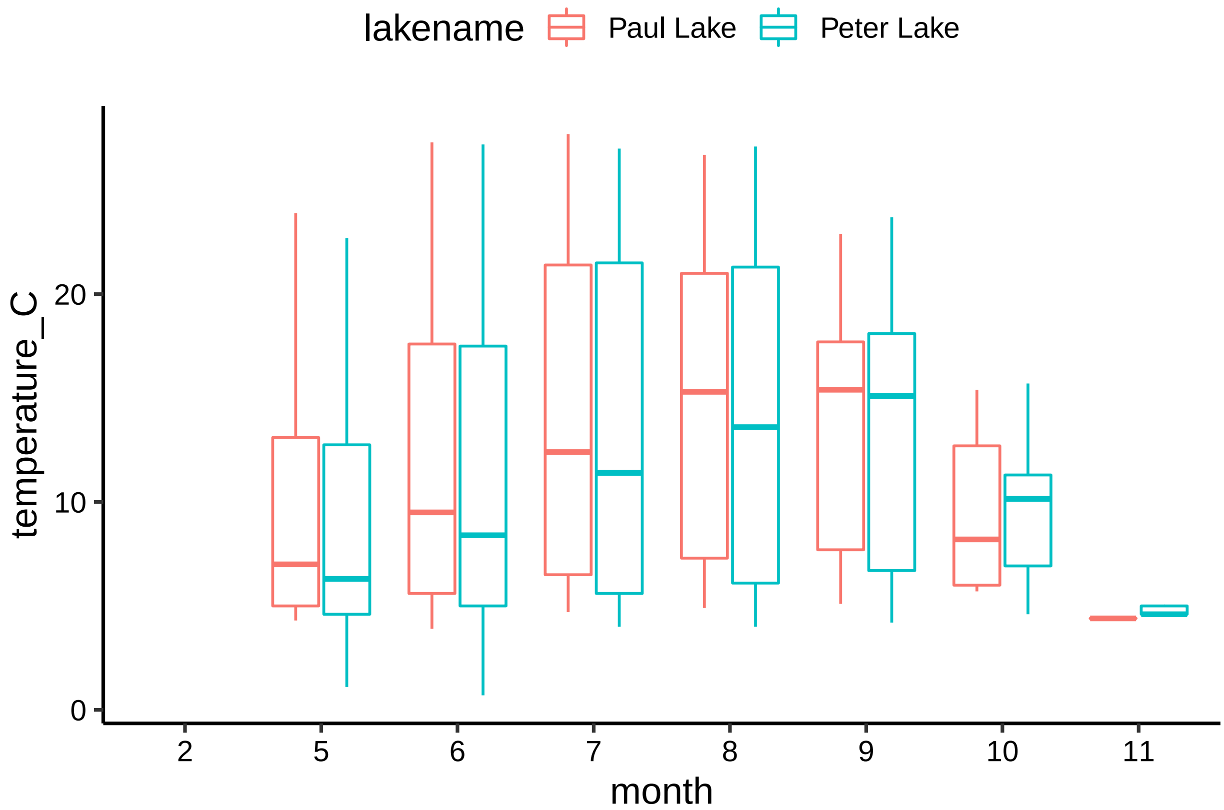
5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and

(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.

Tip: R has a build in variable called `month.abb` that returns a list of months; see https://r-lang.com/month-abb-in-r-with-example

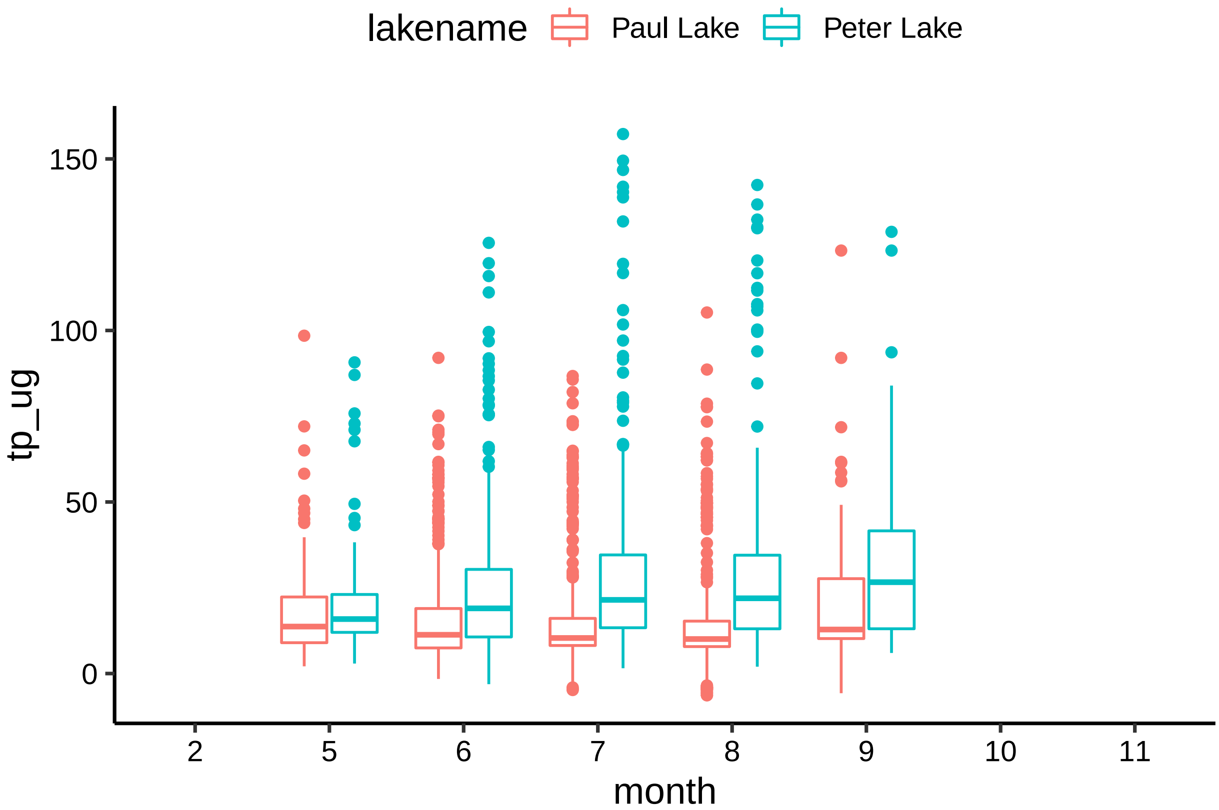
```
# 5

Nutrients$month <- as.factor(Nutrients$month)

Plot_P5.temp <- ggplot(Nutrients, aes(x = month, y = temperature_C)) +
    geom_boxplot(aes(color = lakename))
print(Plot_P5.temp)
```

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



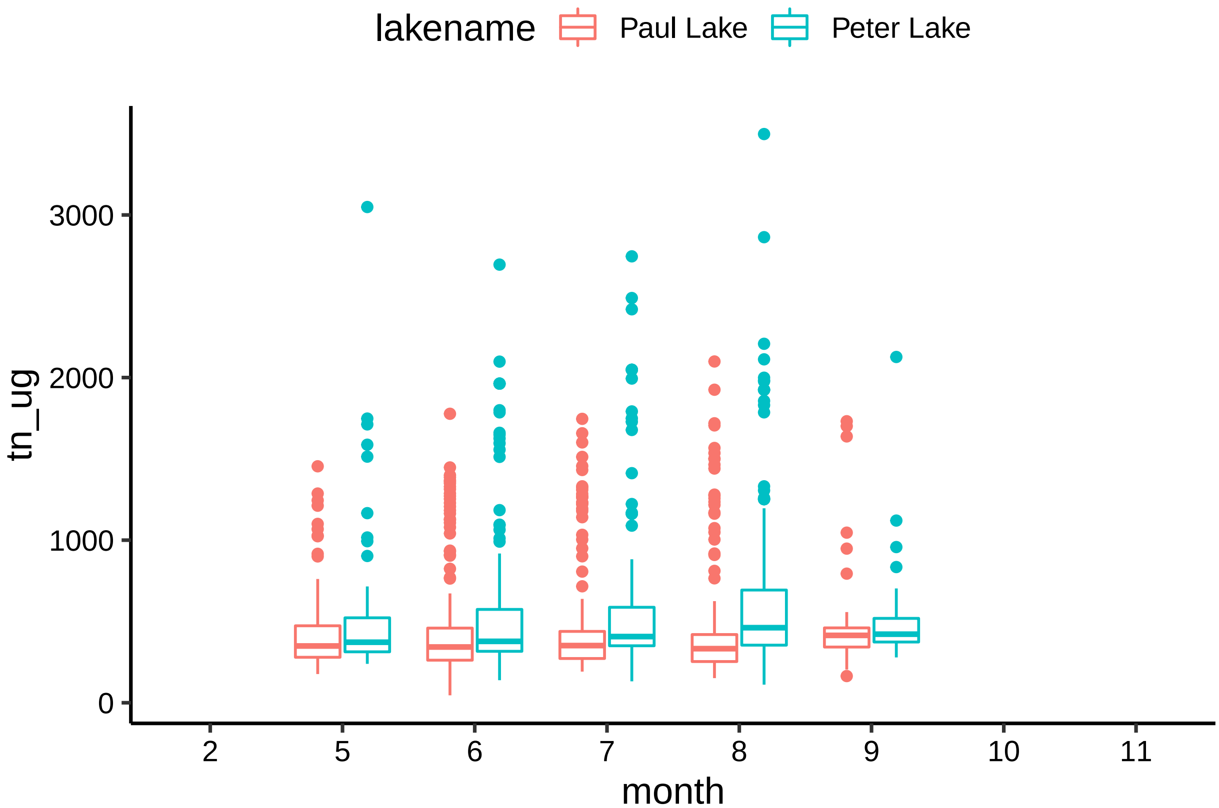

```
Plot_P5.tp <- ggplot(Nutrients, aes(x = month, y = tp_ug)) +
    geom_boxplot(aes(color = lakename))
print(Plot_P5.tp)
```

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




```
Plot_P5.tn <- ggplot(Nutrients, aes(x = month, y = tn_ug)) +
    geom_boxplot(aes(color = lakename))
print(Plot_P5.tn)
```

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



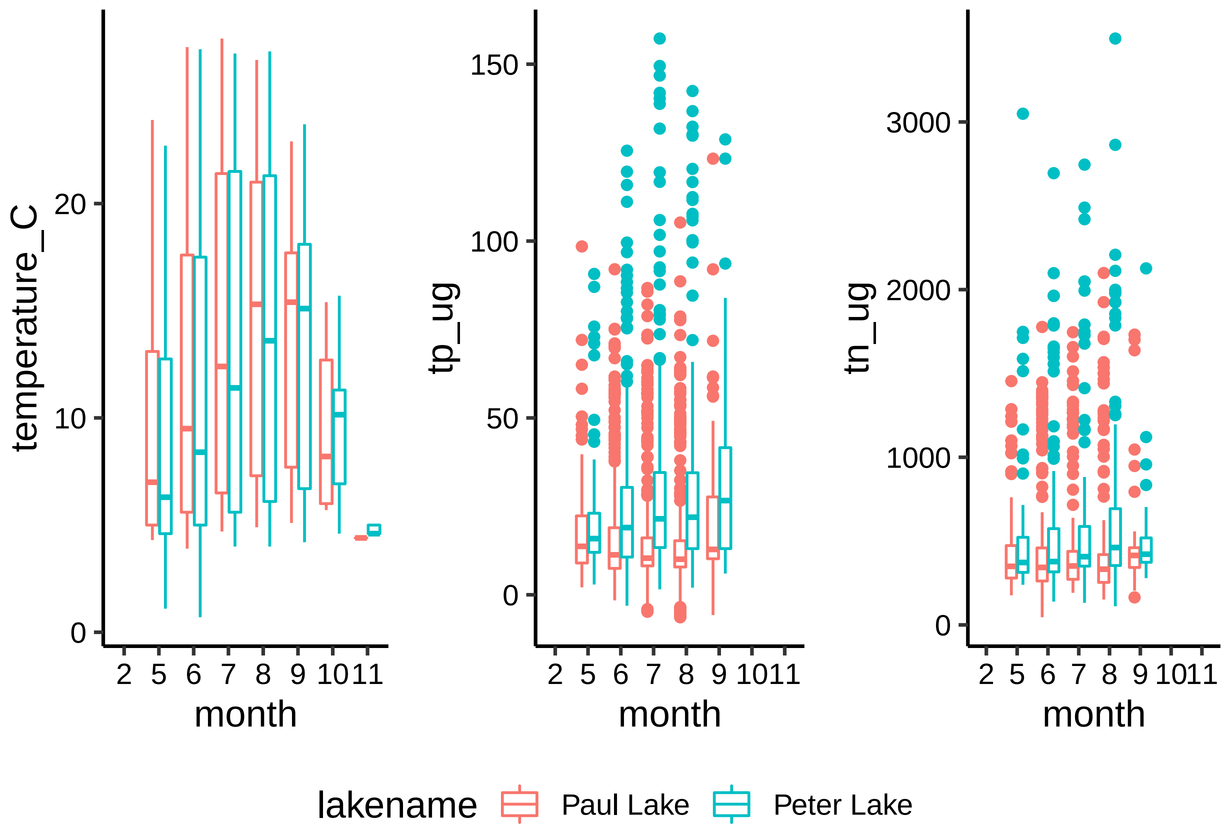

```
plot_nolegend <- plot_grid(Plot_P5.temp + theme(legend.position = "none"),
    Plot_P5.tp + theme(legend.position = "none"), Plot_P5.tn +
        theme(legend.position = "none"), align = "h",
    axis = "bt", nrow = 1, rel_widths = c(3, 3))
```

```
## 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).
```

```
legend <- get_legend(Plot_P5.tn)
```

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

```
plot_grid(plot_nolegend, legend, nrow = 2, rel_heights = c(3,
    0.4))
```

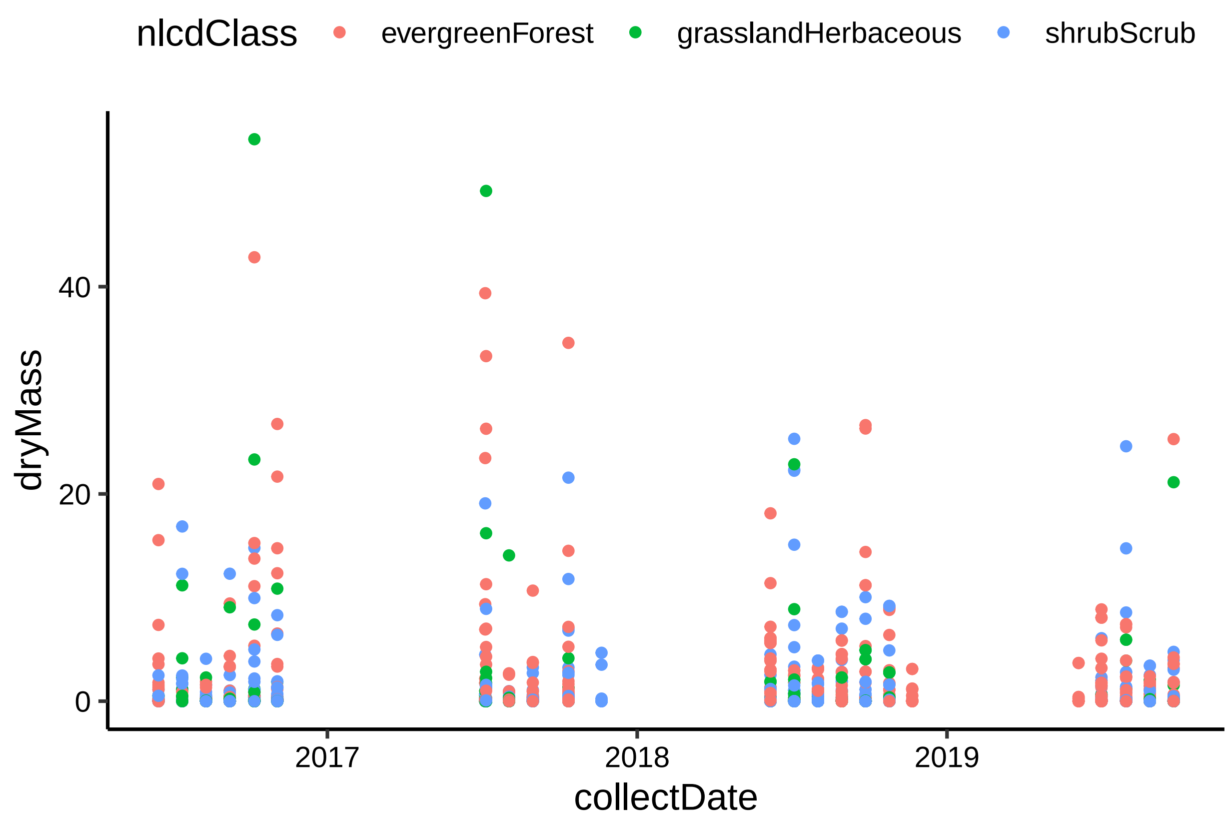


Question: What do you observe about the variables of interest over seasons and between lakes?

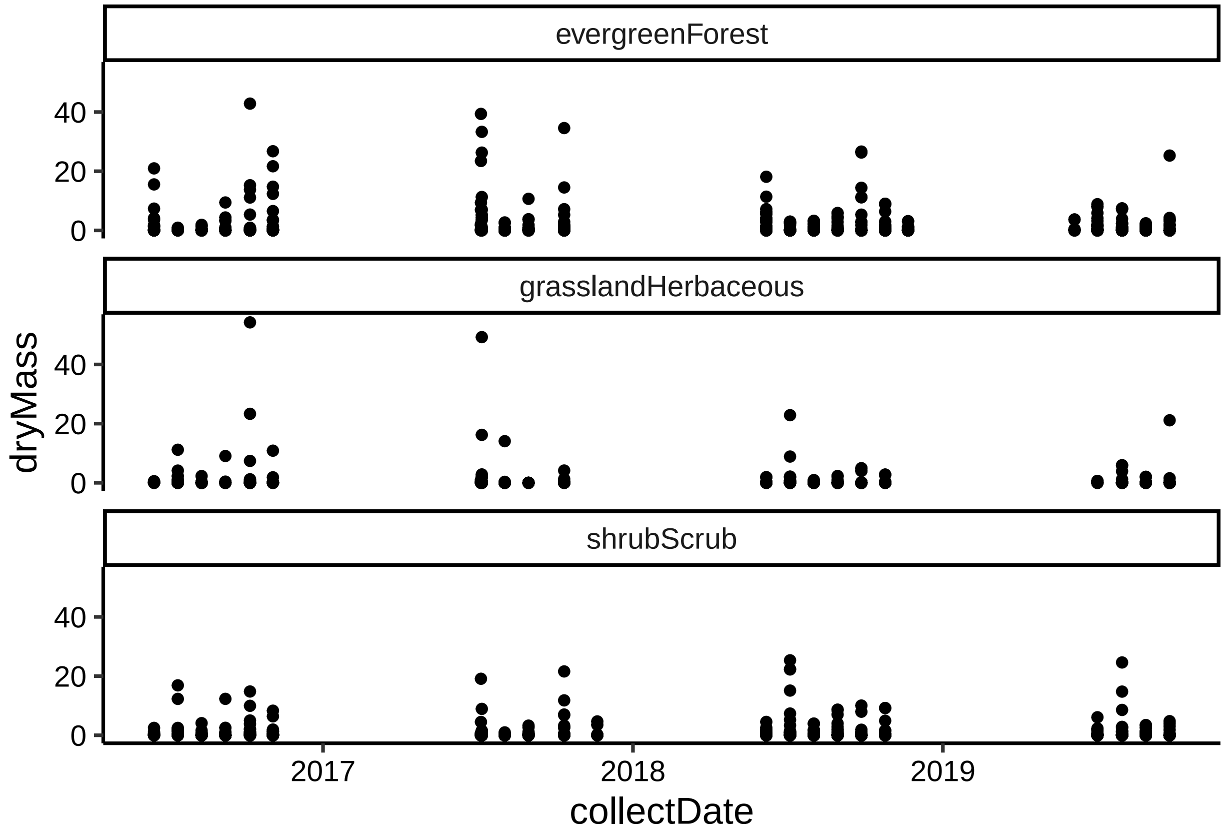
> Answer: The temperature of the lakes across all months seem to be quite similar. The summer season has the warmest temperatures for both lakes. The total phsophorous count is on average higher through all months in Peter Lake than in Paul Lake. Similar to the total phosphorous trend, the total nitrogen count of Peter Lake is great than Paul LAke on average throughout all months. Both nitrogen and phosphorous measurements have higher counts in the summer season than the winter season for both Lakes. The trend in nutrients seems to mirror the temperature trends as well throughout the 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
Plot.P6 <- ggplot(subset(Chemistry_physics, functionalGroup = "Needles"),
    aes(x = collectDate, y = dryMass, color = nlcdClass)) +
    geom_point()
print(Plot.P6)
```




```
# 7
Plot.P7 <- ggplot(subset(Chemistry_physics, functionalGroup = "Needles"),
    aes(x = collectDate, y = dryMass)) + geom_point() +
    facet_wrap(vars(nlcdClass), nrow = 3)
print(Plot.P7)
```



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

Answer:I am a very visual learner and I think its easier for my eyes to compare the different needle classes by color instead of separating them into facets. I would have to say for me that plot 6 is more effective for drawing comparisons. Maybe if I expanded the height of plot 7 enough then it would be better, but its easier to compare on one graph with varying colors in my opinion.