Assignment 5: Data Visualization

Kallie Davis

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_PeterPa` version) and the processed data file for the Niwot Ridge litter dataset (use the [`NEON_NIWO_Litter_mass_trap_Processe` version).

2. Make sure R is reading dates as date format; if not change the format to date.

```
# 1
getwd()
```

```
## [1] "C:/Users/kalli/OneDrive/Desktop/Grad_School/Environmental Data Analytics_ENV_872/EDA-Fall2022/As
```

```
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   1.2.1      v stringr 1.4.1
## v readr   2.1.3      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
```

```
library(cowplot)
```

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

```
NTL_LTER_Lake_Chemistry_PeterPaul <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_Pete
    stringsAsFactors = TRUE)
NEON_NIWO_Litter_mass_trap <- read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
    stringsAsFactors = TRUE)

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

Define your theme

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

```
# 3
mytheme <- theme_bw(base_size = 15) + theme(axis.text = element_text(color = "maroon"))

theme_set(mytheme)
```

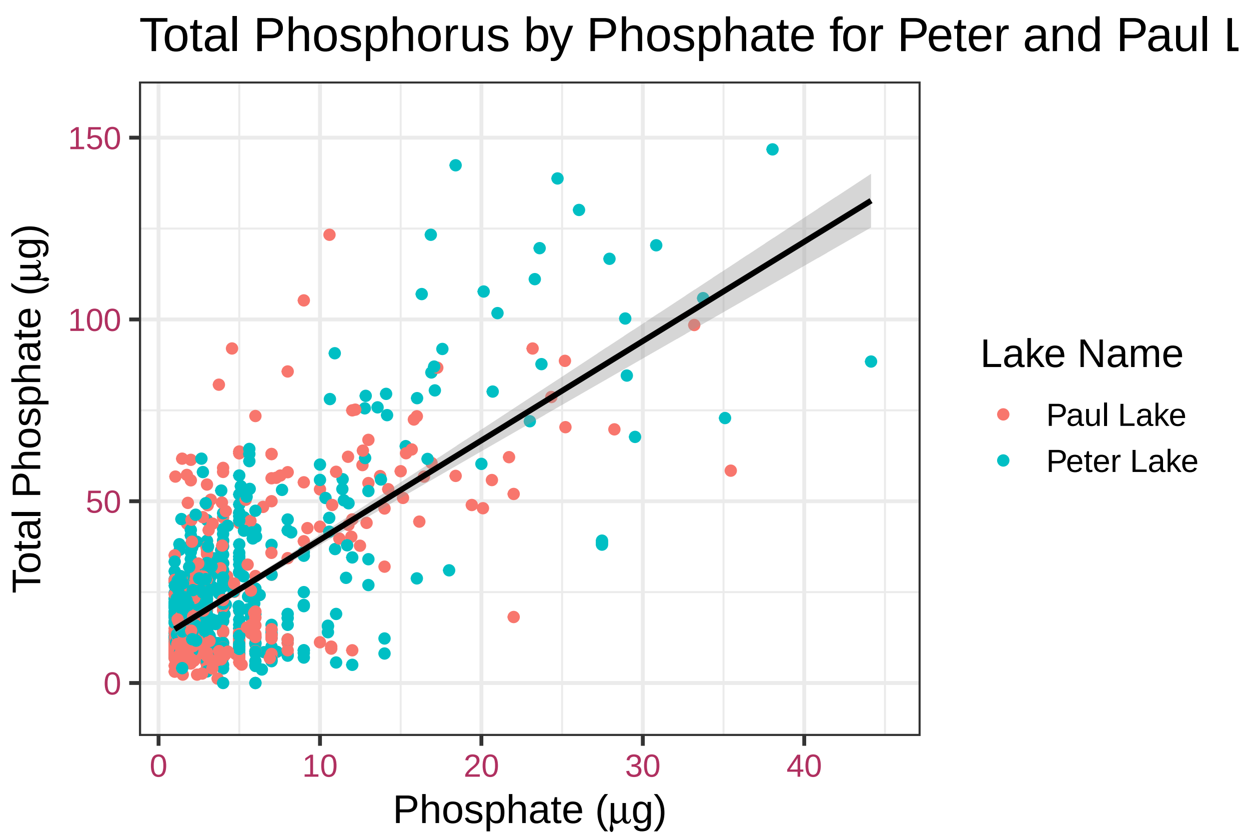
Create graphs

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/or `ylim()`).

```r
# 4
tp_phosphate_PeterPaul <- ggplot(subset(NTL_LTER_Lake_Chemistry_PeterPaul, lakename ==
    "Paul Lake" | lakename == "Peter Lake"), aes(x = po4, y = tp_ug)) + geom_point(aes(color = lakename)
    labs(title = "Total Phosphorus by Phosphate for Peter and Paul Lakes", x = expression("Phosphate ("
        mu * "g)"), y = expression("Total Phosphate (" * mu * "g)"), color = "Lake Name") +
    geom_smooth(method = lm, color = "black") + xlim(1, 45)

print(tp_phosphate_PeterPaul)
```

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

## Warning: Removed 22116 rows containing non-finite values (stat_smooth).

## Warning: Removed 22116 rows containing missing values (geom_point).
```



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

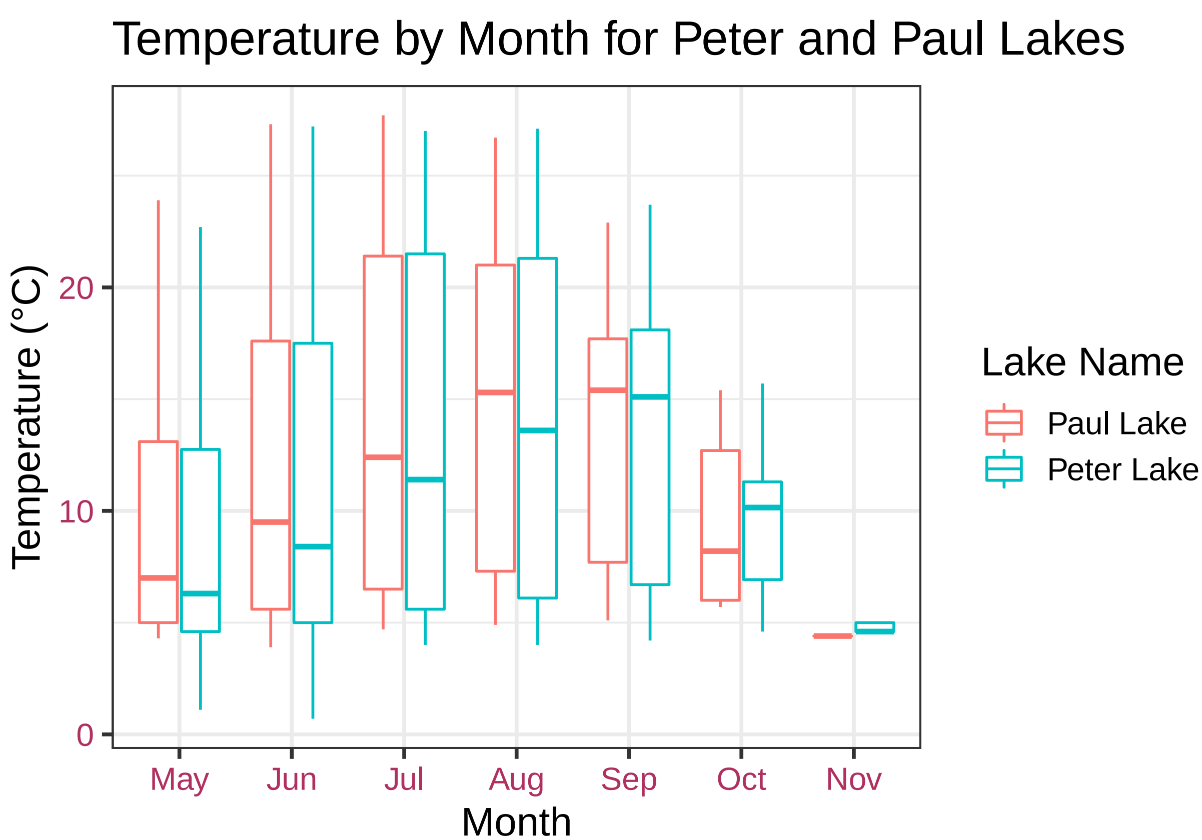
(c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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

```r
# 5 temperature plot
NTL_LTER_Lake_Chemistry_PeterPaul$month <- month.abb[NTL_LTER_Lake_Chemistry_PeterPaul$month]

NTL_LTER_temp <- ggplot(NTL_LTER_Lake_Chemistry_PeterPaul) + geom_boxplot(aes(x = forcats::fct_inorder(
    y = temperature_C, color = lakename)) + labs(title = "Temperature by Month for Peter and Paul Lakes
    x = "Month", y = "Temperature (°C)", color = "Lake Name") + xlim("May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov")

print(NTL_LTER_temp)
```

```
## Warning: Removed 16 rows containing missing values (stat_boxplot).
```

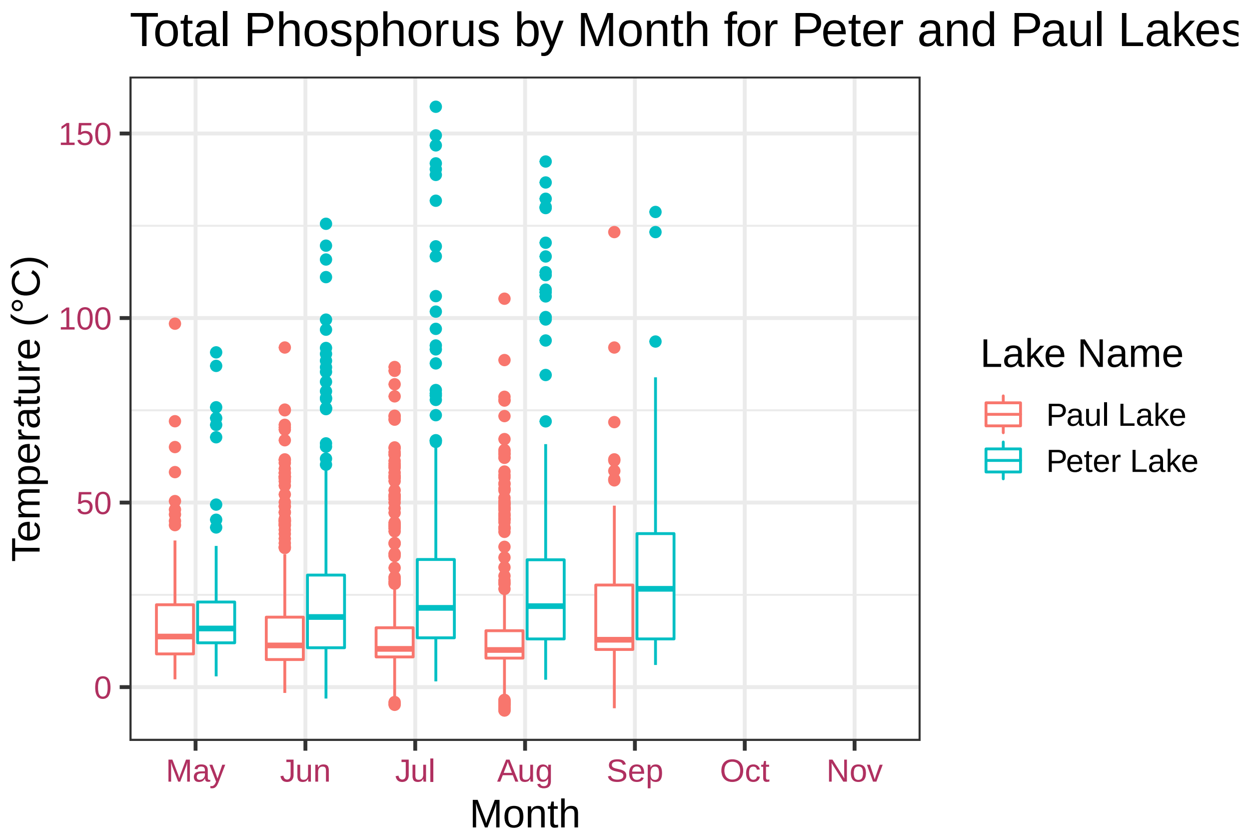
```
## Warning: Removed 3550 rows containing non-finite values (stat_boxplot).
```




```r
# total phosphorus plot
NTL_LTER_tp_ug <- ggplot(NTL_LTER_Lake_Chemistry_PeterPaul) + geom_boxplot(aes(x = forcats::fct_inorder
    y = tp_ug, color = lakename)) + labs(title = "Total Phosphorus by Month for Peter and Paul Lakes",
    x = "Month", y = "Temperature (°C)", color = "Lake Name") + xlim("May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov")

print(NTL_LTER_tp_ug)
```

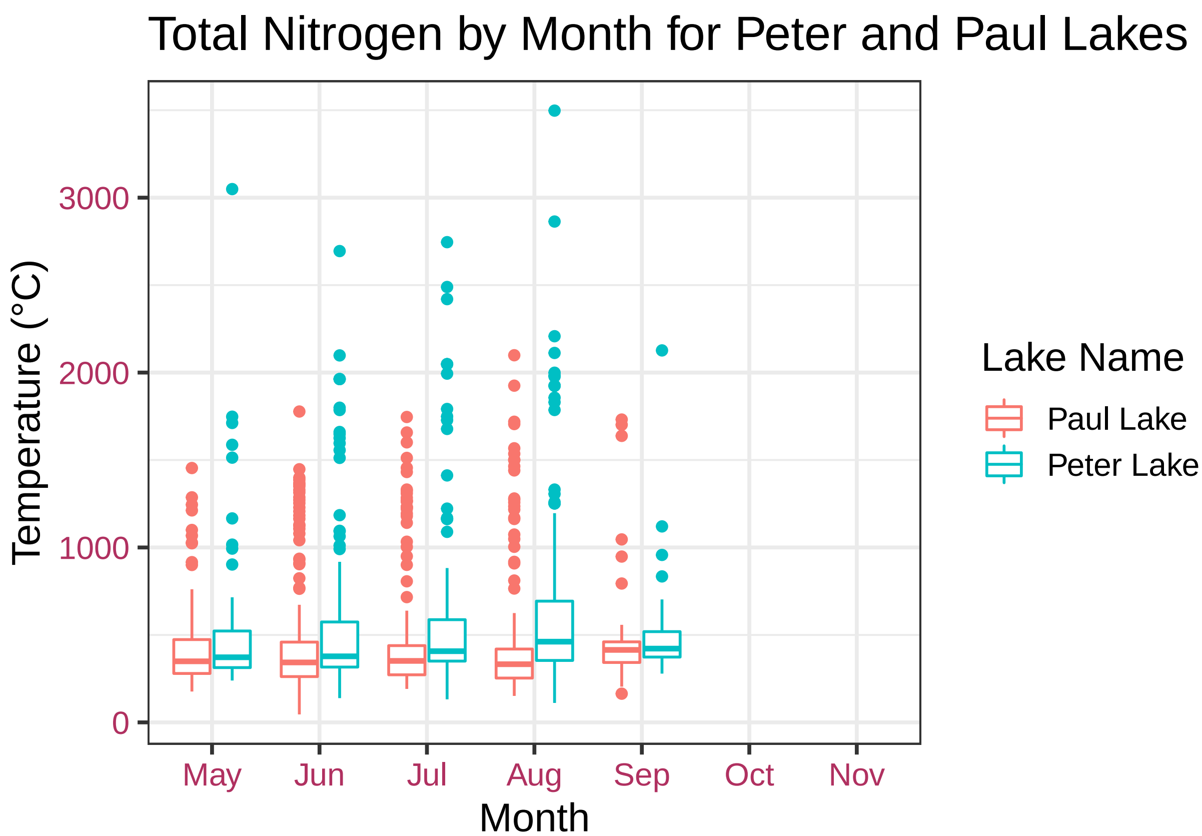
```
## Warning: Removed 16 rows containing missing values (stat_boxplot).
```

```
## Warning: Removed 20713 rows containing non-finite values (stat_boxplot).
```




```r
# total nitrogen plot
NTL_LTER_tn_ug <- ggplot(NTL_LTER_Lake_Chemistry_PeterPaul) + geom_boxplot(aes(x = forcats::fct_inorder
    y = tn_ug, color = lakename)) + labs(title = "Total Nitrogen by Month for Peter and Paul Lakes",
    x = "Month", y = "Temperature (°C)", color = "Lake Name") + xlim("May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov")

print(NTL_LTER_tn_ug)
```

```
## Warning: Removed 16 rows containing missing values (stat_boxplot).

## Warning: Removed 21567 rows containing non-finite values (stat_boxplot).
```




```r
# combined plot
NTL_LTER_combined <- plot_grid(NTL_LTER_temp + theme(legend.position = "none"), NTL_LTER_tp_ug +
    theme(legend.position = "none"), NTL_LTER_tn_ug + theme(legend.position = "none"),
    nrow = 3, ncol = 1, align = "v")
```

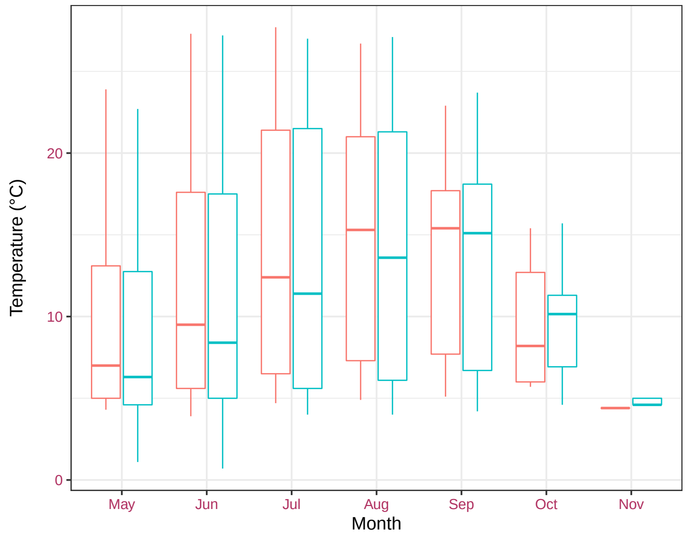
```
## Warning: Removed 16 rows containing missing values (stat_boxplot).

## Warning: Removed 3550 rows containing non-finite values (stat_boxplot).

## Warning: Removed 16 rows containing missing values (stat_boxplot).

## Warning: Removed 20713 rows containing non-finite values (stat_boxplot).

## Warning: Removed 16 rows containing missing values (stat_boxplot).

## Warning: Removed 21567 rows containing non-finite values (stat_boxplot).
```

```r
NTL_LTER_combined_1 <- cowplot::plot_grid(NTL_LTER_combined, cowplot::get_legend(NTL_LTER_temp),
    rel_widths = c(0.8, 0.2))
```

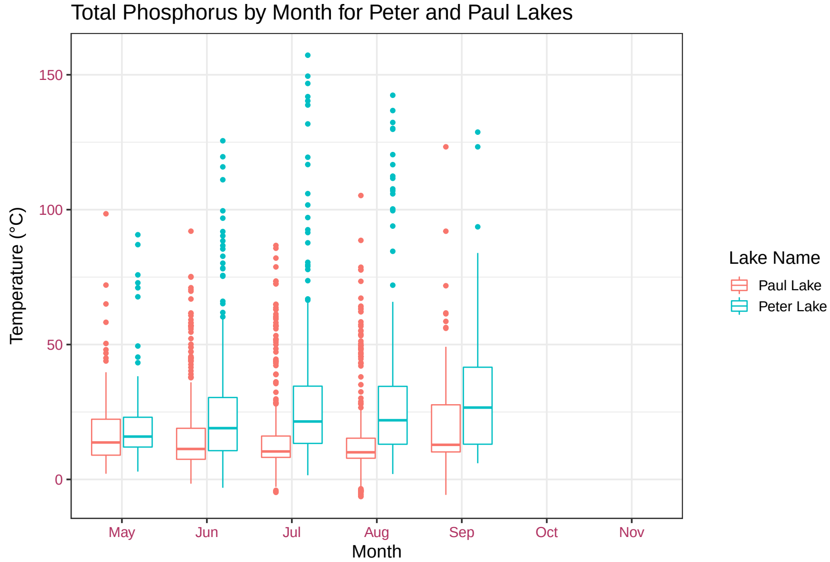
```
## Warning: Removed 16 rows containing missing values (stat_boxplot).

## Warning: Removed 3550 rows containing non-finite values (stat_boxplot).
```

```
print(NTL_LTER_combined_1)
```

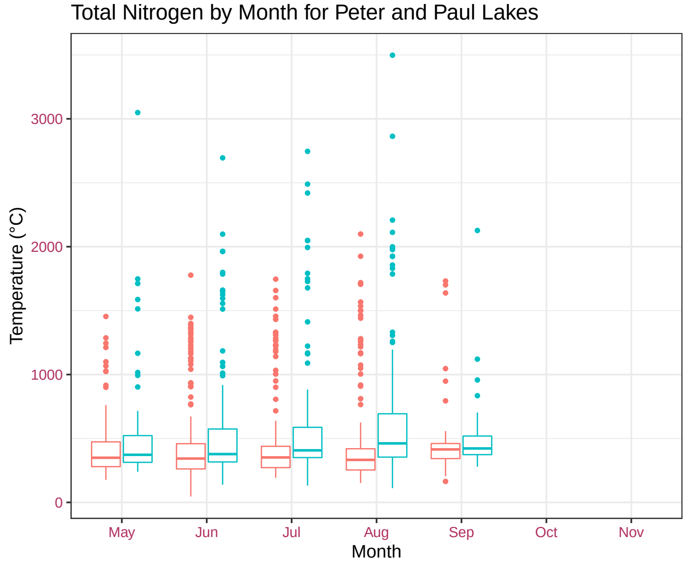
Temperature by Month for Peter and Paul Lakes



Total Phosphorus by Month for Peter and Paul Lakes



Total Nitrogen by Month for Peter and Paul Lakes



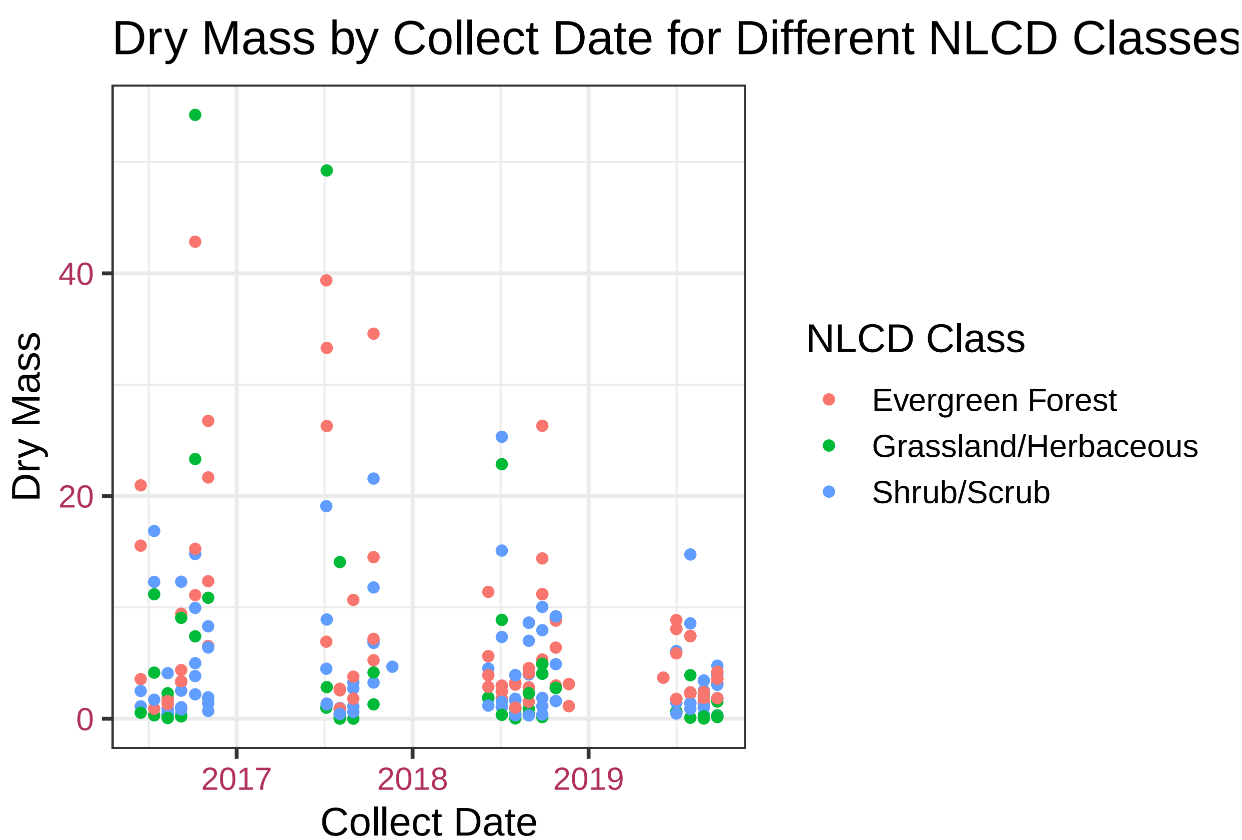
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Total nitrogen and phosphorus have relatively consistent median values across seasons. Median total phosphorus and nitrogen values are greater in Peter lake.

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)

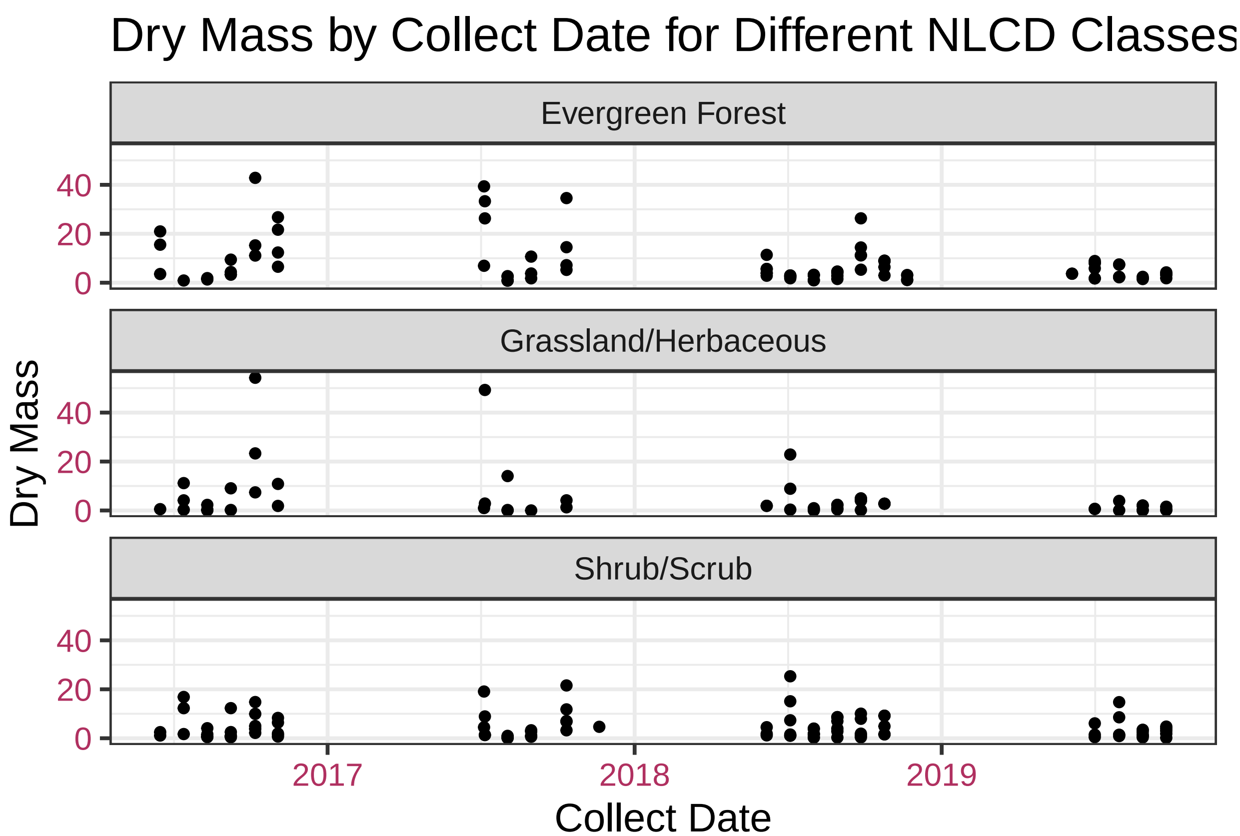
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
# 6
Needles_dryMass_CollectDate <- ggplot(subset(NEON_NIWO_Litter_mass_trap, functionalGroup ==
    "Needles")) + geom_point(aes(x = collectDate, y = dryMass, color = nlcdClass)) +
    labs(x = "Collect Date", y = "Dry Mass", title = "Dry Mass by Collect Date for Different NLCD Classe
        color = "NLCD Class") + scale_color_discrete(labels = c("Evergreen Forest",
    "Grassland/Herbaceous", "Shrub/Scrub"))

print(Needles_dryMass_CollectDate)
```




```
# 7
NLCD_names <- c(evergreenForest = "Evergreen Forest", grasslandHerbaceous = "Grassland/Herbaceous",
    shrubScrub = "Shrub/Scrub")
```

```
Needles_dryMass_CollectDate_faceted <- ggplot(subset(NEON_NIWO_Litter_mass_trap,
    functionalGroup == "Needles")) + geom_point(aes(x = collectDate, y = dryMass)) +
    labs(x = "Collect Date", y = "Dry Mass", title = "Dry Mass by Collect Date for Different NLCD Classe
    facet_wrap(vars(nlcdClass), nrow = 3, labeller = as_labeller(NLCD_names))

print(Needles_dryMass_CollectDate_faceted)
```



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

> Answer:In this case I think the non faceted figure is more effective mainly because it is easier to see where the data vary when on the same plot. The differences are small enough that it is a little challenging to spot major variations on the faceted plot.