Assignment 5: Data Visualization

Isabel Zungailia

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
getwd()  #Check working directory
```

```
## [1] "/home/guest/EDA-Fall2022"
```

```
library(tidyverse)  #Load the tidyverse package
```

```
## -- 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)  #Load the lubridate package
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
```

```
##     date, intersect, setdiff, union
```

```
library(cowplot)  #Load the cowplot package
```

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

```
# Upload processed data files
PeterPaul.chem.nutrients <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Proc
    stringsAsFactors = TRUE)

NiwotRidge.litter <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
    stringsAsFactors = TRUE)

# 2 Change format of dates to 'Date'
# for both datasets
class(PeterPaul.chem.nutrients$sampledate)
```

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

```
PeterPaul.chem.nutrients$sampledate <- as.Date(PeterPaul.chem.nutrients$sampledate,
    format = "%Y-%m-%d")
class(PeterPaul.chem.nutrients$sampledate)
```

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

```
class(NiwotRidge.litter$collectDate)
```

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

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

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

Define your theme

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

```
#3
#Build a theme
mytheme <- theme_classic(base_size = 12) + #Set text size to 12
  theme(axis.text = element_text(color = "black"), #Change text color to black
        legend.position = "top") #Set the legend to be positioned at the top of the plot

#Set the theme as the default for subsequent plots
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.

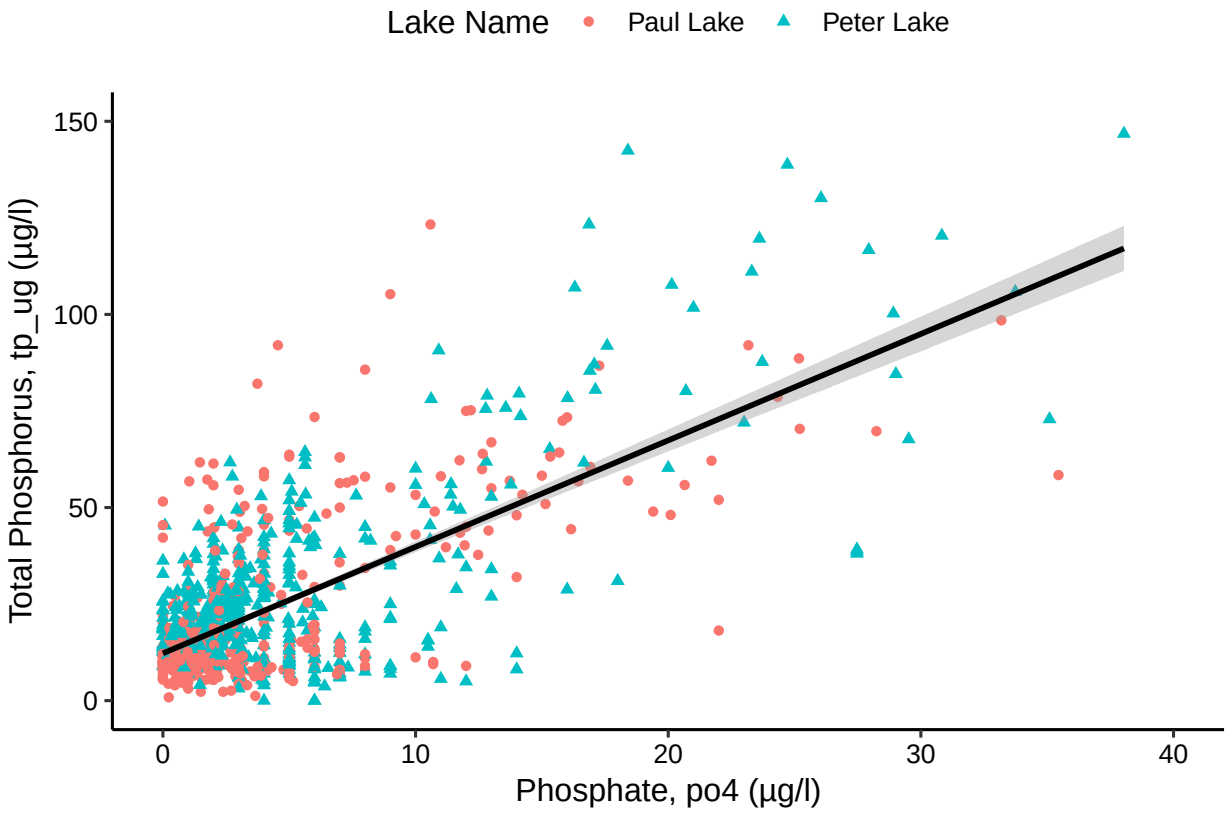
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```
#4
#Plot total phosphorus (`tp_ug`) by phosphate (`po4`)
totalphos.by.phosphate <-
  ggplot(PeterPaul.chem.nutrients, aes(x = po4, y = tp_ug)) +
  xlab(expression("Phosphate, po4 (µg/l)")) + #Change x-axis label
  ylab(expression("Total Phosphorus, tp_ug (µg/l)")) + #Change y-axis label
  xlim(0, 40) + #Adjust axis to hide extreme values
  ylim(0, 150) +
  guides(color=guide_legend(title="Lake Name")) + #Change legend title to "Lake Name"
  guides(shape=guide_legend(title="Lake Name")) +
    geom_point(aes(color = lakename, shape = lakename)) + #Separate aesthetics for Peter and Paul lakes
  geom_smooth(method = lm, color = "black") #Add a line of best fit, color it black
print(totalphos.by.phosphate)
```

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

```
## Warning: Removed 21949 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 21949 rows containing missing values (geom_point).
```



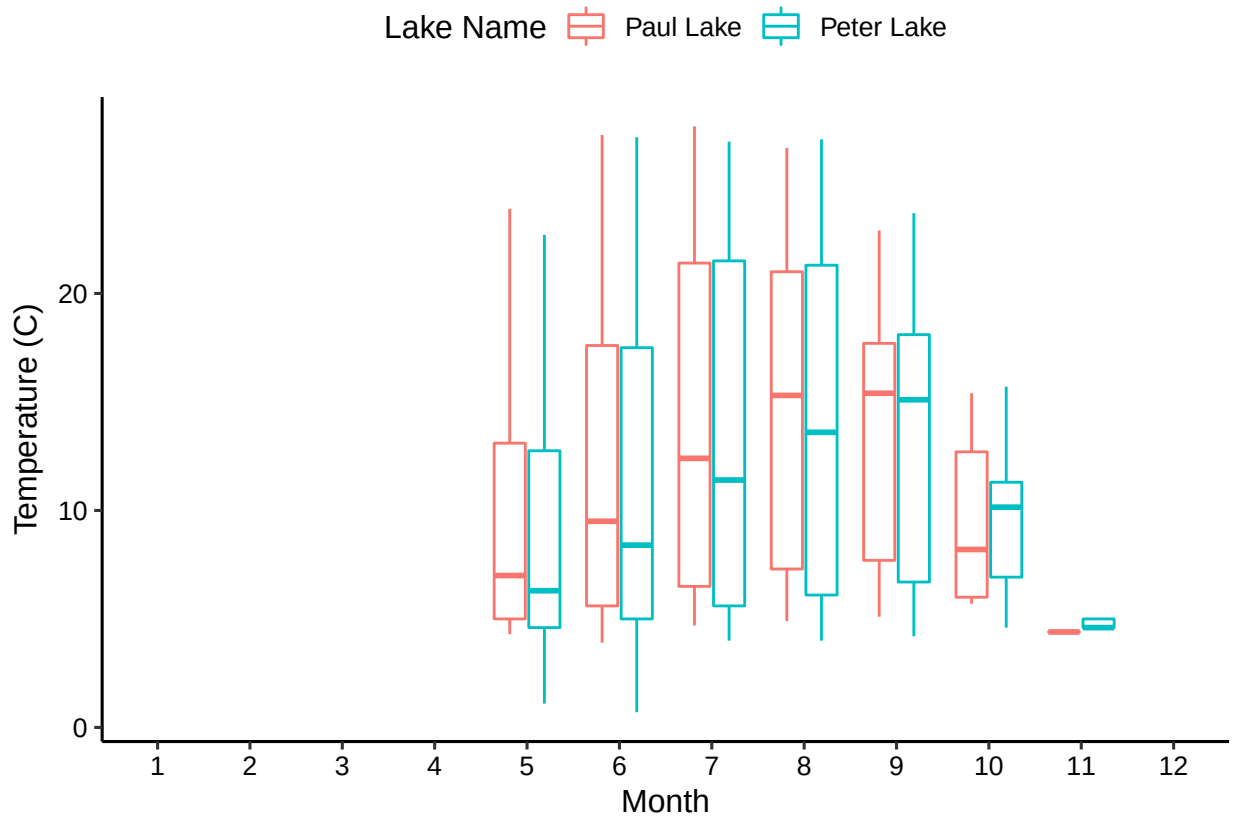
5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and

(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

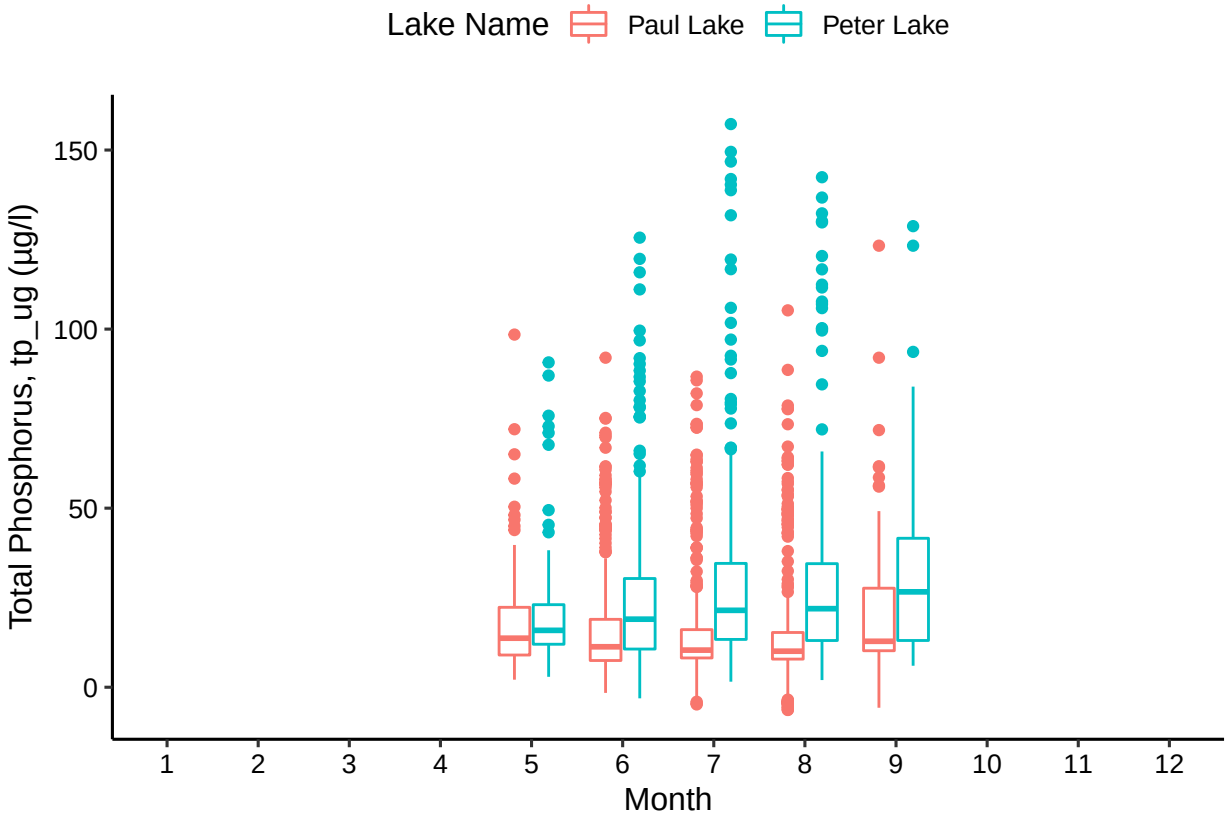
```
#5
#Make three separate boxplots of (a) temperature, (b) TP, and (c) TN
#Temperature boxplot
boxplotA <-
  ggplot(PeterPaul.chem.nutrients, aes(x = factor(month, levels = c(1:12)), y = temperature_C)) + #Conv
  xlab(expression("Month")) + #Set axis labels
  ylab(expression("Temperature (C)")) +
  guides(color=guide_legend(title="Lake Name")) + #Set legend title to "Lake Name
  geom_boxplot(aes(color = lakename)) + #Set lake as a color aesthetic
    scale_x_discrete(drop = FALSE) #Prevent ggplot from dropping levels having no data
print(boxplotA)
```

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



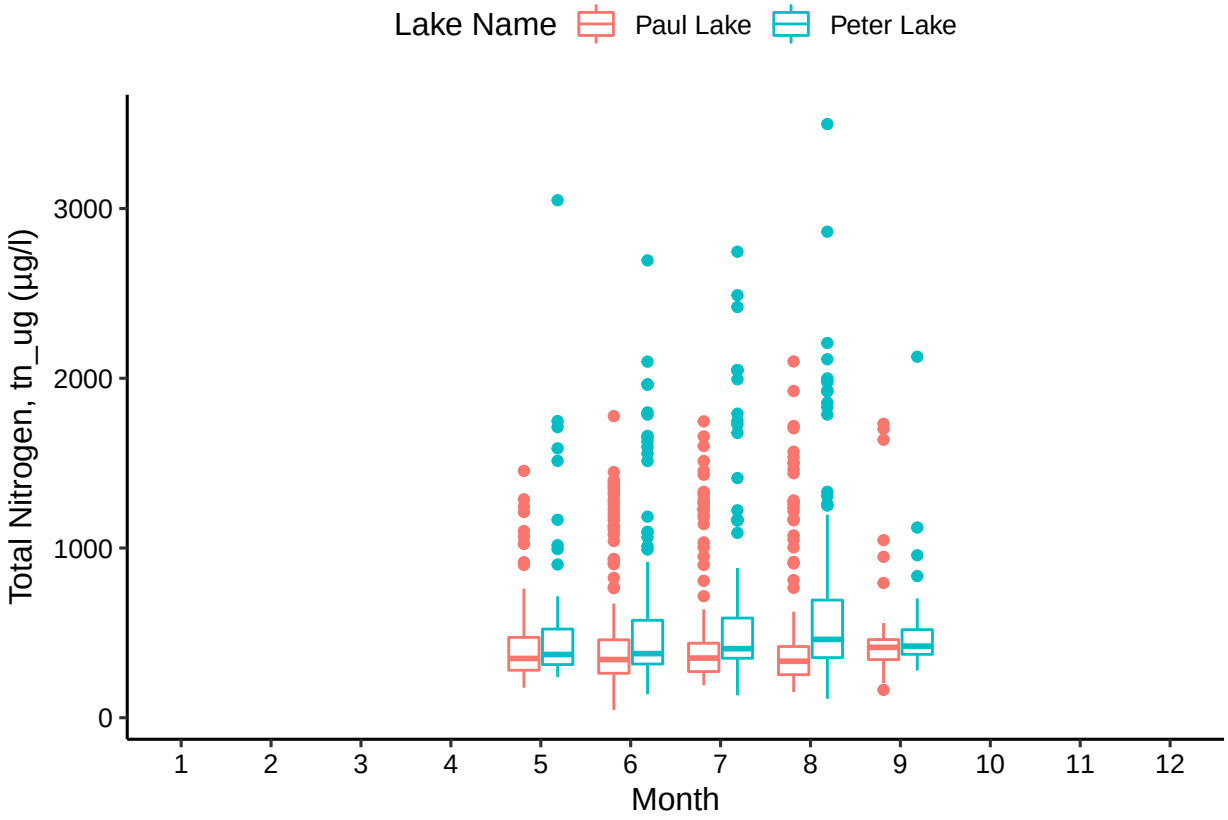

```
#Total phosphorous boxplot
boxplotB <-
  ggplot(PeterPaul.chem.nutrients, aes(x = factor(month, levels = c(1:12)), y = tp_ug)) + #Convert monti
  xlab(expression("Month")) + #Set axis labels
  ylab(expression("Total Phosphorus, tp_ug (µg/l)")) +
  guides(color=guide_legend(title="Lake Name")) + #Set legend title to "Lake Name
  geom_boxplot(aes(color = lakename)) + #Set lake as a color aesthetic
  scale_x_discrete(drop = FALSE) #Prevent ggplot from dropping levels having no data
print(boxplotB)
```

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




```
#Total nitrogen boxplot
boxplotC <-
  ggplot(PeterPaul.chem.nutrients, aes(x = factor(month, levels = c(1:12)), y = tn_ug)) + #Convert month
  xlab(expression("Month")) + #Set axis labels
  ylab(expression("Total Nitrogen, tn_ug (µg/l)")) +
  guides(color=guide_legend(title="Lake Name")) + #Set legend title to "Lake Name"
  geom_boxplot(aes(color = lakename)) + #Set lake as a color aesthetic
  scale_x_discrete(drop = FALSE) #Prevent ggplot from dropping levels having no data
print(boxplotC)
```

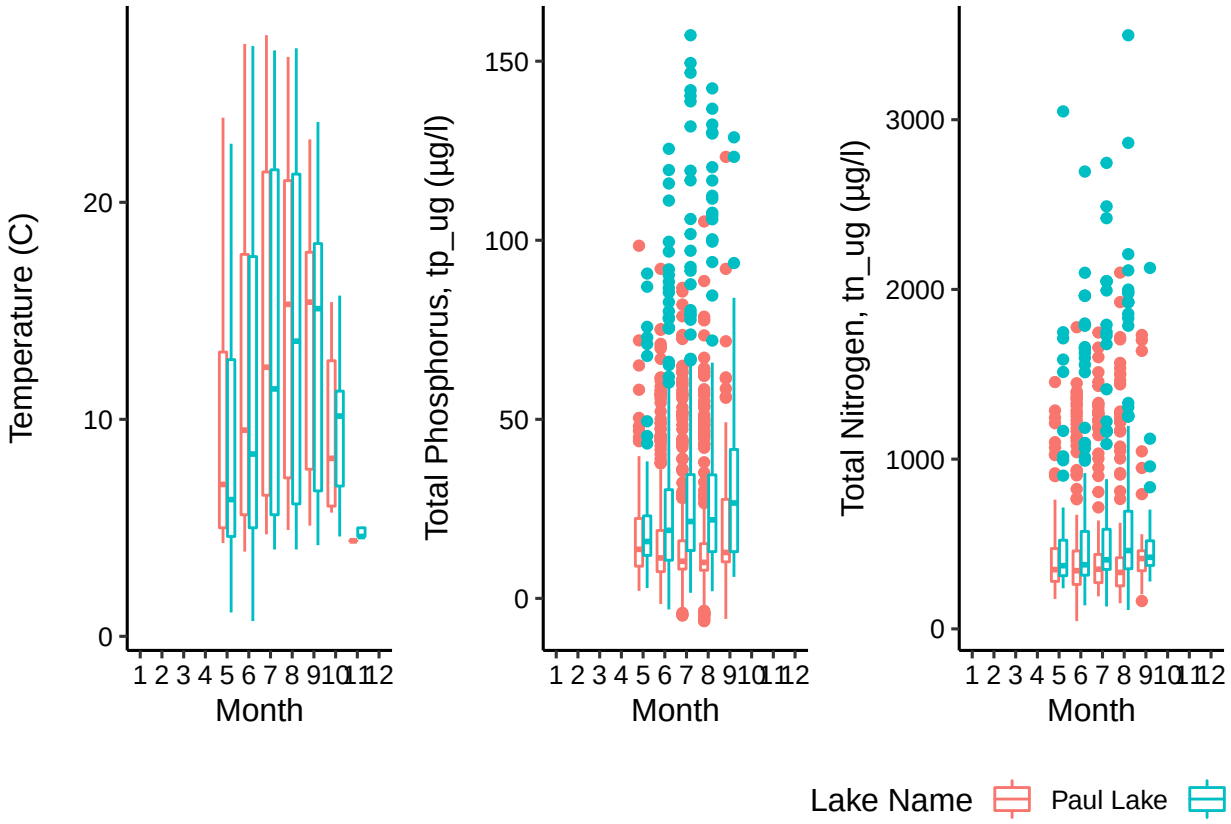
```
## Warning: Removed 21583 rows containing non-finite values (stat_boxplot).
```




```
#Make a cowplot with the three graphs - boxplotA, boxplotB, boxplotC. Make sure there is only one legen
combined_plotsABC <-
plot_grid(boxplotA + theme(legend.position = "none"), boxplotB +  theme(legend.position = "none"), boxpl
```

```
## 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).
```

```
print(combined_plotsABC)
```



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

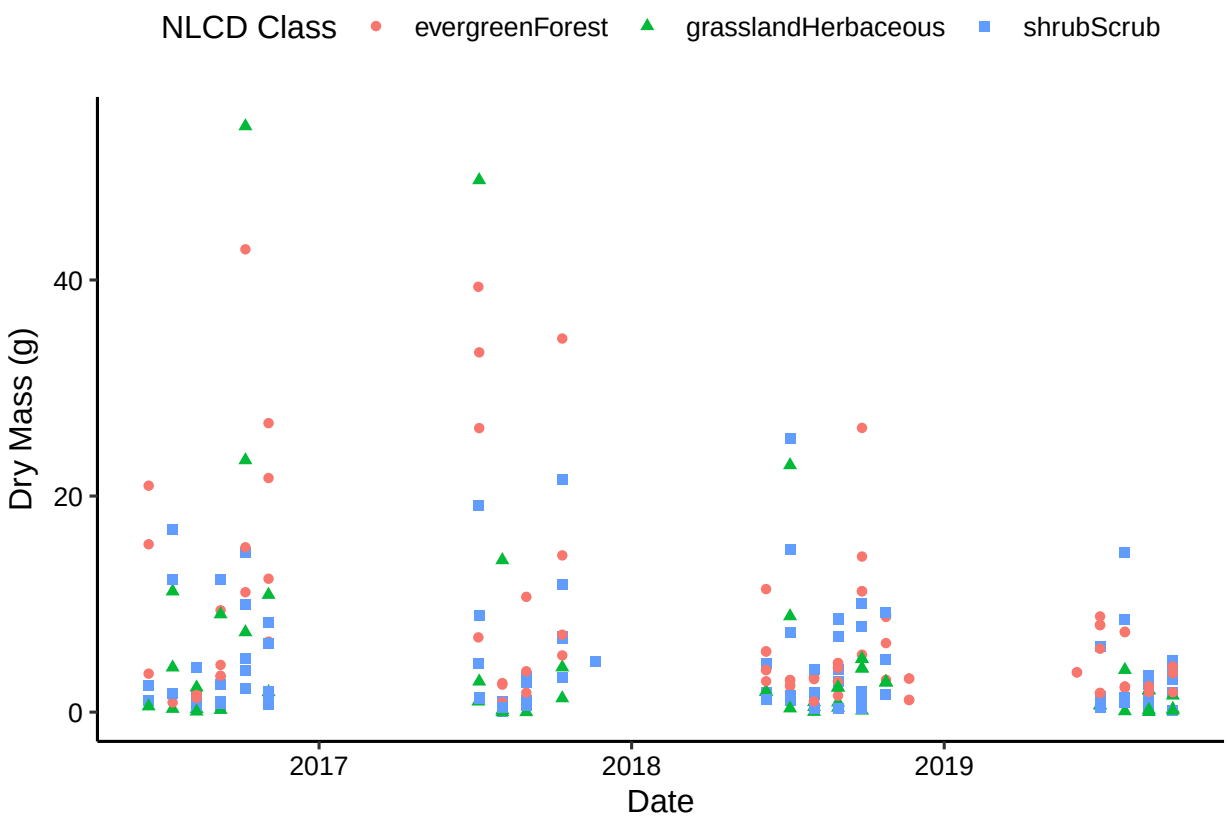
> Answer: The variables of interest - Temperature (C), Total Phosphorous (µg/l), and Nitrogen (µg/l) - follow relatively similar trends in both lakes throughout the year. The mean temperature was slightly higher in Paul Lake throughout every month except October and November, and both lakes recorded their highest temperatures in the months of August and September. The mean total phosphorous (µg/l) levels were slightly higher in Peter Lake throughout every month that was sampled. The greatest mean total phosphorous for Paul Lake was recorded in the months of May and September, while the greatest mean total phosphorous for Peter Lake was recorded in September. The mean nitrogen (µg/l) levels were also slightly higher in Peter Lake (than Paul) throughout every month that was sampled. The greatest mean nitrogen level for Paul Lake was recorded in the month of September, while the greatest mean nitrogen count for Peter Lake was recorded in August. Overall, it appears that the months with warmer temperatures generate higher levels of total phosphorous and nitrogen.

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)

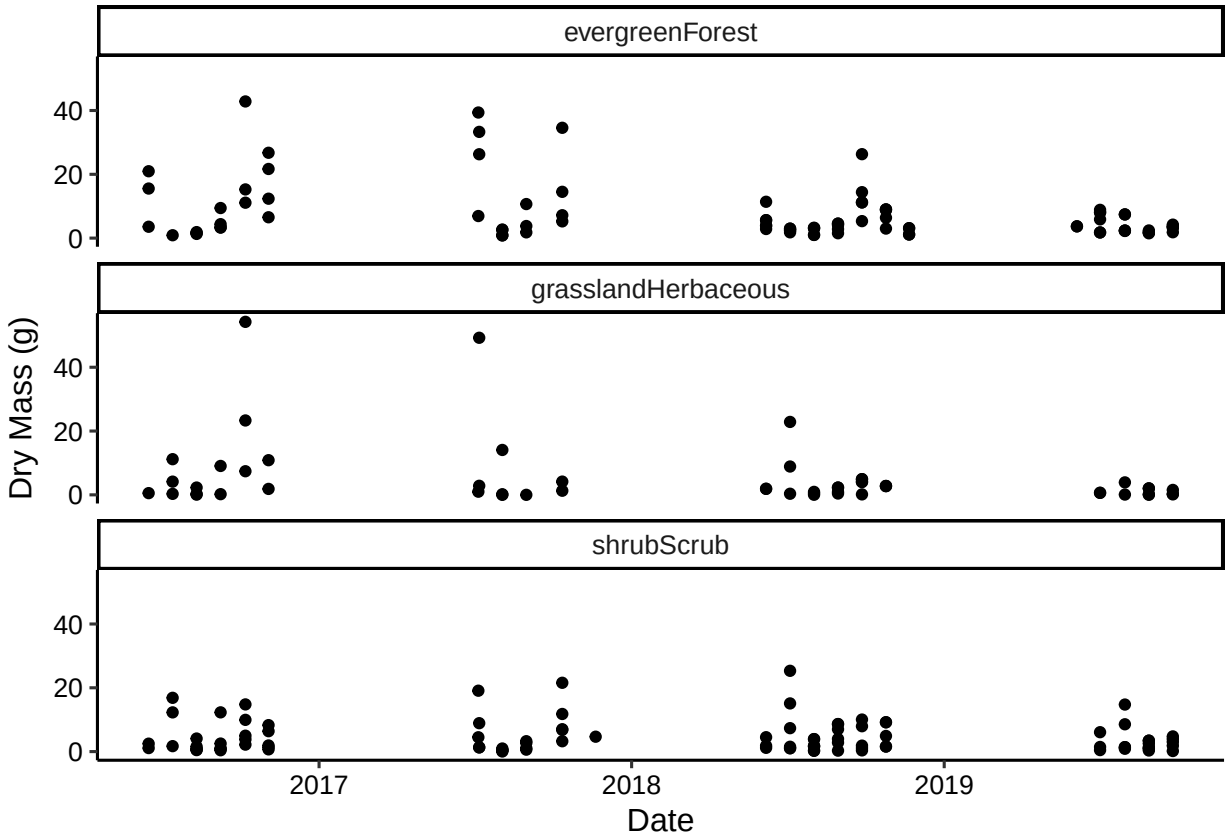
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
#6
#Plot dry mass of needle litter by date
drymass_by_date <-
  ggplot(NiwotRidge.litter[NiwotRidge.litter$functionalGroup %in% "Needles",], aes(x = collectDate, y =
  xlab(expression("Date")) + #Set axis labels
  ylab(expression("Dry Mass (g)")) +
```

```
  guides(color=guide_legend(title="NLCD Class")) + #Set legend title
  guides(shape=guide_legend(title="NLCD Class")) +
  geom_point(aes(color = nlcdClass, shape = nlcdClass)) #Separate by NLCD class with a color aesthetic
  print(drymass_by_date)
```




```
#7
#Plot dry mass of needle litter by date and separate NLCD classes into three facets
drymass_by_date2 <-
  ggplot(NiwotRidge.litter[NiwotRidge.litter$functionalGroup %in% "Needles",], aes(x = collectDate, y =
  geom_point() +
  facet_wrap(vars(nlcdClass), nrow = 3) + #Separate NLCD classes into three facets
    xlab(expression("Date")) + #Set axis labels
  ylab(expression("Dry Mass (g)"))
  print(drymass_by_date2)
```



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

Answer: After comparing the plots (6 vs. 7), I think the plot in #7 is more effective in showing the comparison of trends between the three NLCD classes in the "Needles" functional group. The 'facet_wrap' function generate multi-panel plots, which was a better way to display this dataset rather than having all the classes mixed together (as in #6). It is helpful to be able to visualize the individual trends of dry mass in each NLCD class, but then we can also make comparsions between all three classes by layering the individual plots on top of each other (with 'facet_wrap').