Assignment 5: Data Visualization

Kim Myers

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., "Salk_A05_DataVisualization.Rmd") prior to submission.

The completed exercise is due on Tuesday, February 11 at 1:00 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 (tidy and gathered) 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.

```
#1
getwd()
```

```
## [1] "C:/Users/Temp/Documents/Duke/S20/DataAnalytics/Environmental_Data_Analytics_2020/Assignments"
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(cowplot)
```

```
##
## ********************************************************

## Note: As of version 1.0.0, cowplot does not change the
##   default ggplot2 theme anymore. To recover the previous
##   behavior, execute:
##   theme_set(theme_cowplot())
## ********************************************************
```

```r
peterpaul <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")
niwot <- read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")
#2
str(peterpaul)
```

```
## 'data.frame':    23008 obs. of  15 variables:
##  $ lakename       : Factor w/ 2 levels "Paul Lake","Peter Lake": 1 1 1 1 1 1 1 1 1 1 ...
##  $ year4          : int  1984 1984 1984 1984 1984 1984 1984 1984 1984 1984 ...
##  $ daynum         : int  148 148 148 148 148 148 148 148 148 148 ...
##  $ month          : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ sampledate     : Factor w/ 1103 levels "1984-05-27","1984-05-28",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ depth          : num  0 0.25 0.5 0.75 1 1.5 2 3 4 5 ...
##  $ temperature_C  : num  14.5 NA NA NA 14.5 NA 14.2 11 7 6.1 ...
##  $ dissolvedOxygen: num  9.5 NA NA NA 8.8 NA 8.6 11.5 11.9 2.5 ...
##  $ irradianceWater: num  1750 1550 1150 975 870 610 420 220 100 34 ...
##  $ irradianceDeck : num  1620 1620 1620 1620 1620 1620 1620 1620 1620 1620 ...
##  $ tn_ug          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ tp_ug          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ nh34           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ no23           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ po4            : num  NA NA NA NA NA NA NA NA NA NA ...
```

```r
str(niwot)
```

```
## 'data.frame':    1692 obs. of  13 variables:
##  $ plotID          : Factor w/ 12 levels "NIWO_040","NIWO_041",..: 9 8 9 11 7 7 4 4 4 4 ...
##  $ trapID          : Factor w/ 15 levels "NIWO_040_139",..: 11 10 11 13 9 9 5 5 5 5 ...
##  $ collectDate     : Factor w/ 24 levels "2016-06-16","2016-07-14",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ functionalGroup : Factor w/ 8 levels "Flowers","Leaves",..: 6 5 8 6 4 2 2 6 7 8 ...
##  $ dryMass         : num  0 0.27 0.12 0 1.11 0 0 0 0.07 0.02 ...
##  $ qaDryMass       : Factor w/ 2 levels "N","Y": 1 1 1 1 2 1 1 1 1 1 ...
##  $ subplotID       : int  31 41 31 32 32 32 40 40 40 40 ...
##  $ decimalLatitude : num  40.1 40 40.1 40 40 ...
##  $ decimalLongitude: num  -106 -106 -106 -106 -106 ...
##  $ elevation       : num  3477 3413 3477 3373 3446 ...
##  $ nlcdClass       : Factor w/ 3 levels "evergreenForest",..: 3 1 3 1 3 3 2 2 2 2 ...
##  $ plotType        : Factor w/ 1 level "tower": 1 1 1 1 1 1 1 1 1 1 ...
##  $ geodeticDatum   : Factor w/ 1 level "WGS84": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
peterpaul$sampledate <- as.Date(peterpaul$sampledate, format="%Y-%m-%d")
niwot$collectDate <- as.Date(niwot$collectDate, format = "%Y-%m-%d")
```

Define your theme

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

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

```r
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 by phosphate, 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.

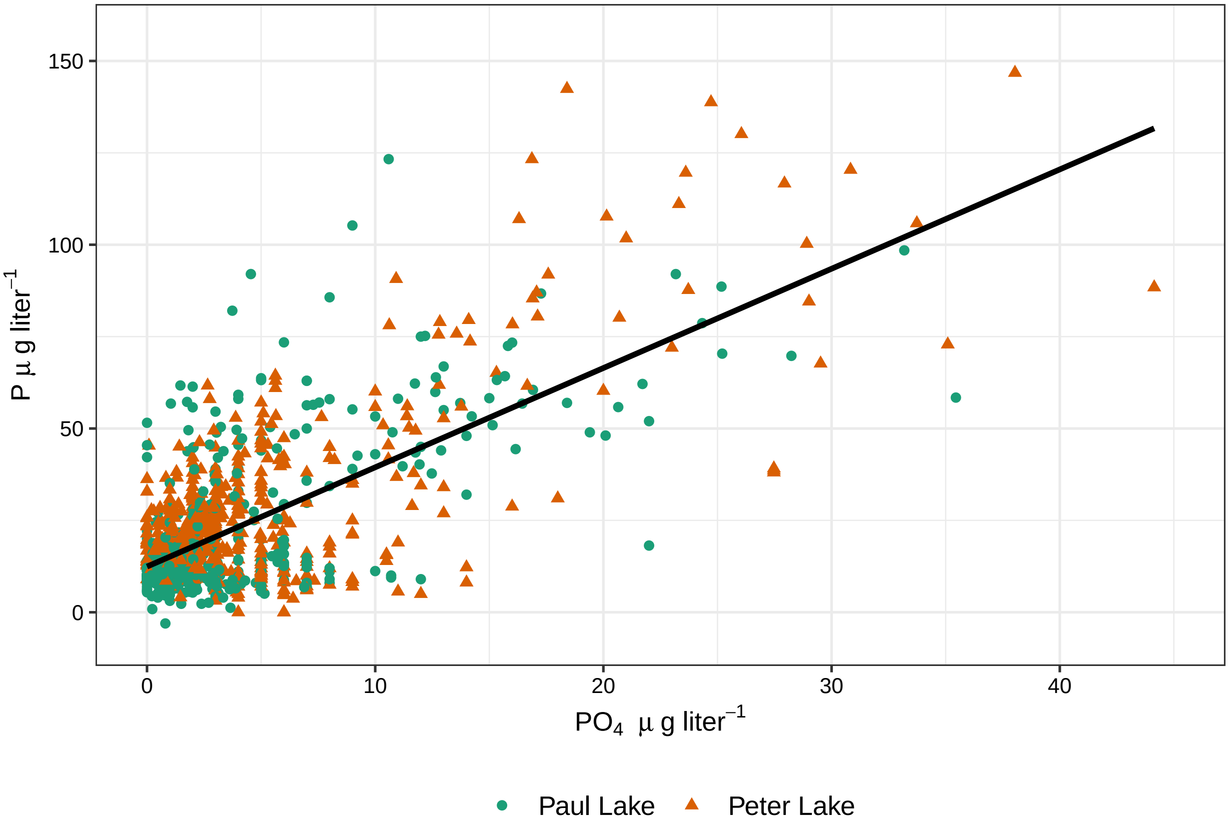
```r
#4
library(RColorBrewer)

phos <- ggplot(peterpaul, aes(x=po4, y=tp_ug, color=lakename)) + geom_point(aes(shape=lakename))+ geom_s

print(phos)
```

```
## Warning: Removed 21947 rows containing non-finite values (stat_smooth).
```

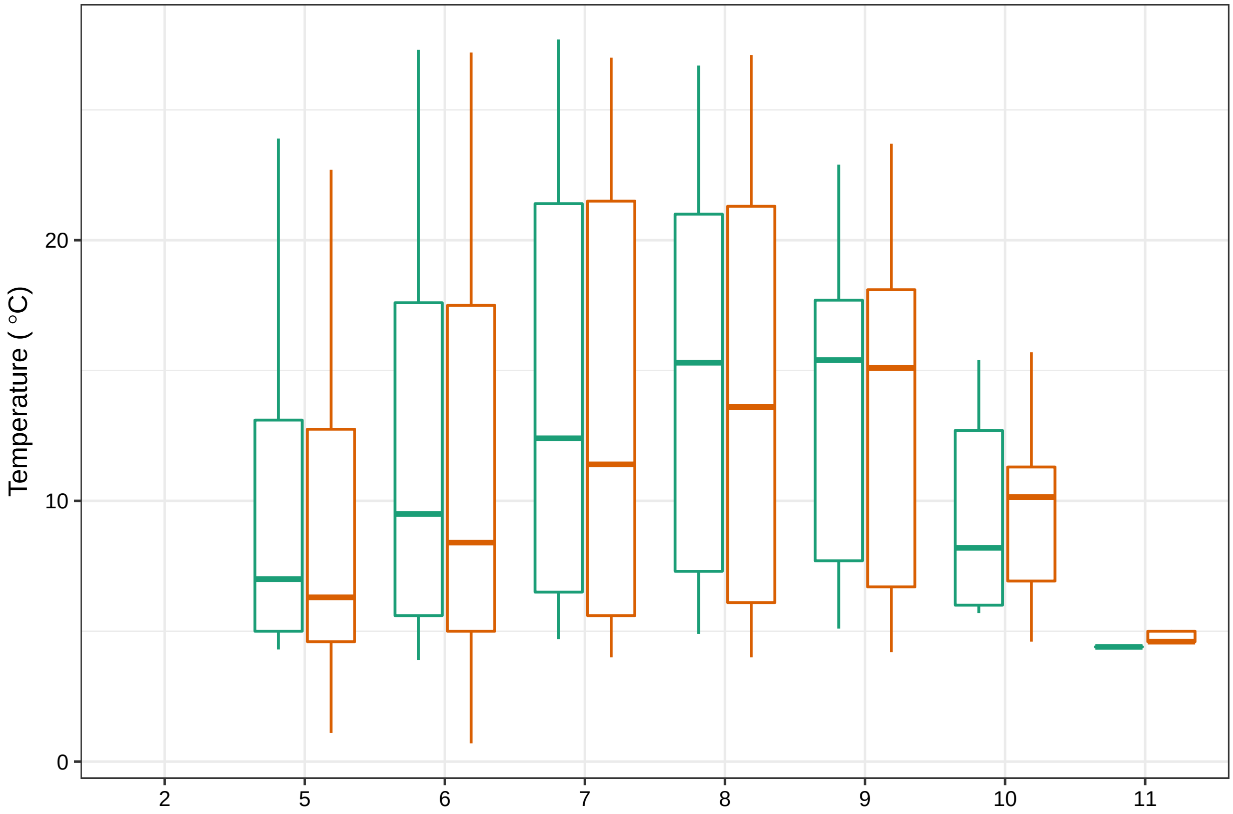
```
## Warning: Removed 21947 rows containing missing values (geom_point).
```



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.

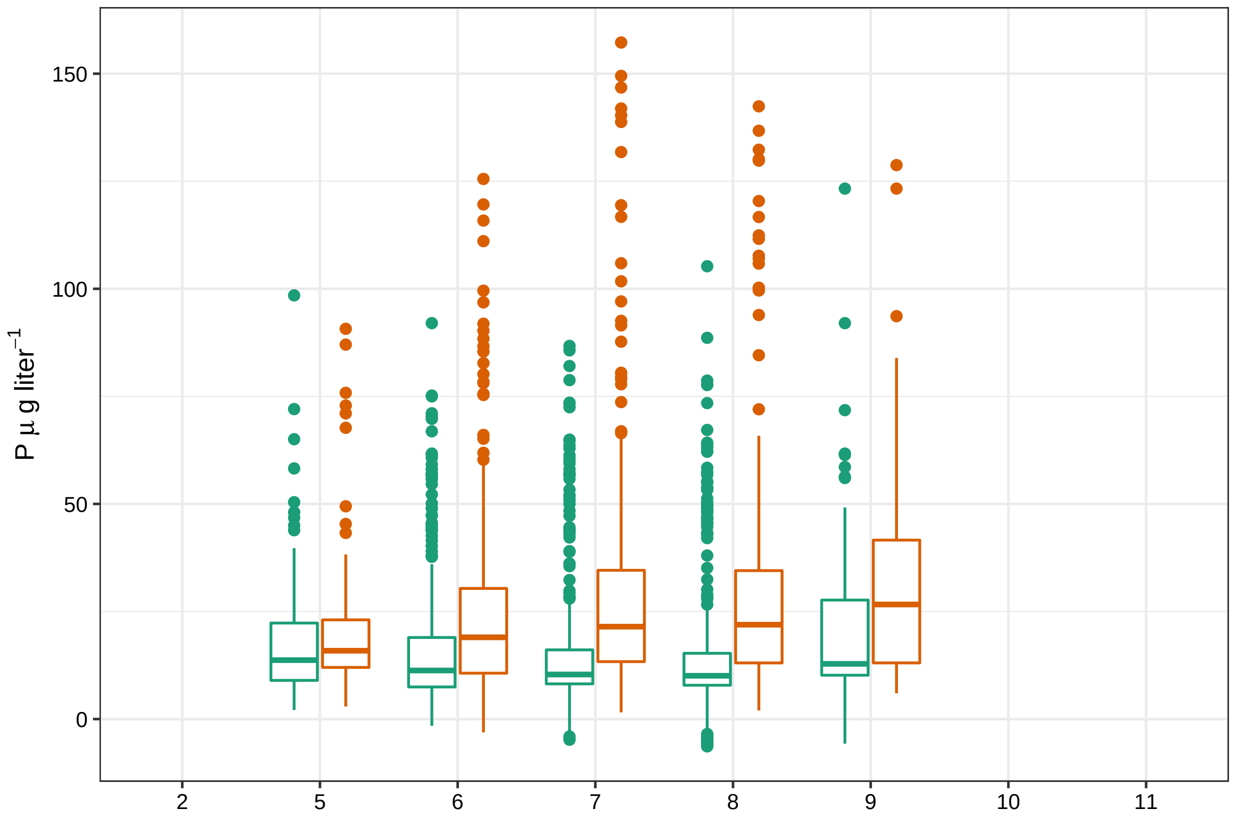
```
#a
temp <- ggplot(peterpaul, aes(x=factor(month), y=temperature_C, color=lakename)) + geom_boxplot() + labs
print(temp)
```

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



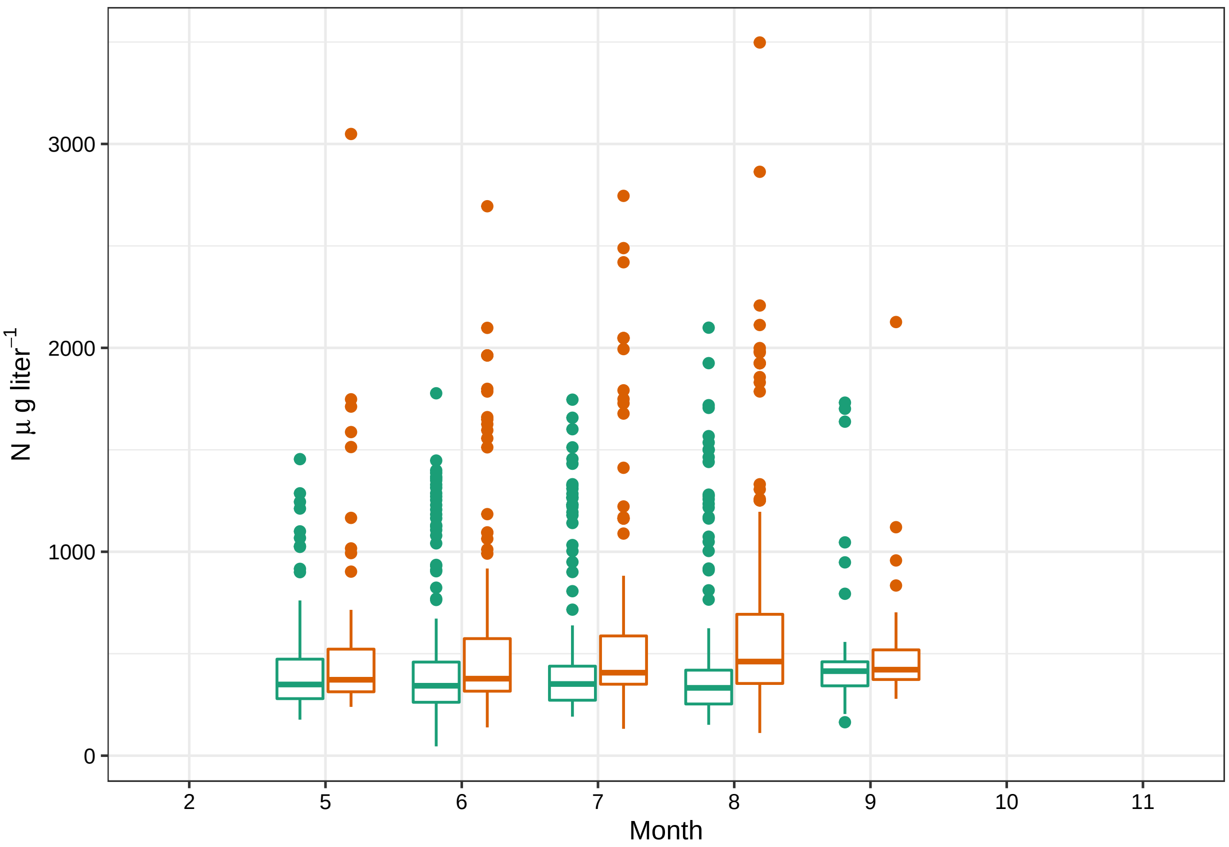
```
#b
tphos <- ggplot(peterpaul, aes(x=factor(month), y=tp_ug, color=lakename)) + geom_boxplot() + labs(y=expr
print(tphos)
```

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



```
#c
tnitro <- ggplot(peterpaul, aes(x=factor(month), y=tn_ug, color=lakename)) + geom_boxplot() + labs(y=exp
print(tnitro)
```

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



```
#d
##extract legnd
legend <- get_legend(temp + theme(legend.position="top",legend.direction = "vertical"))
```

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

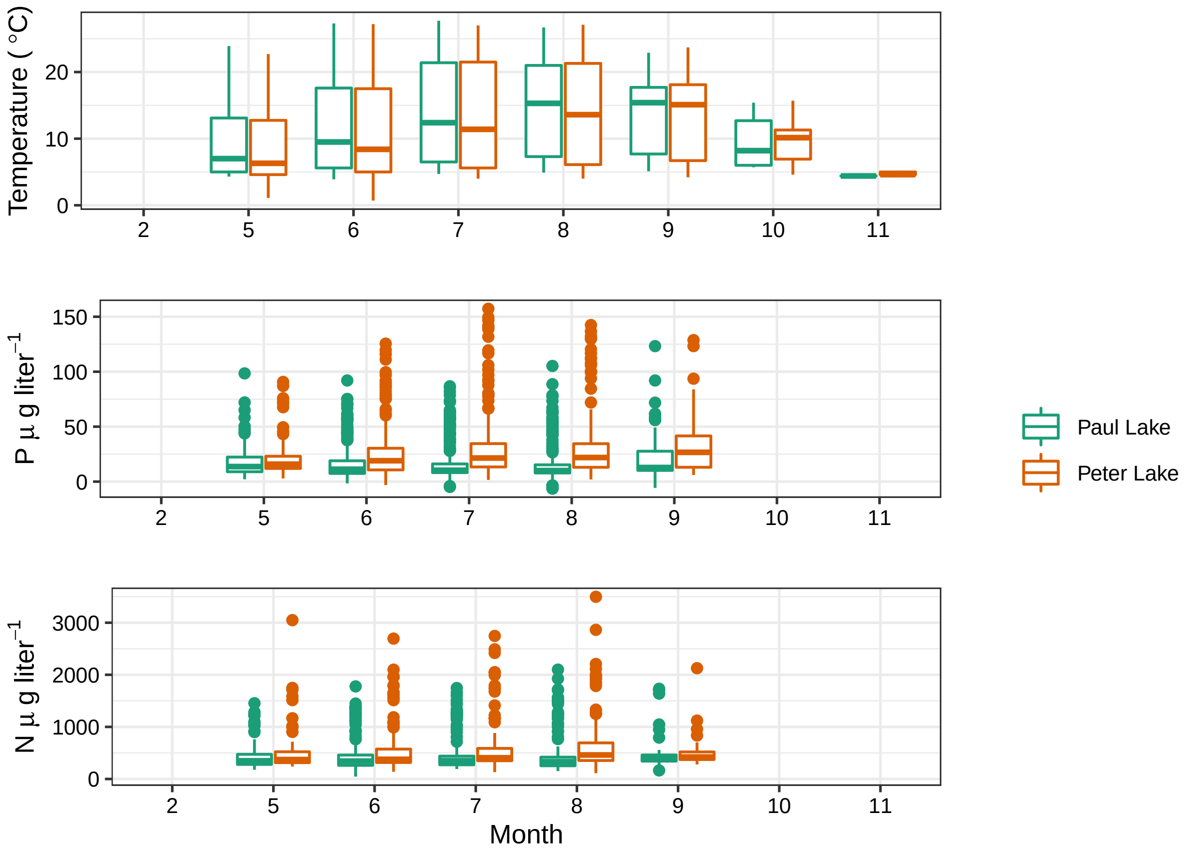
```
##create gridded figure
nptemp <- plot_grid(temp, tphos, tnitro, nrow = 3, align = 'h')
```

```
## 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(nptemp)
##add common legend to grid
nptemp_legend <- plot_grid(nptemp, legend, ncol = 2, rel_widths = c(2, .6))
print(nptemp_legend)
```

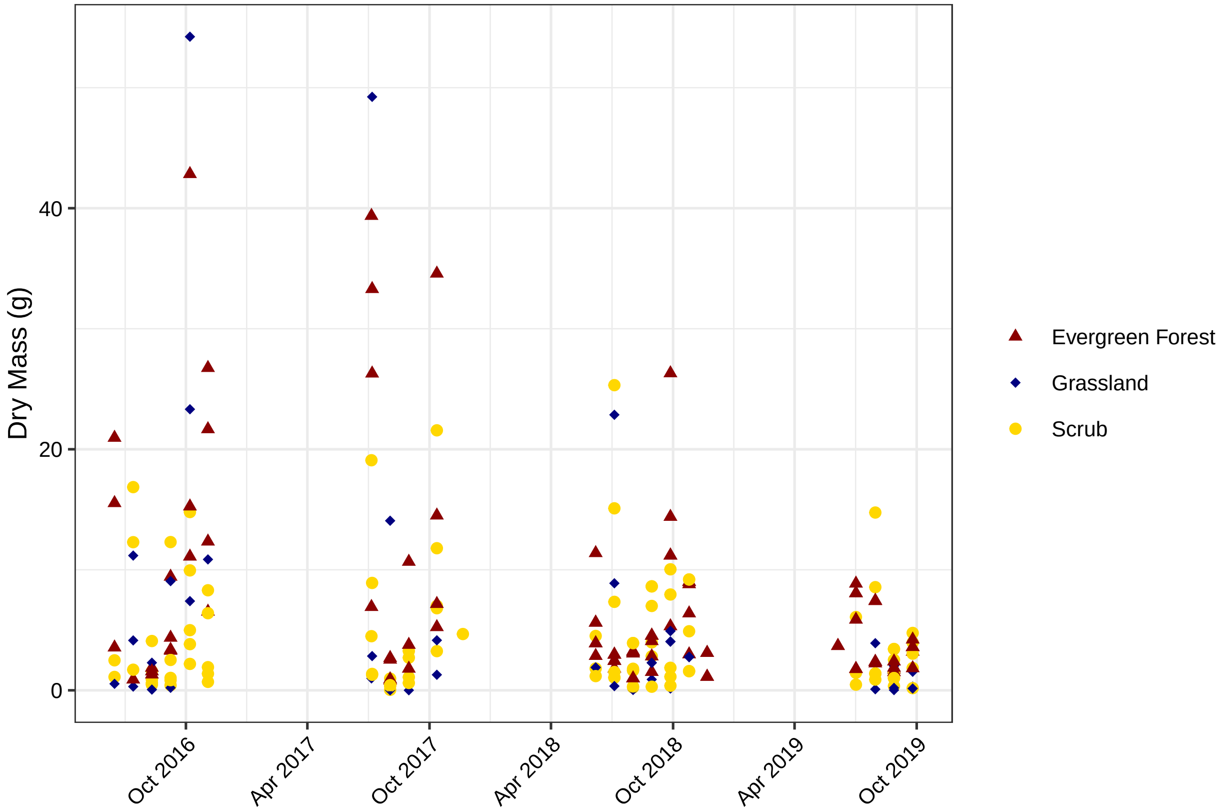


Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Generally, temperature increases similarly in both lakes in the summer and decreases in the winter. However, Peter Lake has consistently greater concentrations of nitrogen and phosphorus yearround than those of Paul Lake. These concentrations are relatively constant throughout the year.

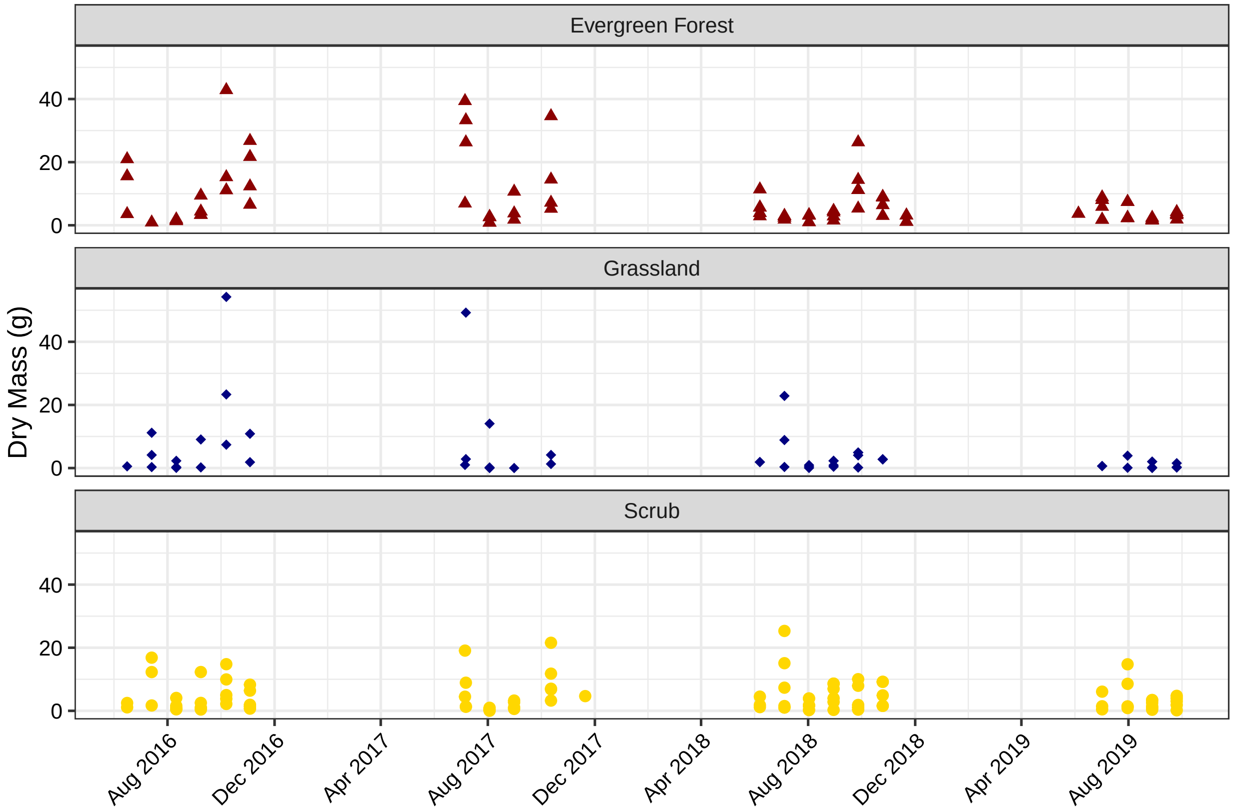
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
#6
needles <-
  ggplot(subset(niwot, functionalGroup == "Needles"),
         aes(x=collectDate,y=dryMass,color=nlcdClass,
        shape=nlcdClass)) +
        geom_point() +
        labs(x=" ",y="Dry Mass (g)", color="", shape="")+
        theme(legend.direction = "vertical",
        legend.position = "right",
        axis.text.x=element_text(angle=45, hjust=1)) +
        scale_color_manual(values=c("darkred", "navy","gold"),
        labels = c("Evergreen Forest","Grassland","Scrub")) +
        scale_shape_manual(values=c(17, 18,19),
        labels = c("Evergreen Forest","Grassland","Scrub")) +
        scale_x_date(date_breaks = "6 months", date_labels =  "%b %Y")
print(needles)
```



```
#7
nlcd.labs <- c("Evergreen Forest","Grassland","Scrub")
names(nlcd.labs) <- c("evergreenForest","grasslandHerbaceous","shrubScrub")

needles_nlcd <-
  ggplot(subset(niwot, functionalGroup == "Needles")) +
  geom_point(aes(x=collectDate,y=dryMass,color=nlcdClass, shape=nlcdClass)) +
  labs(x=" ",y="Dry Mass (g)", color="", shape="") +
  facet_wrap(vars(nlcdClass), nrow = 3,
      labeller = labeller(nlcdClass = nlcd.labs)) +
  scale_x_date(date_breaks = "4 months", date_labels =  "%b %Y") +
  theme(legend.position = "none", axis.text.x = element_text(angle=45, hjust=1)) +
  scale_color_manual(values=c("darkred", "navy","gold")) +
  scale_shape_manual(values=c(17, 18,19))
print(needles_nlcd)
```



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

> Answer: I think plot 7 is more effective because it's difficult to determine trends within nlcd cover types when they're all overlapping, even when they're differentiated by color.