Assignment 5: Data Visualization

Nikki Egna

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] "/Users/nikkiegna/Desktop/Classes/Spring 2020/Environmental Data Analytics/Environmental_Data_Ana
```

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


chemistry_nutrients <- read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processe
chemistry_physics <- read.csv("../Data/Processed/NTL-LTER_Lake_ChemistryPhysics_PeterPaul_Processed.csv
nutrients <- read.csv("../Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv")
gathered <- read.csv("../Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaulGathered_Processed.csv")
summary <- read.csv("../Data/Processed/NTL-LTER_Lake_Summaries_PeterPaul_Processed.csv")

#2
class(chemistry_nutrients$sampledate)
```

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

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

class(chemistry_physics$sampledate)
```

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

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

class(nutrients$sampledate)
```

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

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

class(gathered$sampledate)
```

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

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

Define your theme

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

```
mytheme <- theme_classic(base_size = 14) +
  theme(legend.position = "right",
        legend.text = element_text(size = 12), legend.title =
          element_text(size = 12), axis.text = element_text(color = "black"))

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.

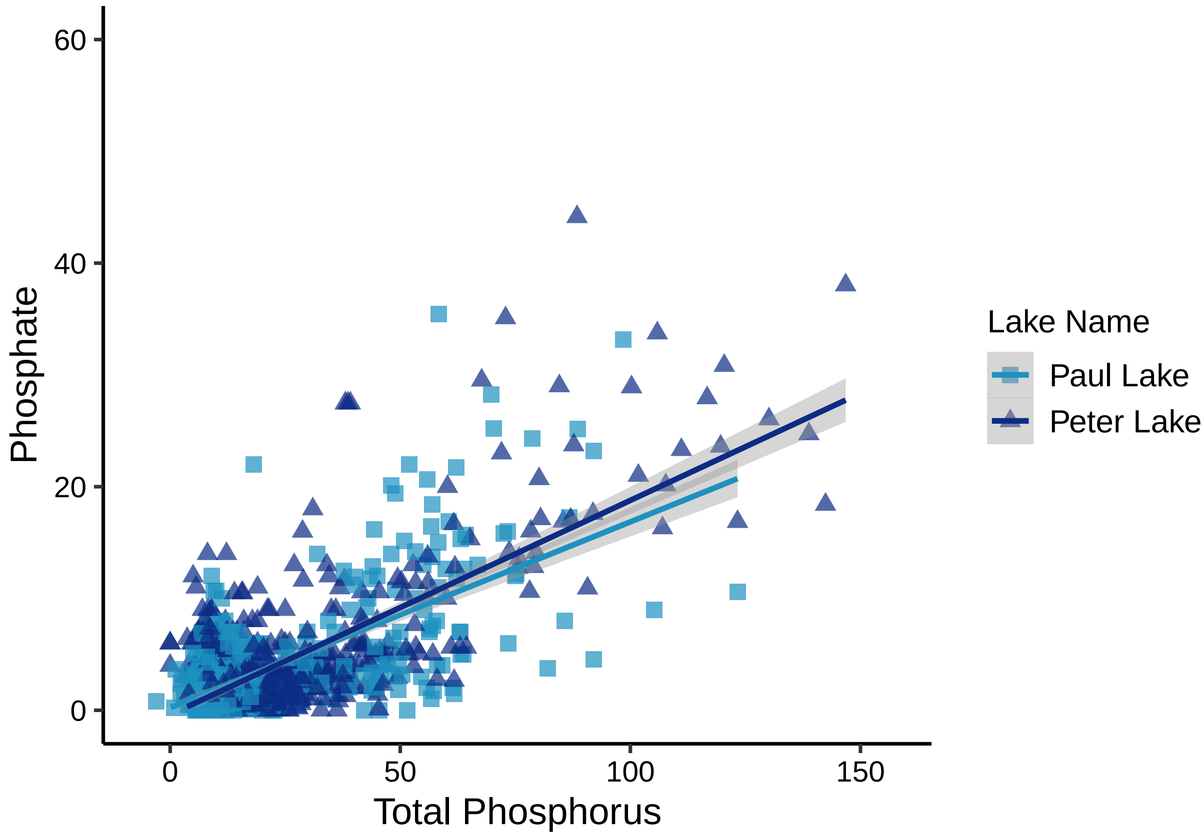
```
phosphorus_graph<-
  ggplot(nutrients, aes(x = tp_ug, y = po4, shape = lakename, color = lakename)) +
  geom_point(alpha = 0.7, size = 2.5) +
  labs(x="Total Phosphorus", y="Phosphate") +
  geom_smooth(method = lm) +
  scale_color_manual(name="Lake Name",values = c("#1d91c0", "#0c2c84")) +
  scale_shape_manual(name="Lake Name",values = c(15, 17))+
  ylim(0,60)

print(phosphorus_graph)
```

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

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

## Warning: Removed 4 rows containing missing values (geom_smooth).
```



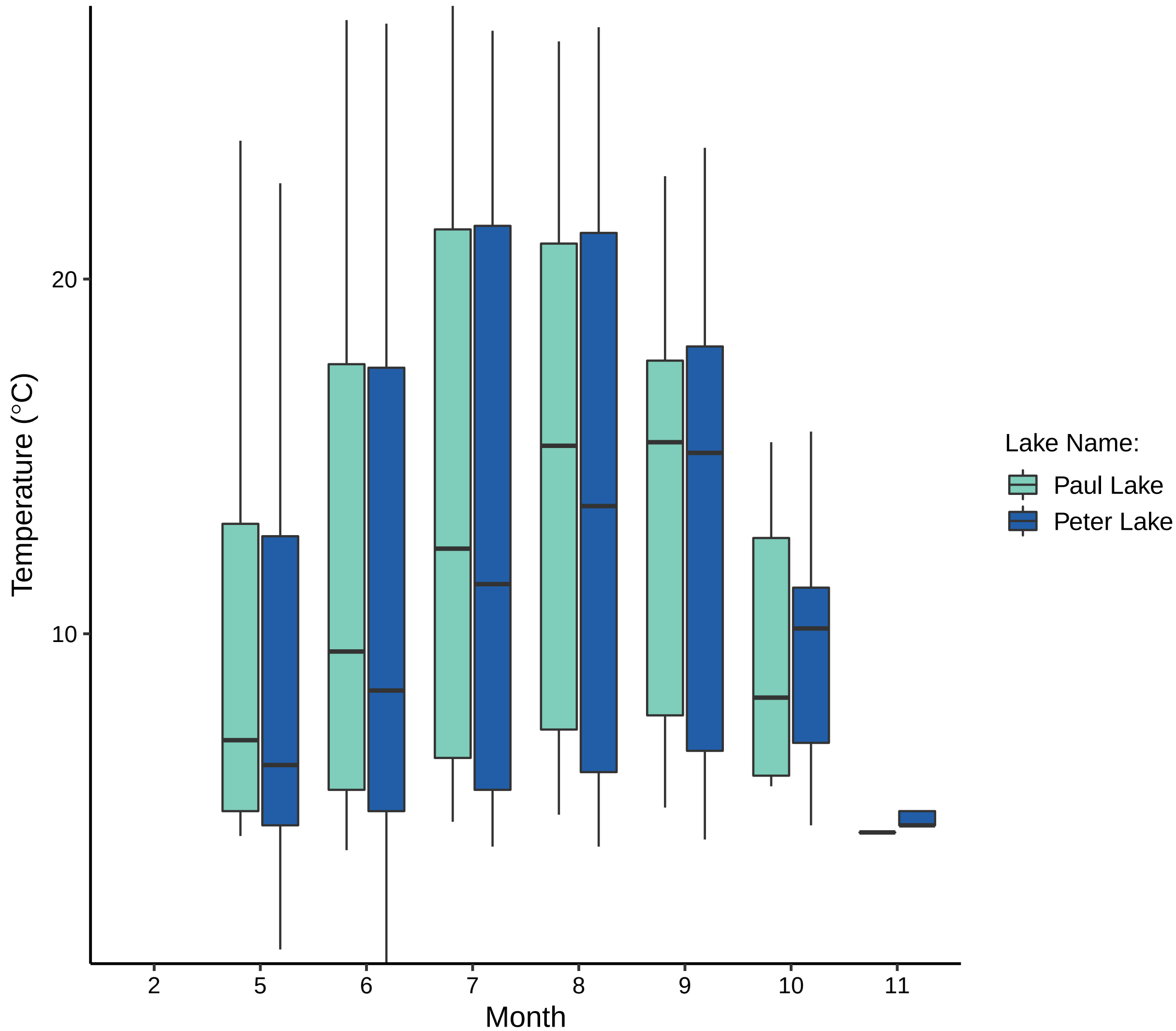
5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

```
#a
chemistry_nutrients$month <- as.factor(chemistry_nutrients$month)

tempplot <-
  ggplot(chemistry_nutrients, aes(x = month, y = temperature_C)) +
  geom_boxplot(aes(fill= lakename)) +
  labs(x= "Month", y=expression(paste("Temperature (", degree,"C)")))+
  scale_y_continuous(expand = c(0, 0))+
  scale_fill_manual(name="Lake Name:",values = c("#7fcdbb", "#225ea8"))

print(tempplot)
```

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

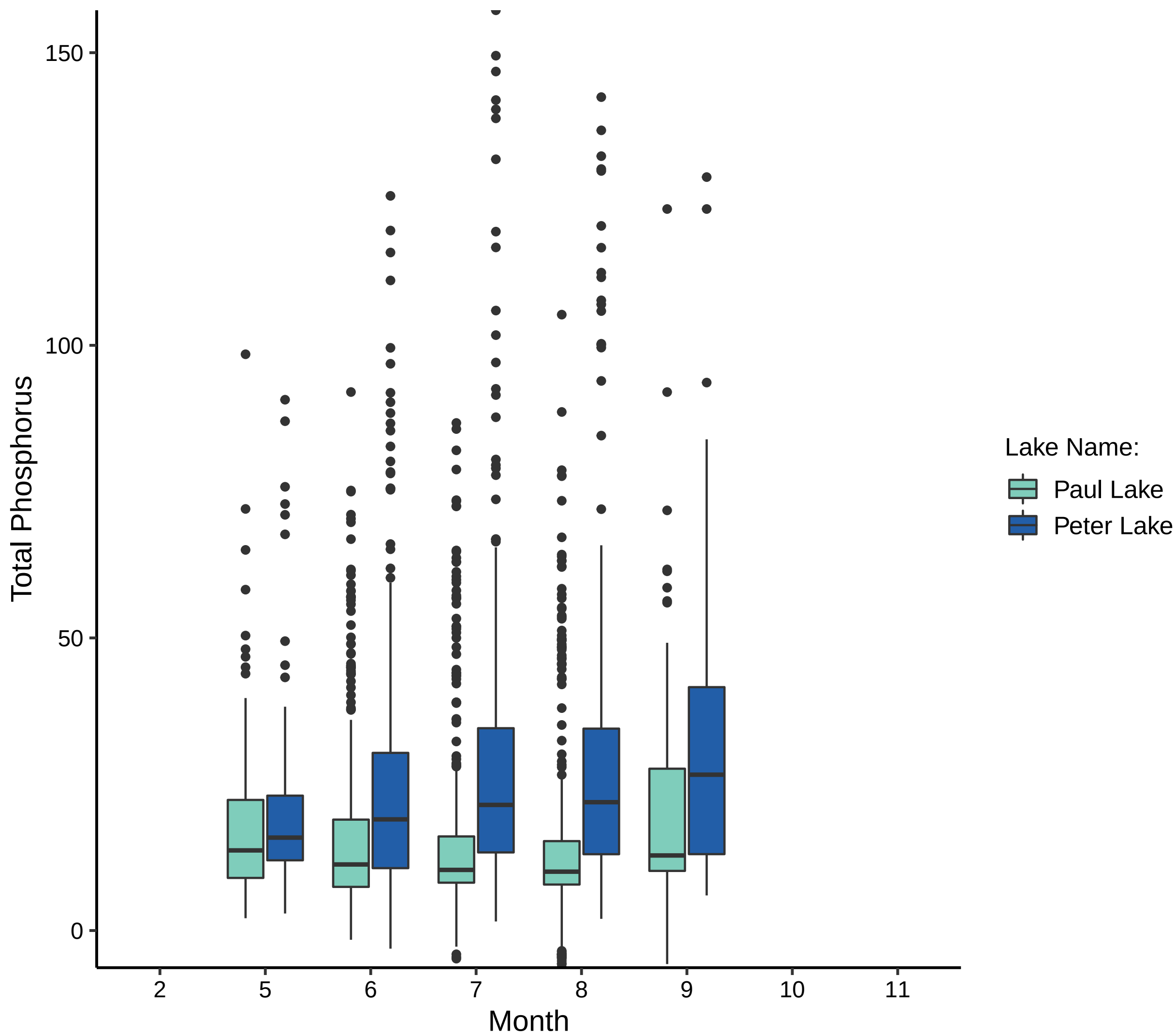



```r
#b

TPplot <-
  ggplot(chemistry_nutrients, aes(x = month, y = tp_ug)) +
  geom_boxplot(aes(fill= lakename)) +
  labs(x= "Month", y= "Total Phosphorus")+
  scale_y_continuous(expand = c(0, 0))+
  scale_fill_manual(name="Lake Name:",values = c("#7fcdbb", "#225ea8"))

print(TPplot)
```

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

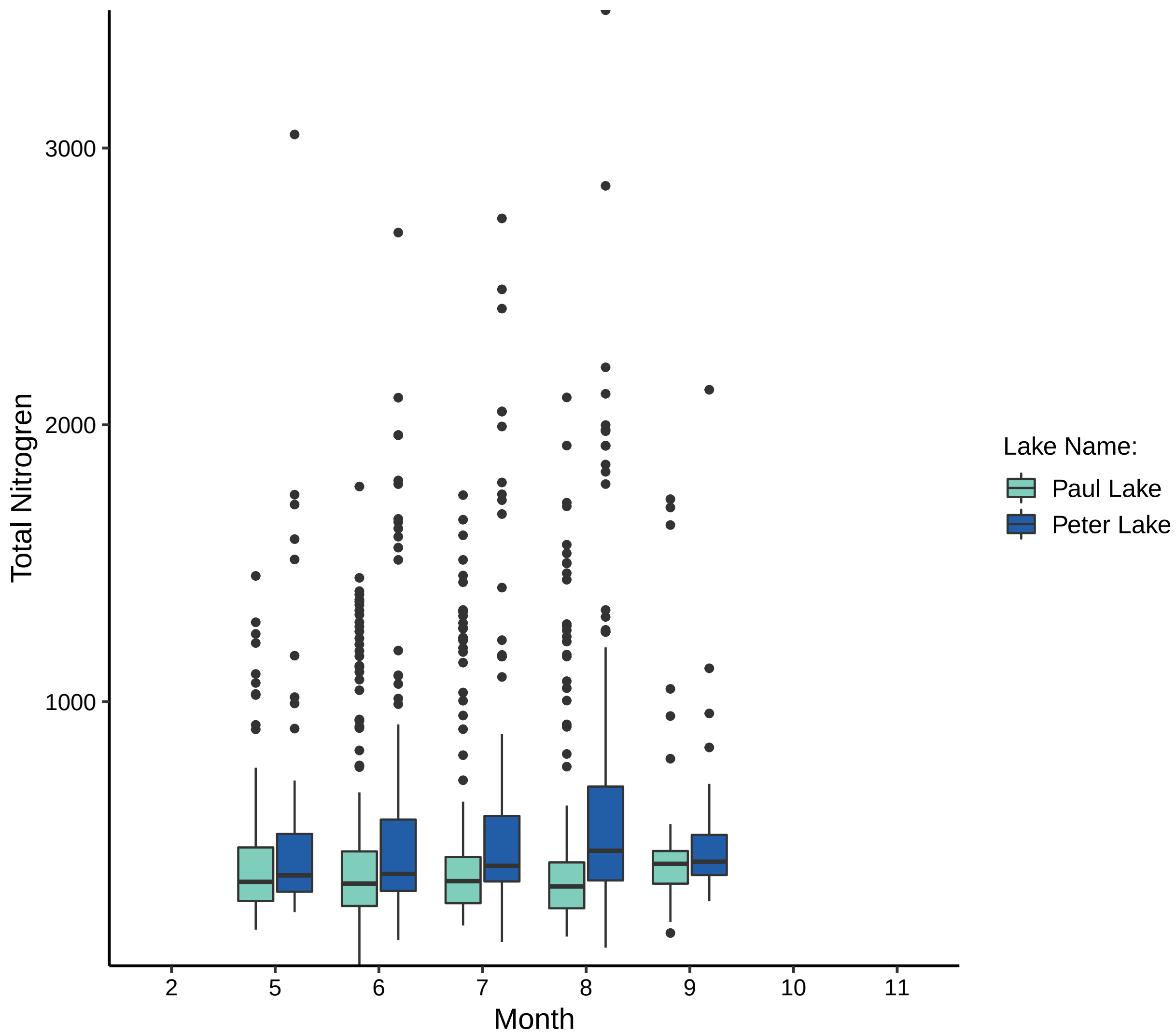



```r
#c

TNplot <-
  ggplot(chemistry_nutrients, aes(x = month, y = tn_ug)) +
  geom_boxplot(aes(fill= lakename)) +
  labs(x= "Month", y= "Total Nitrogren")+
  scale_y_continuous(expand = c(0, 0))+
  scale_fill_manual(name="Lake Name:",values = c("#7fcdbb", "#225ea8"))

print(TNplot)
```

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



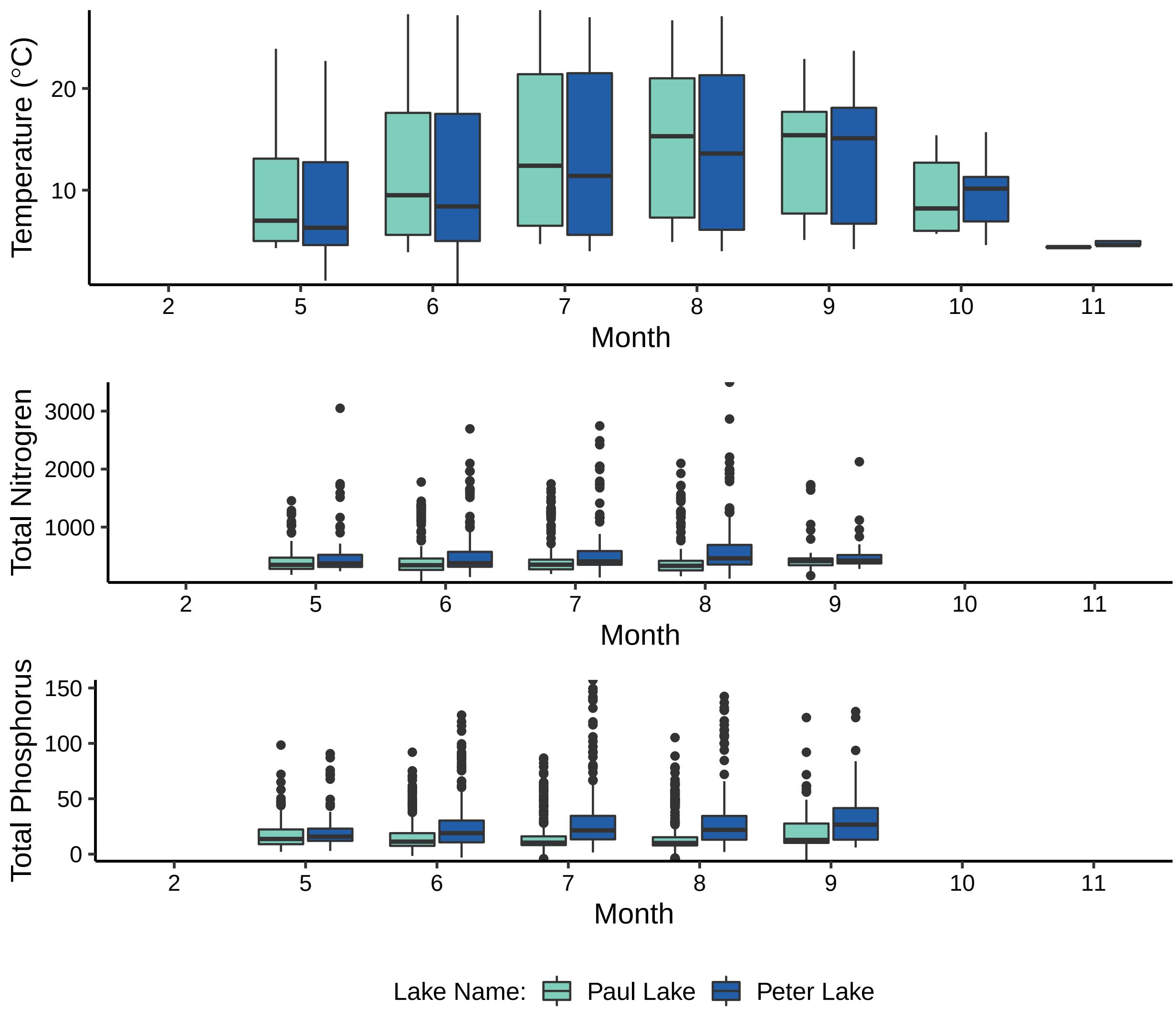

```
plot_grid(tempplot + theme(legend.position = "none"), TNplot+ theme(legend.position = "none"), TPplot+ t
```

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

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

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

## Warning: Graphs cannot be horizontally aligned unless the axis parameter is
## set. Placing graphs unaligned.
```



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

> Answer: Logically, we can see that the temperature increases in the summer months and decreases in the winter months. Nitrogren and Phosphorus levels remain relatively stable throughout the year. Both Paul and Peter Lakes have similar temperatures, however Peter Lake generally has slightly higher Nitrogen and Phosphorus levels.

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)

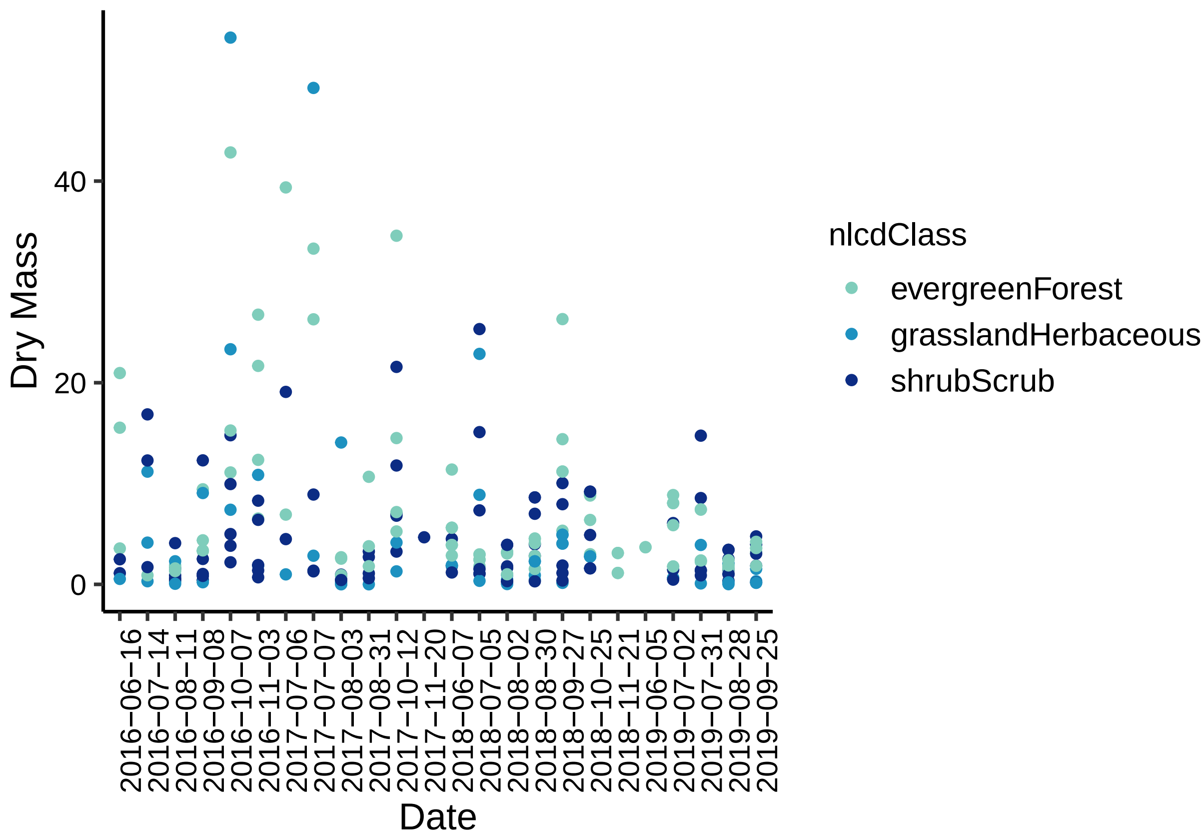
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

```
#6
Litter <- read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")

litterplot <- ggplot(subset(Litter, functionalGroup %in% "Needles"),
                     aes(x=collectDate, y=dryMass))+
```

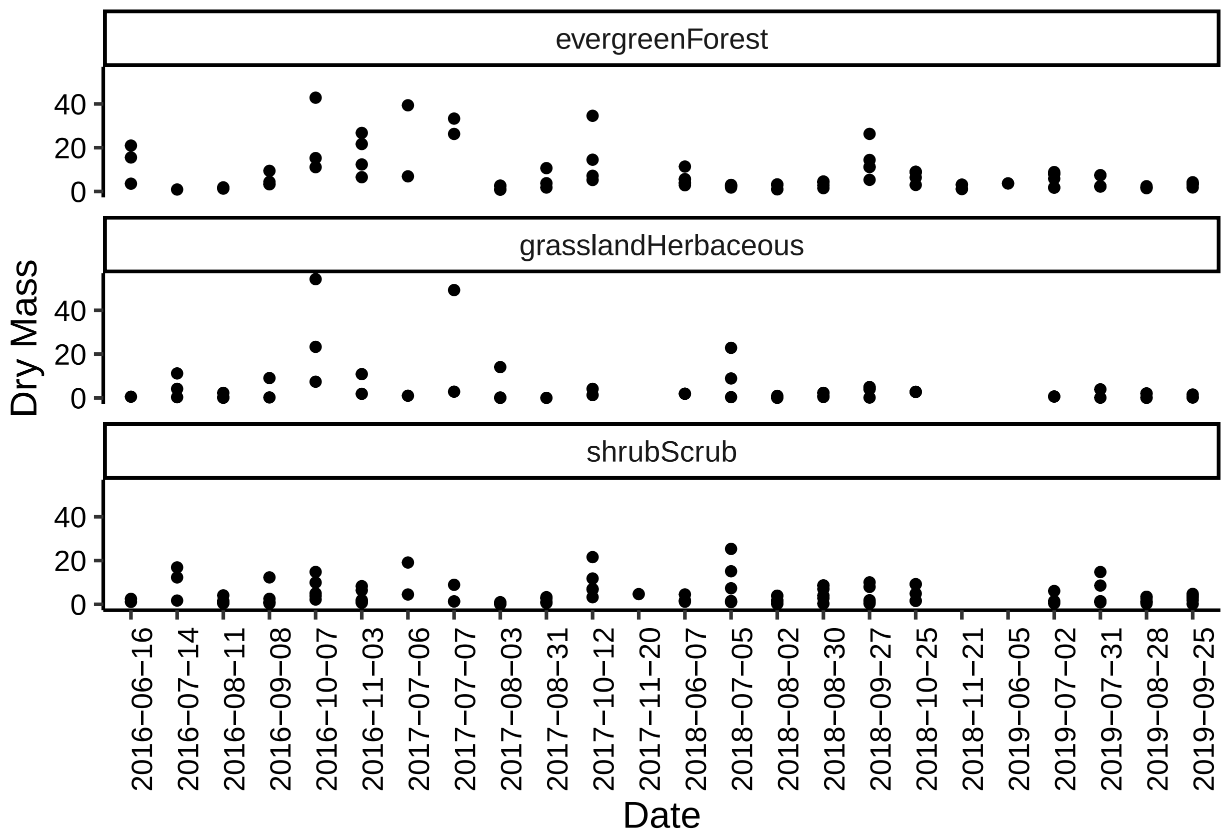
```r
  geom_point(aes(color= nlcdClass))+
  labs(x="Date", y= "Dry Mass")+
  scale_color_manual(values = c("#7fcdbb", "#1d91c0", "#0c2c84"))+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

print(litterplot)
```




```r
#7
litterplot2 <- ggplot(subset(Litter, functionalGroup %in% "Needles"),
                      aes(x=collectDate, y=dryMass))+
  geom_point()+
  labs(x="Date", y= "Dry Mass")+
  scale_color_manual(values = c( "#0c2c84"))+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  facet_wrap(vars(nlcdClass), nrow = 3)

print(litterplot2)
```



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

> Answer: I think plot 7 is better because plot 6 is very busy and is difficult to see clear trends within each NLCD class. Separating the NLCD classes allows the reader to more easily see the relationship between each class and month, and also more easily compare the three classes to one another.