Assignment 5: Data Visualization

Emily McNamara

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/emilymcnamara/Desktop/Env Data Analytics/Environmental_Data_Analytics_2020"
```

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------ tidy
```

```
## 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_
## 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
library(viridis)
```

```
## Loading required package: viridisLite
```

```r
library(RColorBrewer)
library(colormap)

PeterPaul.chem.nutrients <-
  read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")

PeterPaul.chem.nutrients.gathered <-
  read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaulGathered_Processed.csv")

NIWO.Litter <-
  read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")


#2
class(PeterPaul.chem.nutrients$sampledate)
```

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

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


class(PeterPaul.chem.nutrients.gathered$sampledate)
```

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

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

class(NIWO.Litter$collectDate)
```

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

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

Define your theme

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

```r
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")

theme_set(mytheme)
```

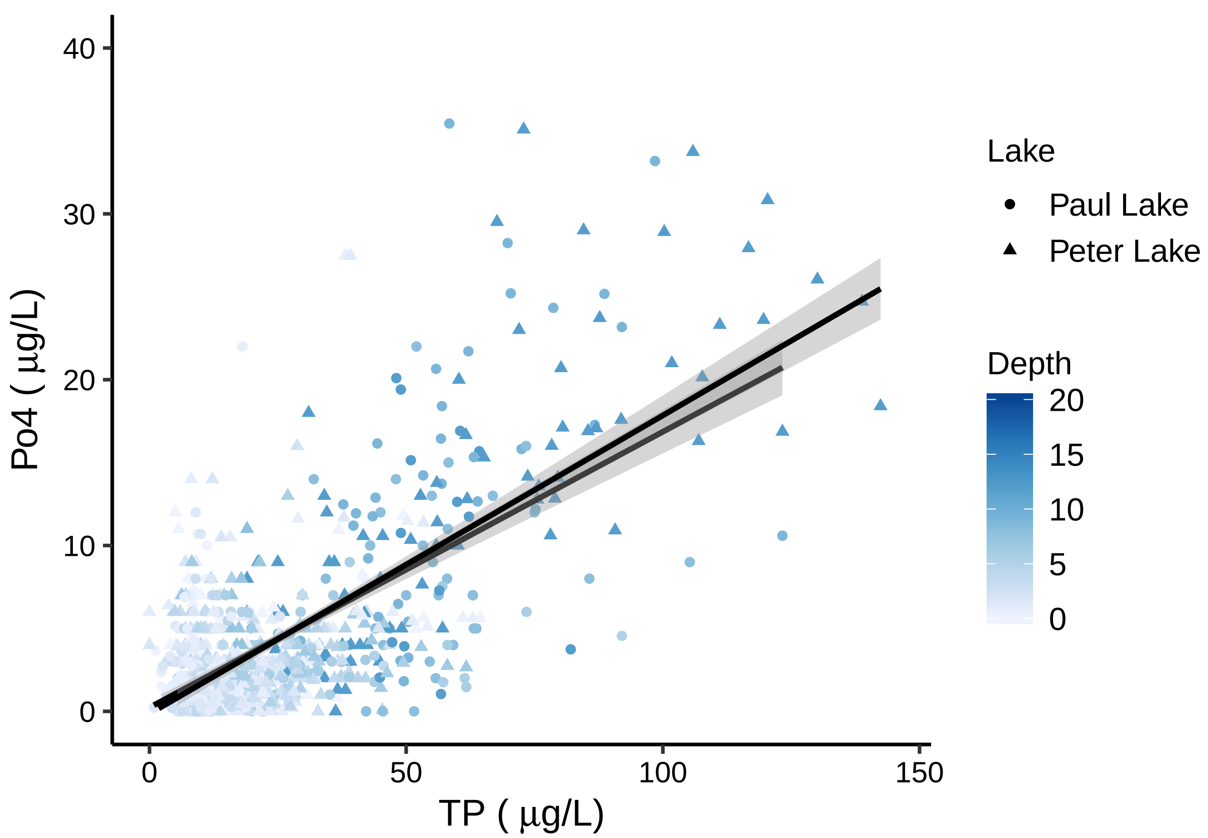
Create graphs

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.

```
NTL.TP.PO4 <-
  ggplot(PeterPaul.chem.nutrients, aes(x = tp_ug, y = po4, color = depth, shape = lakename)) +
  geom_point() +
   labs(x = expression(paste("TP ( ", mu,"g/L)")),
       y = expression(paste("Po4 ( ", mu,"g/L)")),
      color = "Depth", shape = "Lake") +
  scale_color_distiller(palette = "Blues", direction = 1) +
  scale_y_continuous(limits = c(0,40)) +
  scale_x_continuous(limits = c(0, 145)) +
  geom_smooth(method='lm', formula= y~x, color = "black") +
  theme(legend.position = "right",
        legend.text = element_text(size = 12), legend.title = element_text(size = 12))
print(NTL.TP.PO4)
```

```
## Warning: Removed 21950 rows containing non-finite values (stat_smooth).

## Warning: Removed 21950 rows containing missing values (geom_point).

## Warning: Removed 1 rows containing missing values (geom_smooth).
```

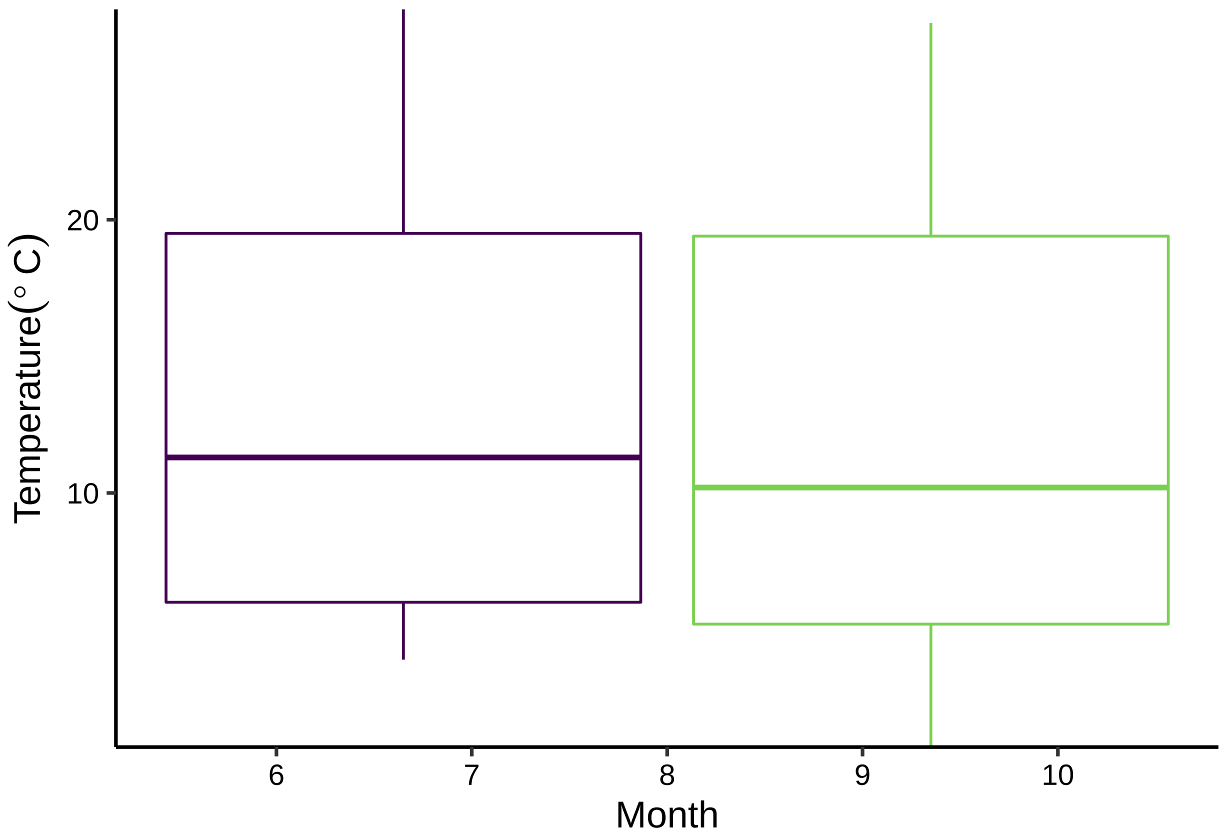


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the

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.

```
# Temperature Boxplot

Temp.Boxplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = temperature_C, color = lakename)) +
  geom_boxplot() +
  labs(x = expression(paste("Month")),
       y = expression(paste("Temperature" (degree~C))), color = "Lake") +
  scale_y_continuous(expand = c(0,0)) +
  scale_color_viridis(discrete = TRUE, option = "viridis", end = 0.8) +
  theme(legend.position = "none")
print(Temp.Boxplot)
```

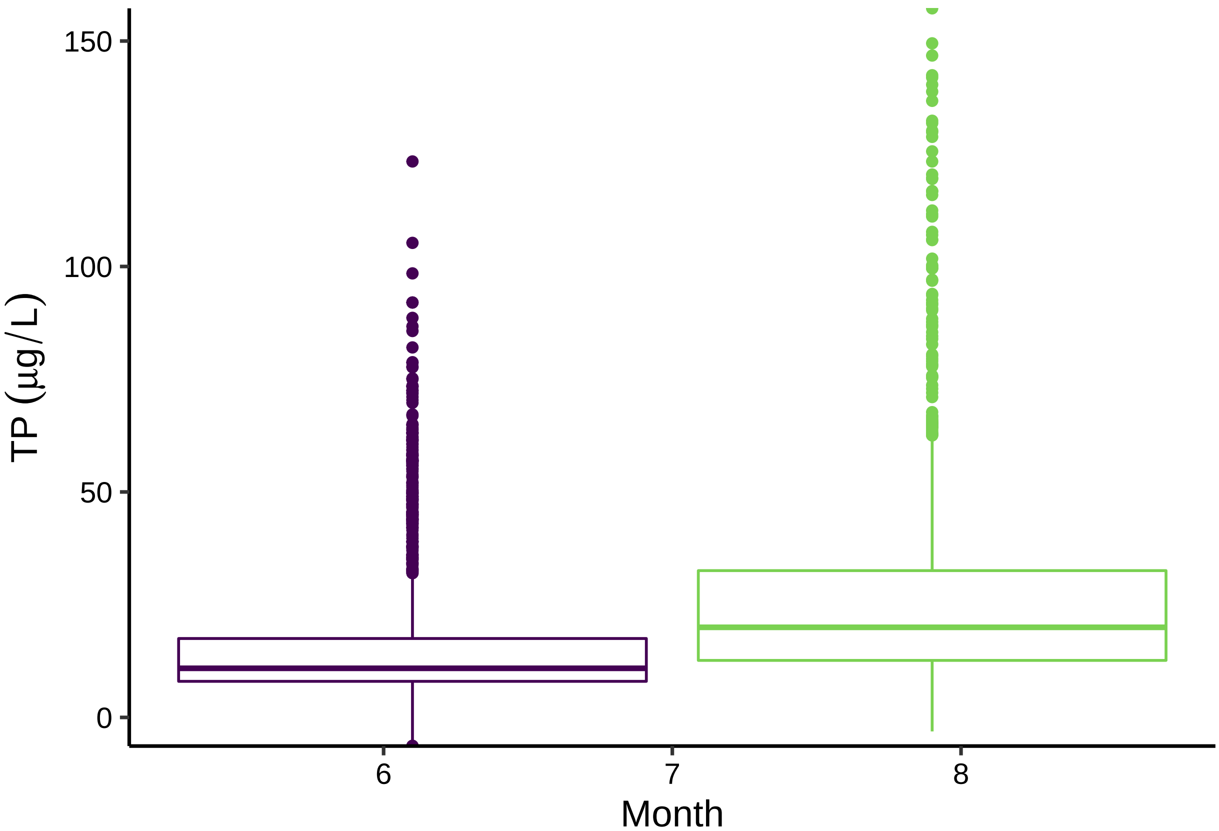
```
## Warning: Removed 3566 rows containing non-finite values (stat_boxplot).
```




```
# TP Boxplot

TP.Boxplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tp_ug, color = lakename)) +
  geom_boxplot() +
  labs(x = expression(paste("Month")),
       y = expression(paste("TP"~ (mu*g / L))), color = "Lake") +
  scale_y_continuous(expand = c(0,0)) +
  scale_color_viridis(discrete = TRUE, option = "viridis", end = 0.8) +
  theme(legend.position = "none")
```

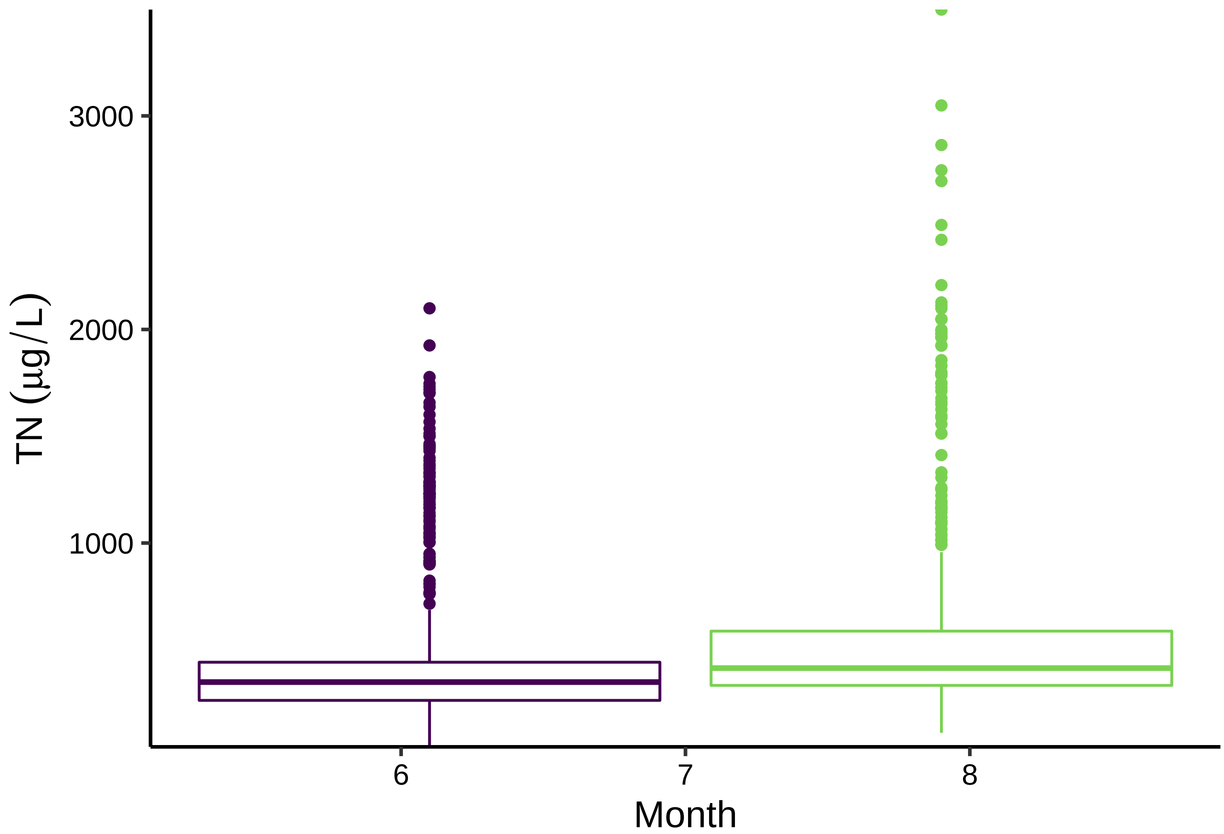
```r
print(TP.Boxplot)
```

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




```r
# TN Boxplot

TN.Boxplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tn_ug, color = lakename)) +
  geom_boxplot() +
  labs(x = expression(paste("Month")),
       y = expression(paste("TN"~ (mu*g / L))), color = "Lake") +
   scale_y_continuous(expand = c(0,0)) +
  scale_color_viridis(discrete = TRUE, option = "viridis", end = 0.8) +
  theme(legend.position = "none")
print(TN.Boxplot)
```

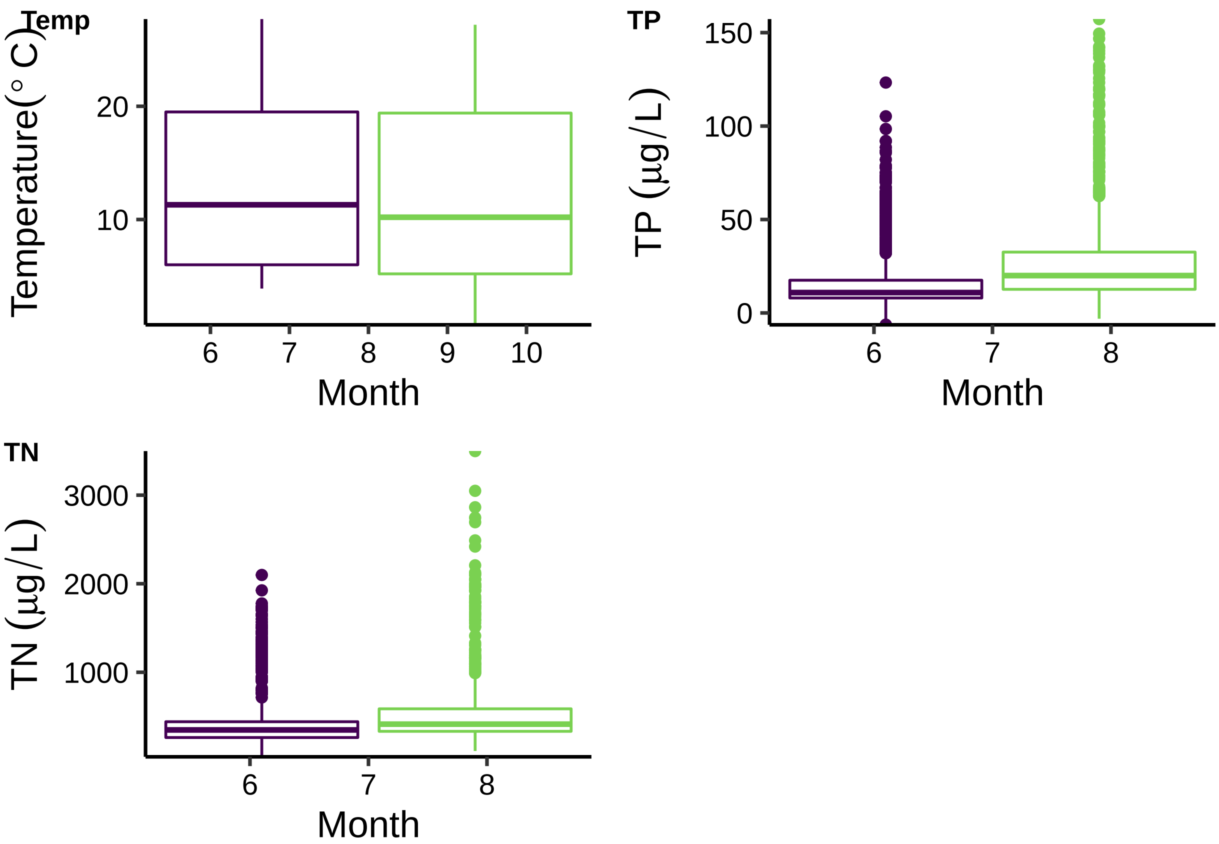
```
## Warning: Removed 21583 rows containing non-finite values (stat_boxplot).
```




```
# 3 Graphs Combined


three.plots <- plot_grid(Temp.Boxplot, TP.Boxplot, TN.Boxplot, labels = c('Temp', 'TP', 'TN'),
                         label_size = 10, align = "vh", nrow = 2)
```

```
## 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).
```

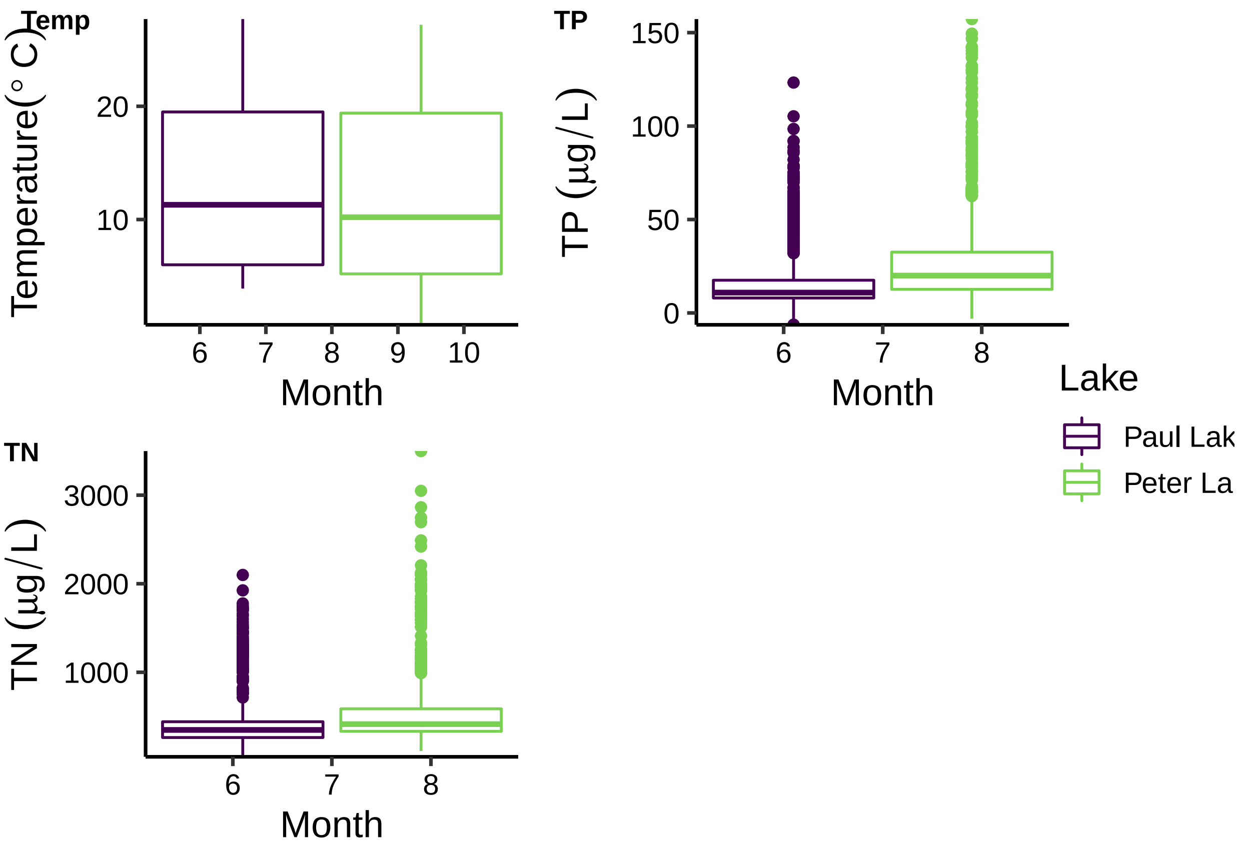
```
print(three.plots)
```




```
legend <- get_legend(Temp.Boxplot + theme(legend.position = "right"))
```

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

```
plot_grid(three.plots, legend, rel_widths = c(3, .4))
```

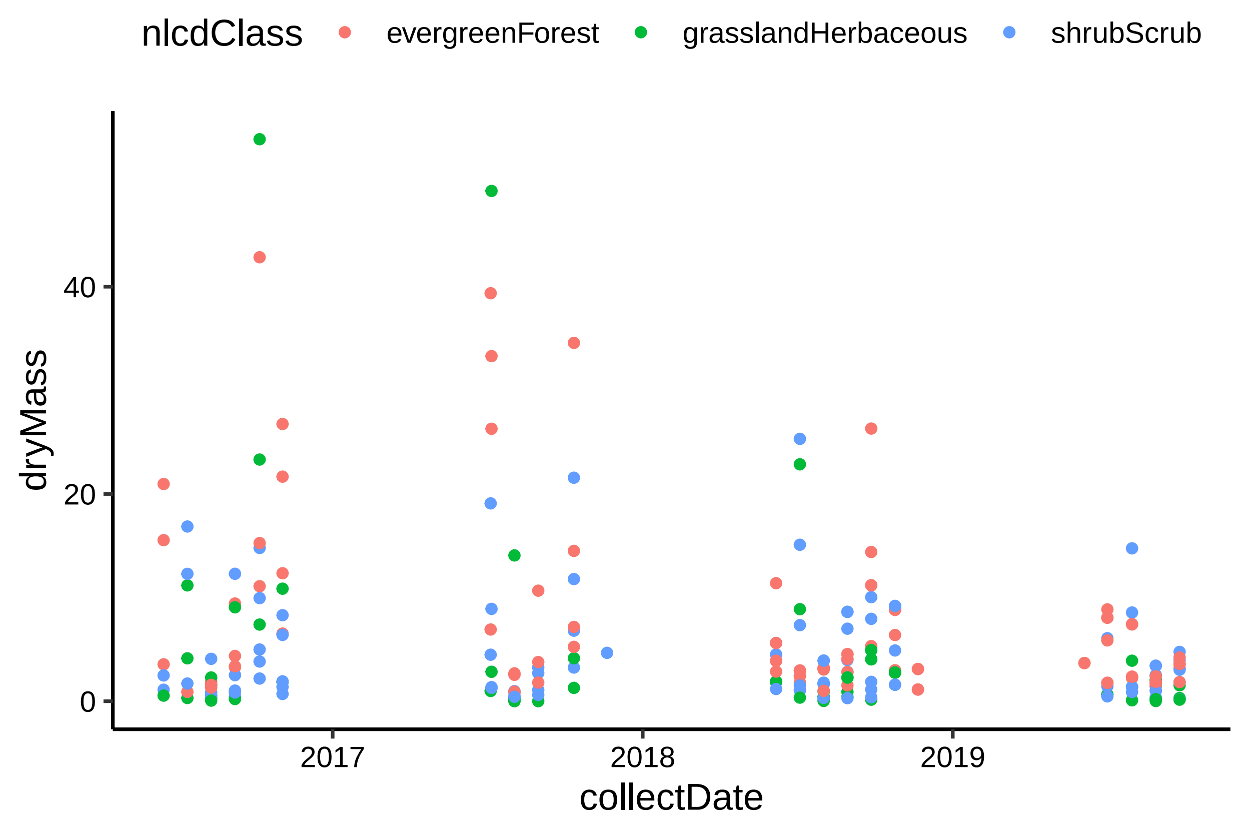


Question: What do you observe about the variables of interest over seasons and between lakes?

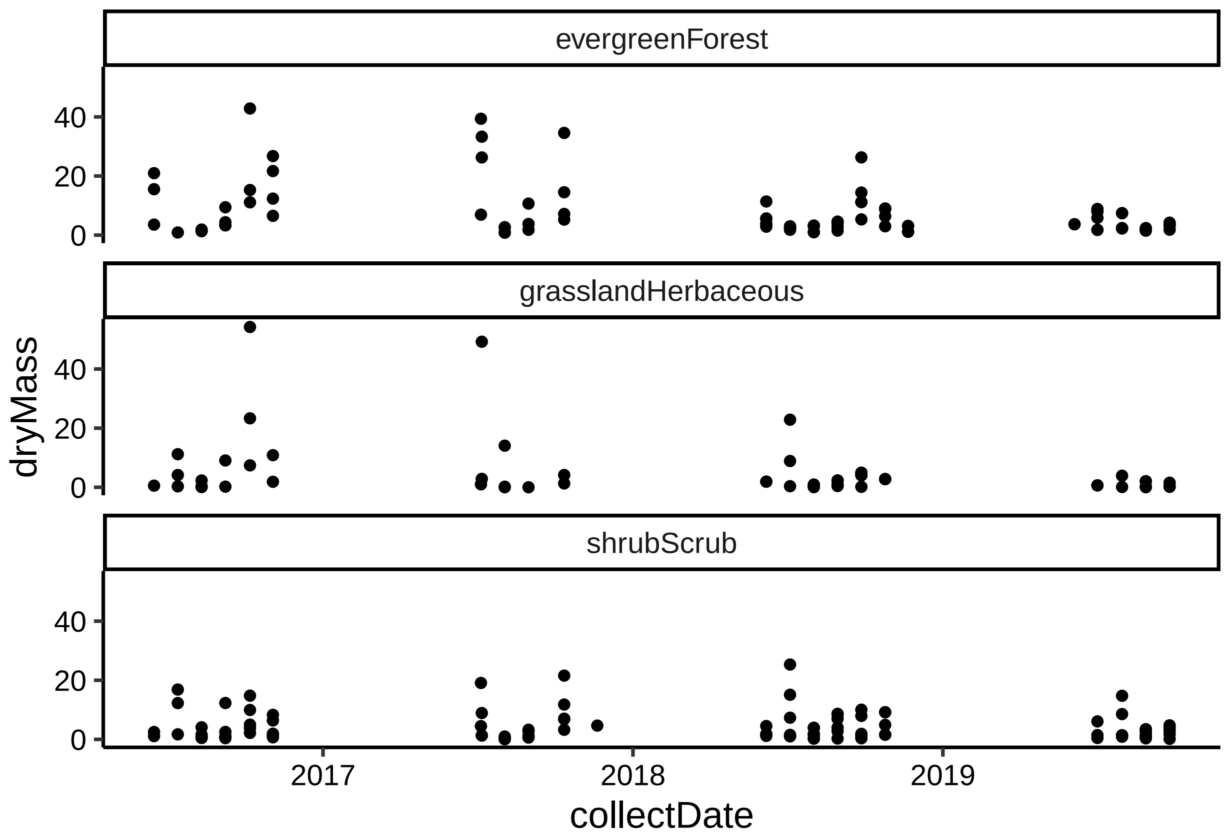
> Answer: In the summer, Paul Lake has slightly higher temperatures, but lower TP and TN concentrations.

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

NIWO.needles <-
  ggplot(subset(NIWO.Litter, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point()

 print(NIWO.needles)
```




```
# 7
NIWO.needles.facet <-
  ggplot(subset(NIWO.Litter, functionalGroup == "Needles"),
        aes(x = collectDate, y = dryMass)) +
 geom_point() +
  facet_wrap(vars(nlcdClass), nrow = 3)

 print(NIWO.needles.facet)
```



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

> Answer: Plot 7 is more effective because you are able to easily differentiate the NLCD classes whereas plot 6 displayed all the NLCD classes within each collection year so it was difficult to distinguish the dry mass and collection date of each class.