Assignment 5: Data Visualization

Victoria Thompson

Fall 2024

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Visualization

Directions

1. Rename this file `<FirstLast>_A05_DataVisualization.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 your code is tidy; use line breaks to ensure your code fits in the knitted output.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up your session

1. Set up your session. Load the tidyverse, lubridate, here & cowplot packages, and verify your home directory. Read in the NTL-LTER processed data files for nutrients and chemistry/physics for Peter and Paul Lakes (use the tidy `NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv` version in the Processed_KEY folder) and the processed data file for the Niwot Ridge litter dataset (use the `NEON_NIWO_Litter_mass_trap_Processed.csv` version, again from the Processed_KEY folder).

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

```r
#1
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
library(here)
```

```
## here() starts at /home/guest/EDE_Fall2024
```

```
library(cowplot)
```

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

```
here()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

```
NTL.PeterPaul.chem<- read.csv(
  file = here('Data','Processed_KEY',
              'NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv'),
  stringsAsFactors = T
)

NEON.litter<- read.csv(
  file = here('Data','Processed_KEY',
              'NEON_NIWO_Litter_mass_trap_Processed.csv'),
  stringsAsFactors = T
)


#2
#changing to date format
NTL.PeterPaul.chem$sampledate <- ymd(NTL.PeterPaul.chem$sampledate)
NEON.litter$collectDate <- ymd(NEON.litter$collectDate)

class (NTL.PeterPaul.chem$sampledate)
```

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

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

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

Define your theme

3. Build a theme and set it as your default theme. Customize the look of at least two of the following:

- Plot background

- Plot title
- Axis labels
- Axis ticks/gridlines
- Legend

```
#3
plottheme <- theme_gray(base_size = 14) +
  theme(
    #background color (fills in light blue)
    plot.background = element_rect(fill = "lightblue", color = NA),
    #plot title (face = bold/italic/underline, size= text size, color =
    # text color, hjust= text alignment )
    plot.title = element_text(face = "bold", size = 16, color = "black", hjust = 0.5),
    #axis text
    axis.text = element_text(color = "darkblue"),
    #legend at the bottom
    legend.position = "bottom"
  )

# set default
theme_set(plottheme)
```

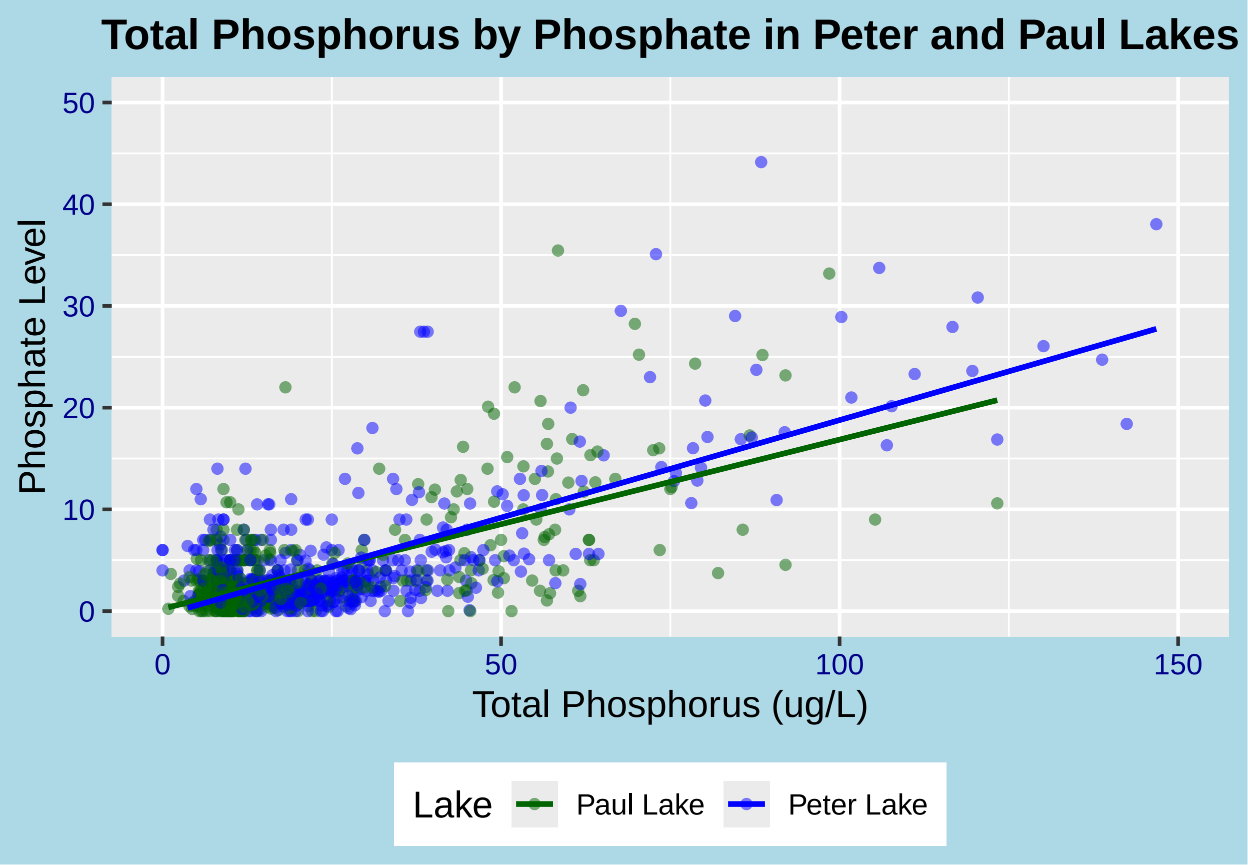
Create graphs

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 line(s) of best fit using the `lm` method. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```
#4
NTL.PeterPaul.chem %>%
  #set axes and color by lake
  ggplot(aes(x = tp_ug, y = po4, color = lakename)) +
  #make points transparent so easier to see
 geom_point(alpha = 0.5)+
  #adding title, axis labels, legend labels
 labs(
title = "Total Phosphorus by Phosphate in Peter and Paul Lakes",
y = "Phosphate Level",
x = "Total Phosphorus (ug/L)", color = "Lake") +
  # add lines of best fit
  geom_smooth(
method = lm,
se=FALSE) +
  # adding y limits to exclude outlier
  xlim(0, 150) +
  ylim(0,50) +
  # manually recoloring points and lines to match aesthetic
  scale_color_manual(
    values = c("Peter Lake" = "blue", "Paul Lake" = "dark green")
  )
```

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

## Warning: Removed 21948 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 21948 rows containing missing values or values outside the scale range
## (`geom_point()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_smooth()`).
```

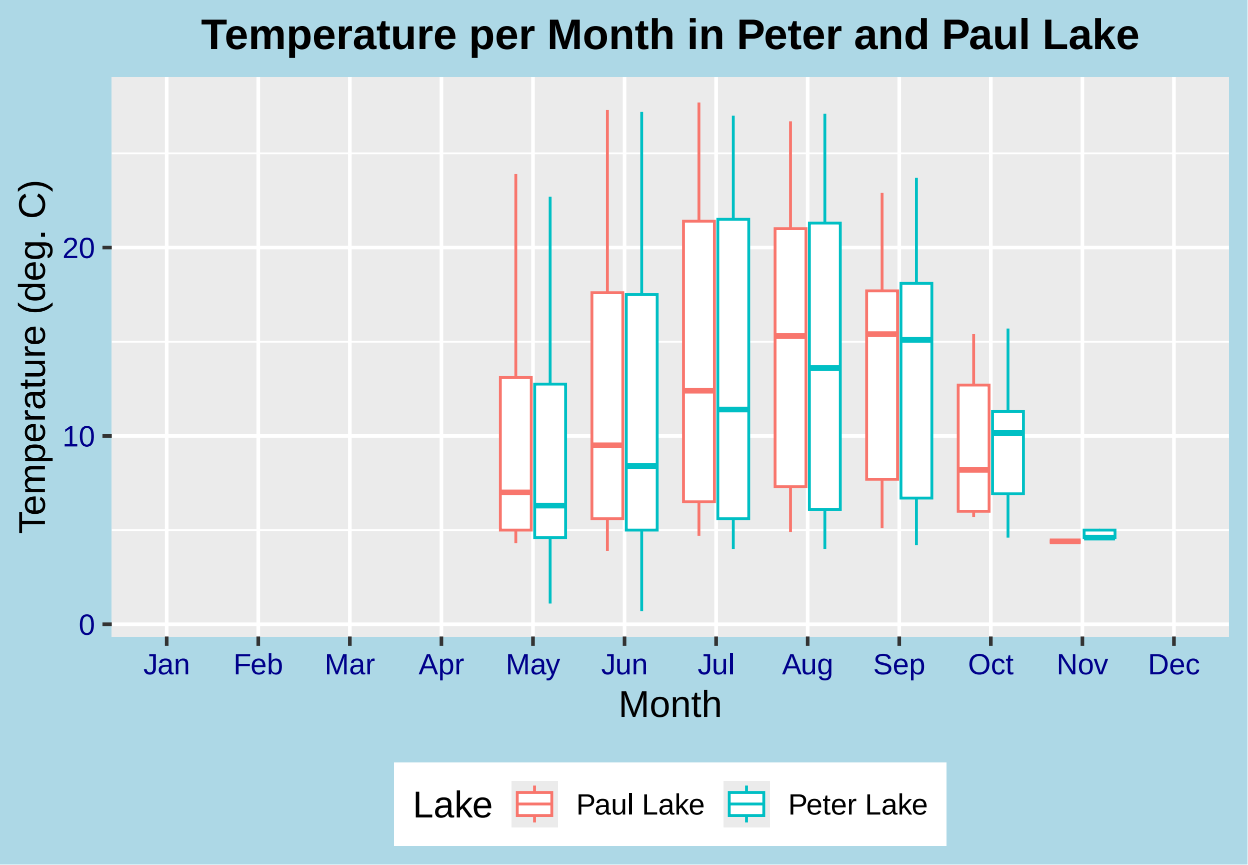


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

Tips: * Recall the discussion on factors in the lab section as it may be helpful here. * Setting an axis title in your theme to `element_blank()` removes the axis title (useful when multiple, aligned plots use the same axis values) * Setting a legend's position to "none" will remove the legend from a plot. * Individual plots can have different sizes when combined using `cowplot`.

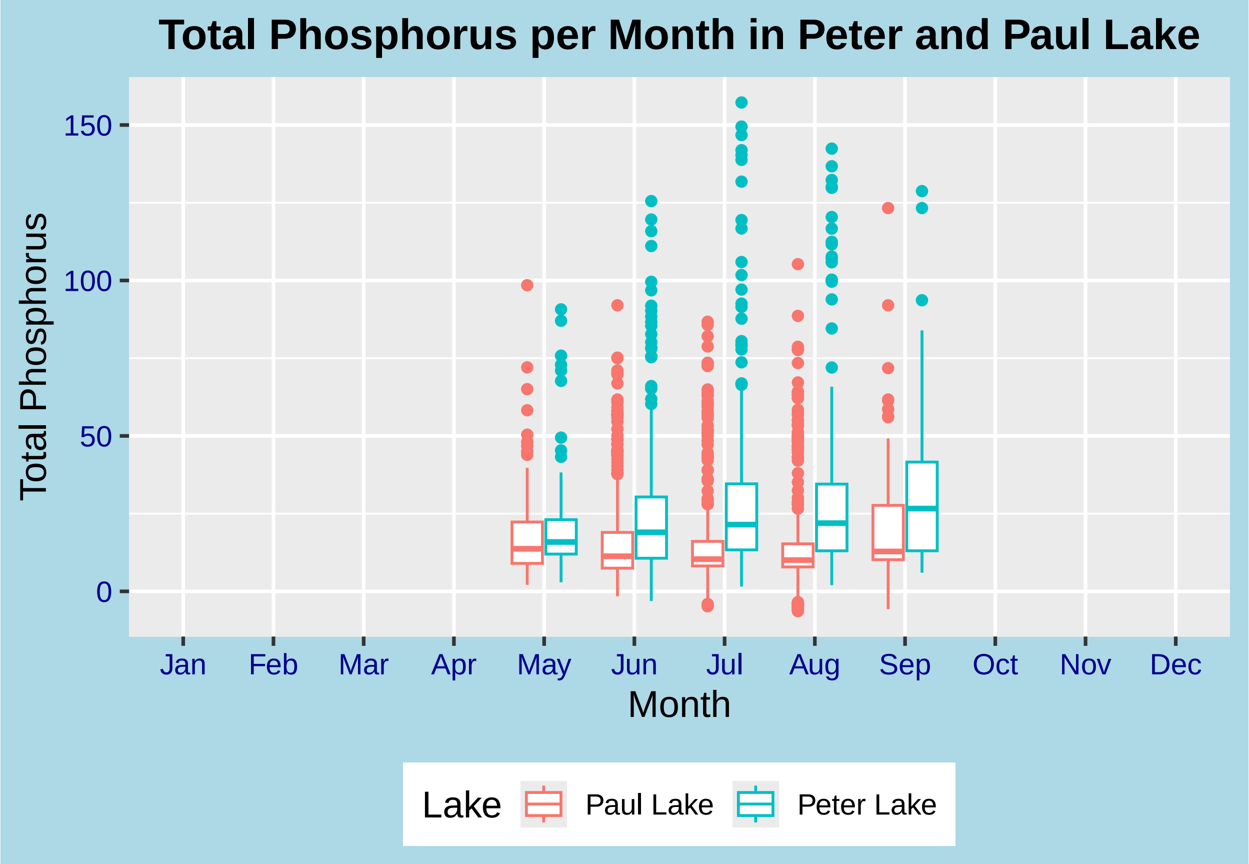
```
#5
#(a)
temp.plot <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=temperature_C,
color=factor(lakename)
))+
# drop x values to "false" so all months are included
scale_x_discrete(drop=F)+
geom_boxplot()+
labs(
x='Month', y = 'Temperature (deg. C)',
color='Lake', title = "Temperature per Month in Peter and Paul Lake" )

temp.plot
```

```
## Warning: Removed 3566 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



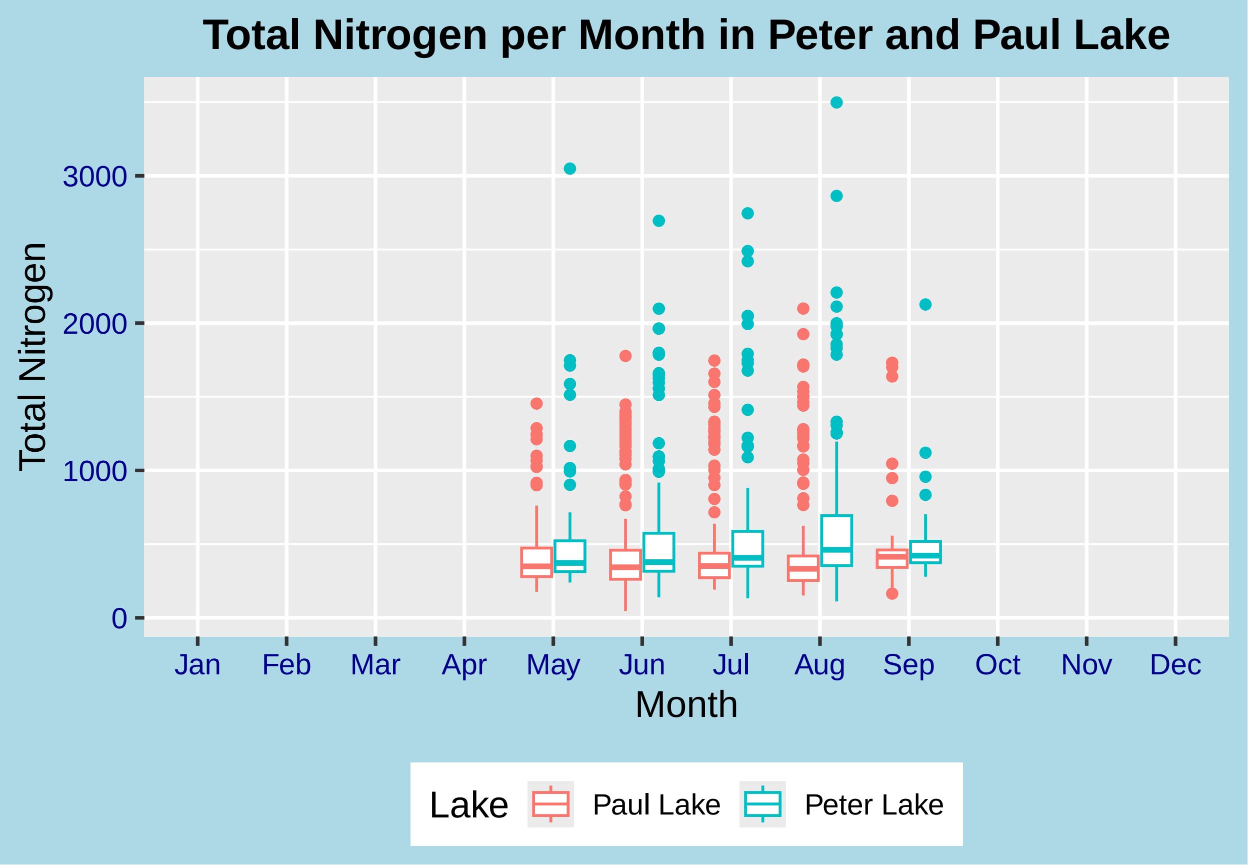
```
#(b)
P.plot <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=tp_ug,
color=factor(lakename)
))+
# drop x values to "false" so all months are included
scale_x_discrete(drop=F)+
geom_boxplot()+
labs(
x='Month', y = 'Total Phosphorus',
color='Lake', title = "Total Phosphorus per Month in Peter and Paul Lake" )

P.plot
```

```
## Warning: Removed 20729 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



```
#(c)
N.plot <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=tn_ug,
color=factor(lakename)
))+
# drop x values to "false" so all months are included
scale_x_discrete(drop=F)+
geom_boxplot()+
labs(
x='Month', y = 'Total Nitrogen',
color='Lake', title = "Total Nitrogen per Month in Peter and Paul Lake" )

N.plot
```

```
## Warning: Removed 21583 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



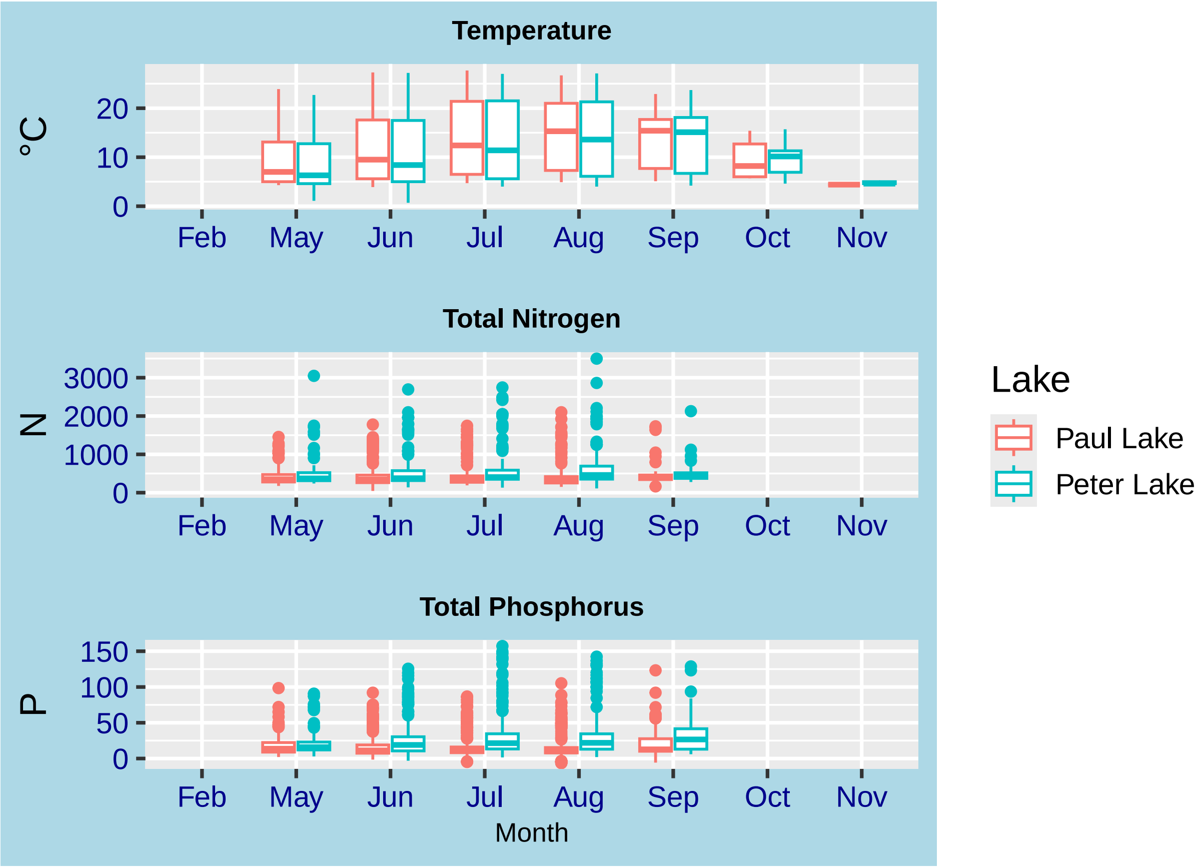
```
########

#create a cowplot that combines the three graphs.

#removing labels as needed, truncating titles for readbility, removing extra
#legends, removing extra months

#(a)
temp.plot2 <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=temperature_C,
color=factor(lakename)
))+
geom_boxplot()+
labs(
x=element_blank(), y = '°C',
color='Lake', title = "Temperature" ) +
  theme(legend.position = "none",
        #shrink titles so they fit better
        plot.title = element_text(size = 10))

#(b)
P.plot2 <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=tp_ug,
color=factor(lakename)
))+
geom_boxplot()+
labs(
x='Month', y = 'P',
color='Lake', title = "Total Phosphorus" )+
  theme(legend.position = "none",
        #shrink titles so they fit better
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 10))

#(c)
N.plot2 <- NTL.PeterPaul.chem %>%
ggplot(aes(
  #change months to factor to use names, not numbers
x=factor(month,levels = 1:12,labels = month.abb),
y=tn_ug,
color=factor(lakename)
))+
geom_boxplot()+
labs(
x=element_blank(), y = 'N',
color='Lake', title = "Total Nitrogen" )+
  theme(legend.position = "none",
        #shrink titles so they fit better
        plot.title = element_text(size = 10))

# I was having a hard time getting the legend without creating gaps between
#each plot. I used the cowplot function "get_legend" to apply one legend to
# all plots.
legend <- get_legend(temp.plot2 + theme(legend.position = "right"))
```

```
## Warning: Removed 3566 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```

```
## Warning in get_plot_component(plot, "guide-box"): Multiple components found;
## returning the first one. To return all, use ‘return_all = TRUE‘.
```

```
plot_grid(plot_grid(temp.plot2, N.plot2, P.plot2, nrow = 3, align = 'v'),
  legend,  # Add the legend on the right
  ncol = 2,  #  2 columns
  rel_widths = c(3, 1)
)
```

```
## Warning: Removed 3566 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).

## Warning: Removed 21583 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).

## Warning: Removed 20729 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



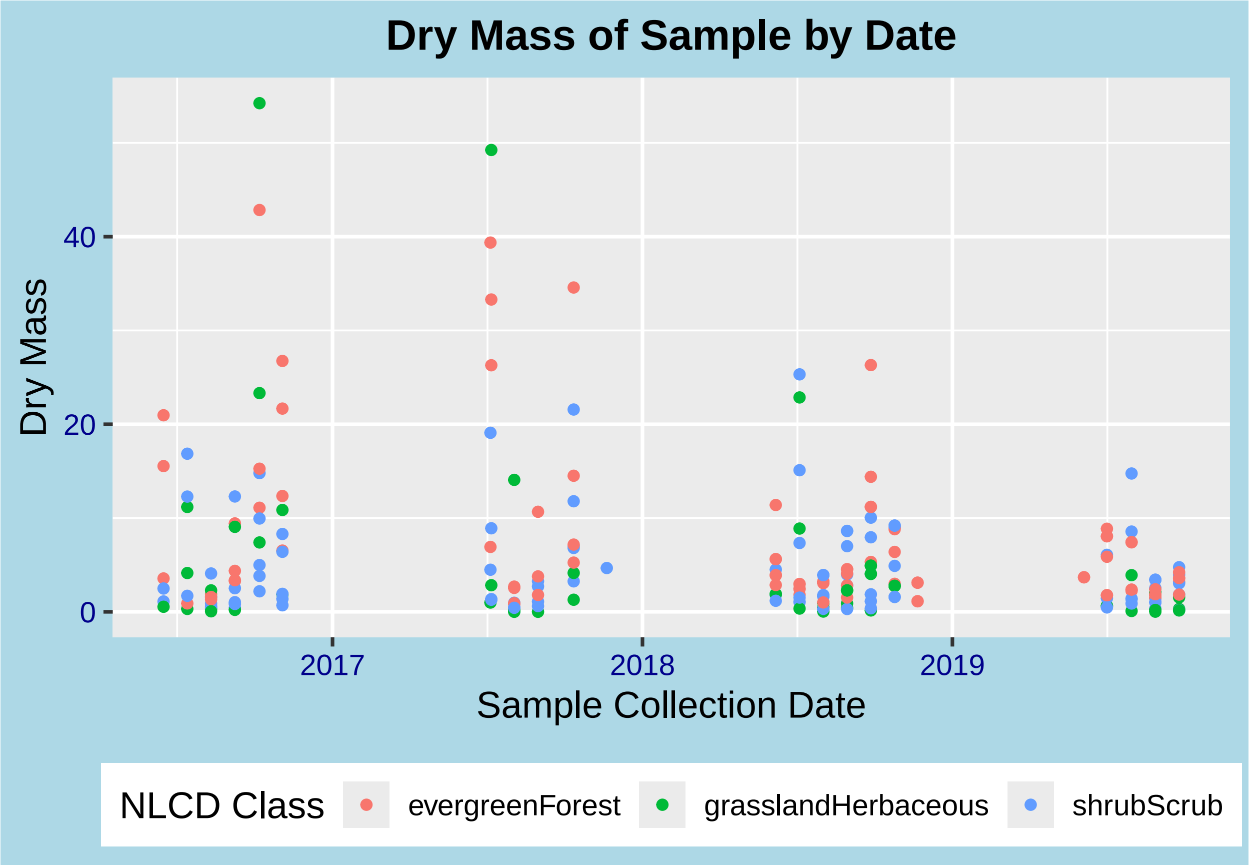
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: For temperatre, the summer months tend to have the highest values, with late spring/fall values generally being lower. That being said, the median value increases until Sept, but the upper quartile values remain highest in summer, indicating a wider range of high temps in the summer. Generally, the summer months have a greater range of temperatures, while the colder months have a smaller range between their upper and lower quartile values. The temps of Peter and Paul lakes are generlaly very similar. For nitrogen levels, there is not a clear variation by season, but Peter lake tends to have higher values across the entire year. It also has a higher

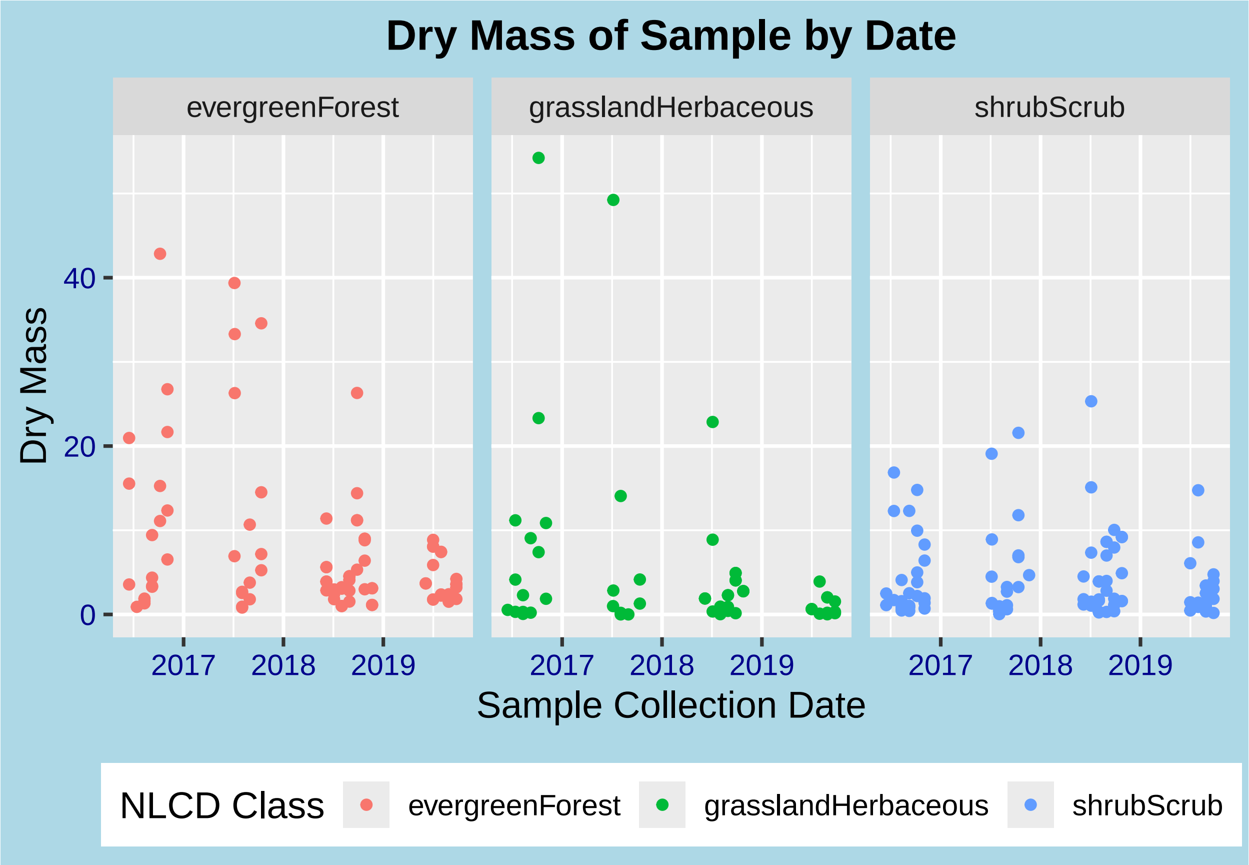
range of outlying values than Paul Lake. Finally, for phosphorus in Peter Lake, the amount of P increases over the course of the year. The opposite occurs in Paul Lake, where P decreases from the beginning of year until Fall. Like nitrogen, Peter Lake has a wider range of high values, and has generlaly higher values across the year than Paul 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.

```r
#6

litter.plot<-NEON.litter %>%
filter(functionalGroup == 'Needles') %>%
ggplot(
mapping = aes(
x=collectDate,
y=dryMass,
color=nlcdClass)
) +
geom_point()+
labs(
x= 'Sample Collection Date', y = 'Dry Mass',
color='NLCD Class', title = "Dry Mass of Sample by Date" )

litter.plot
```



```
#7
litter.plot.facet<-NEON.litter %>%
filter(functionalGroup == 'Needles') %>%
ggplot(
mapping = aes(
x=collectDate,
y=dryMass, color = nlcdClass)
) +
geom_point()+
  facet_wrap(
    facets = vars(nlcdClass),
    nrow=1, ncol=3
  ) +
labs(
x= 'Sample Collection Date', y = 'Dry Mass',
color='NLCD Class', title = "Dry Mass of Sample by Date" )

litter.plot.facet
```



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 makes the trends of each type of NLCD much more apparent. Plot 6 is a little overwhelming with data: because all of the points are plotted on each other, it is much harder to distinguish the trends of the individual NLCD. Plot 7 appears much cleaner and is much easier to draw conlcusions from, making it the more effective plot.