Assignment 5: Data Visualization

Tasneem Ahsanullah

Spring 2023

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. 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_PeterPaul_Processed.csv` version) and the processed data file for the Niwot Ridge litter dataset (use the `NEON_NIWO_Litter_mass_trap_Processed.csv` version).

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

```
#1
#importing tidyverse, lubridate, here & cowplot
library(tidyverse); library(lubridate); library(here);
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble  3.1.8     v dplyr   1.1.0
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 1.0.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
##
```

```
## Attaching package: 'lubridate'
##
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
##
##
## here() starts at /Users/tasneemahsanullah/Desktop/Classes/EDA/DataAnalytics
```

```
library(cowplot)
```

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

```
#Assigned a variable to the processed data folder location
processed_data = "Data/Processed"

#reading in the data
NTL.chem.PeterPaul <- read.csv(
  here(processed_data,
  "NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"),
  stringsAsFactors = TRUE)

NTL.chem.phys.PeterPaul <- read.csv(
  here(processed_data,
  "NTL-LTER_Lake_ChemistryPhysics_PeterPaul_Processed.csv"),
  stringsAsFactors = TRUE)

NTL.nutrients.PeterPaul <- read.csv(
  here(processed_data,
  "NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv"),
  stringsAsFactors = TRUE)

NEON.litter <- read.csv(
  here(processed_data,
  "NEON_NIWO_Litter_mass_trap_Processed.csv"),
  stringsAsFactors = TRUE)

#2
#checking if dates are in date format
class(NTL.chem.PeterPaul$sampledate)
```

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

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

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

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

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

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

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

```
#changing format to date
NTL.chem.PeterPaul$sampledate <- ymd(NTL.chem.PeterPaul$sampledate)

NTL.chem.phys.PeterPaul$sampledate <- ymd(NTL.chem.phys.PeterPaul$sampledate)

NTL.nutrients.PeterPaul$sampledate <- ymd(NTL.nutrients.PeterPaul$sampledate)

NEON.litter$collectDate <- ymd(NEON.litter$collectDate)
```

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

```
#custom theme with adjusted plot background, title and legend background/title
lake.theme <- theme_classic() +
  theme(
    plot.background = element_rect(
      color='black',
      fill='seashell'
    ),
    plot.title = element_text(
      colour = 'black',
      size = '14'
    ),
    legend.background = element_rect(
      color='black',
      fill ='white'
    ),
    legend.title = element_text(
      color='black'
    )

  )

#setting the theme for all plots
theme_set(lake.theme)
```

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()`).

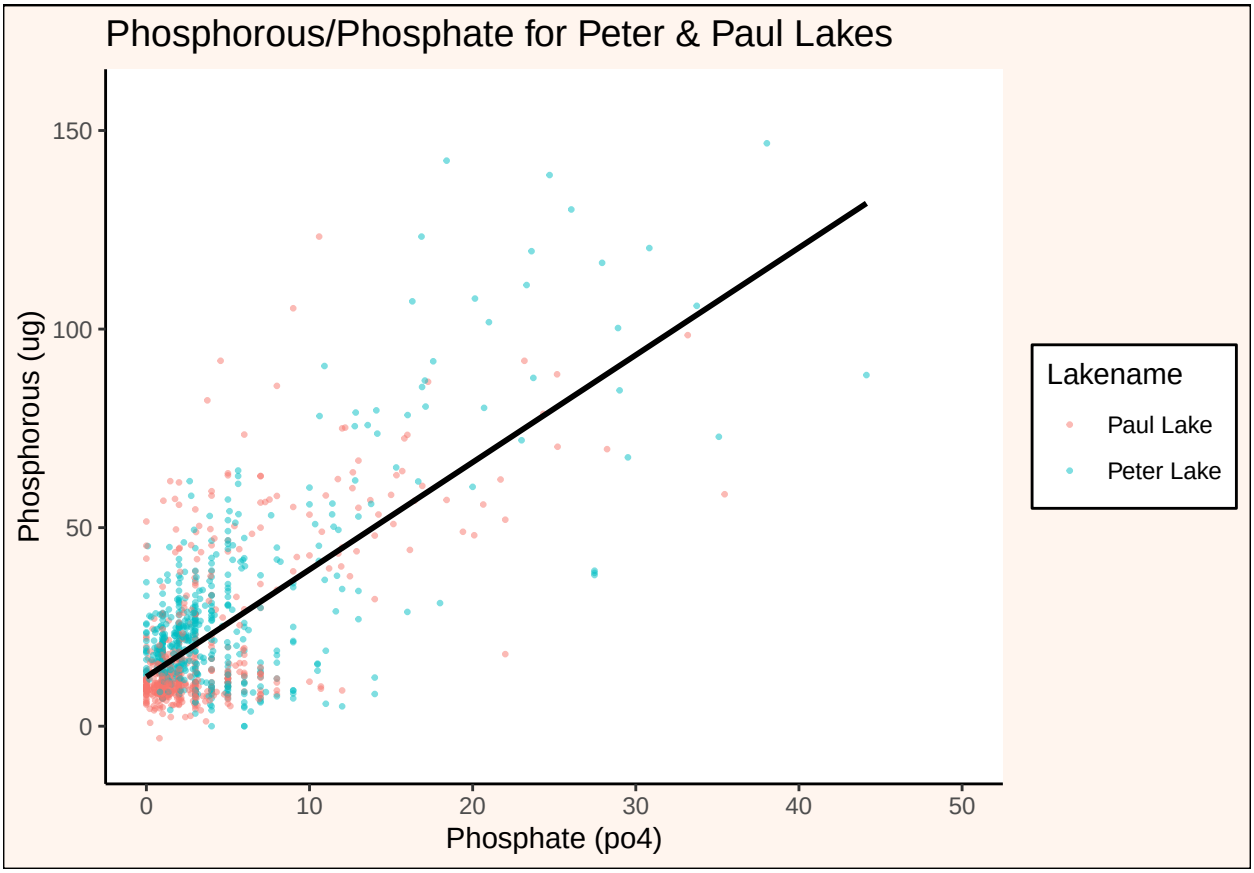
```
#4
#phosphorous by phosphate plot differentiated by lake
#with a line of best fit
phos.plot <- NTL.chem.PeterPaul %>%
  ggplot(aes(
      x=po4,
      y=tp_ug,
      color=lakename)
    ) +
  geom_point(size=0.5,alpha=0.5) +
    labs(
    title="Phosphorous/Phosphate for Peter & Paul Lakes",
    x="Phosphate (po4)",
    y="Phosphorous (ug)",
    color="Lakename"
    ) +
  geom_smooth(
    method='lm',
    se=FALSE,
    color='Black') +
  xlim(0,50)

print(phos.plot)
```

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

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

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

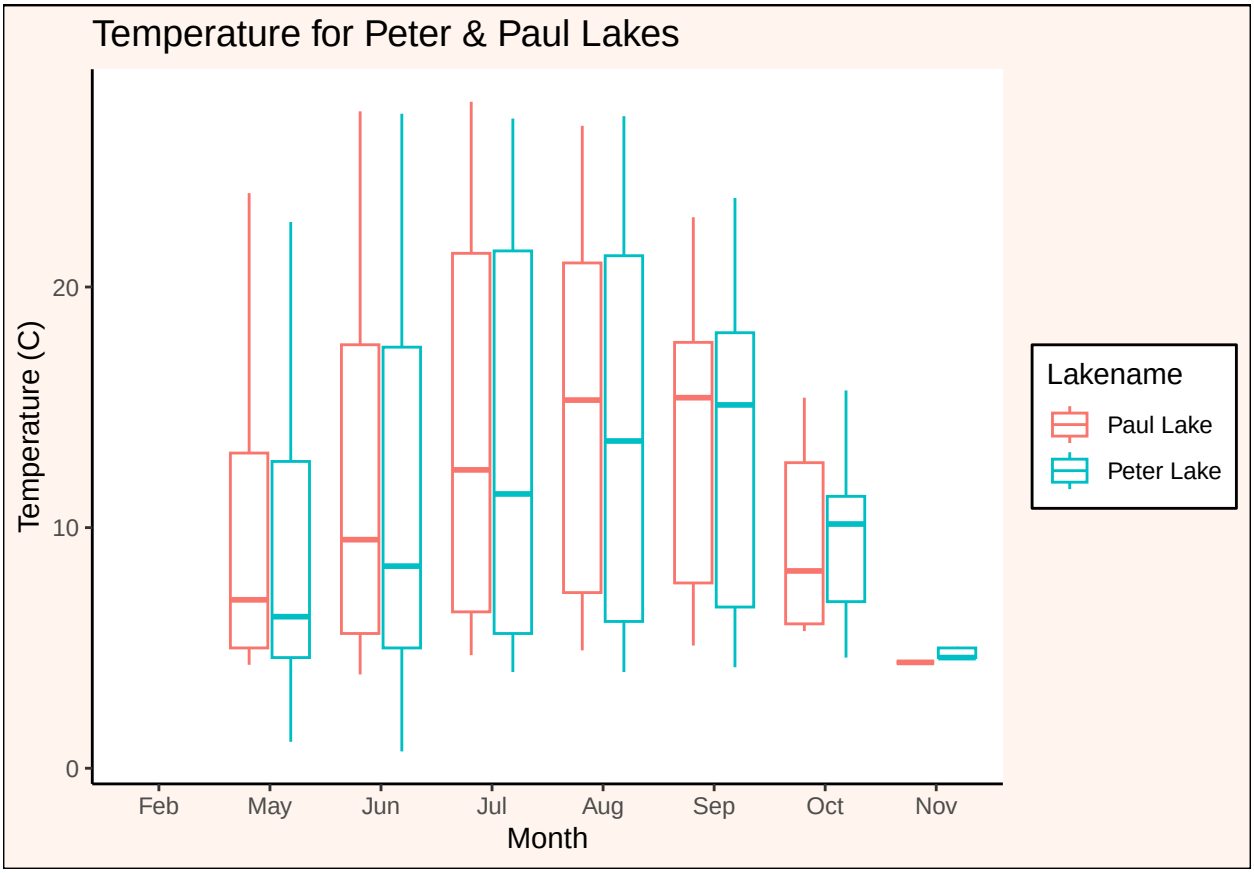


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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

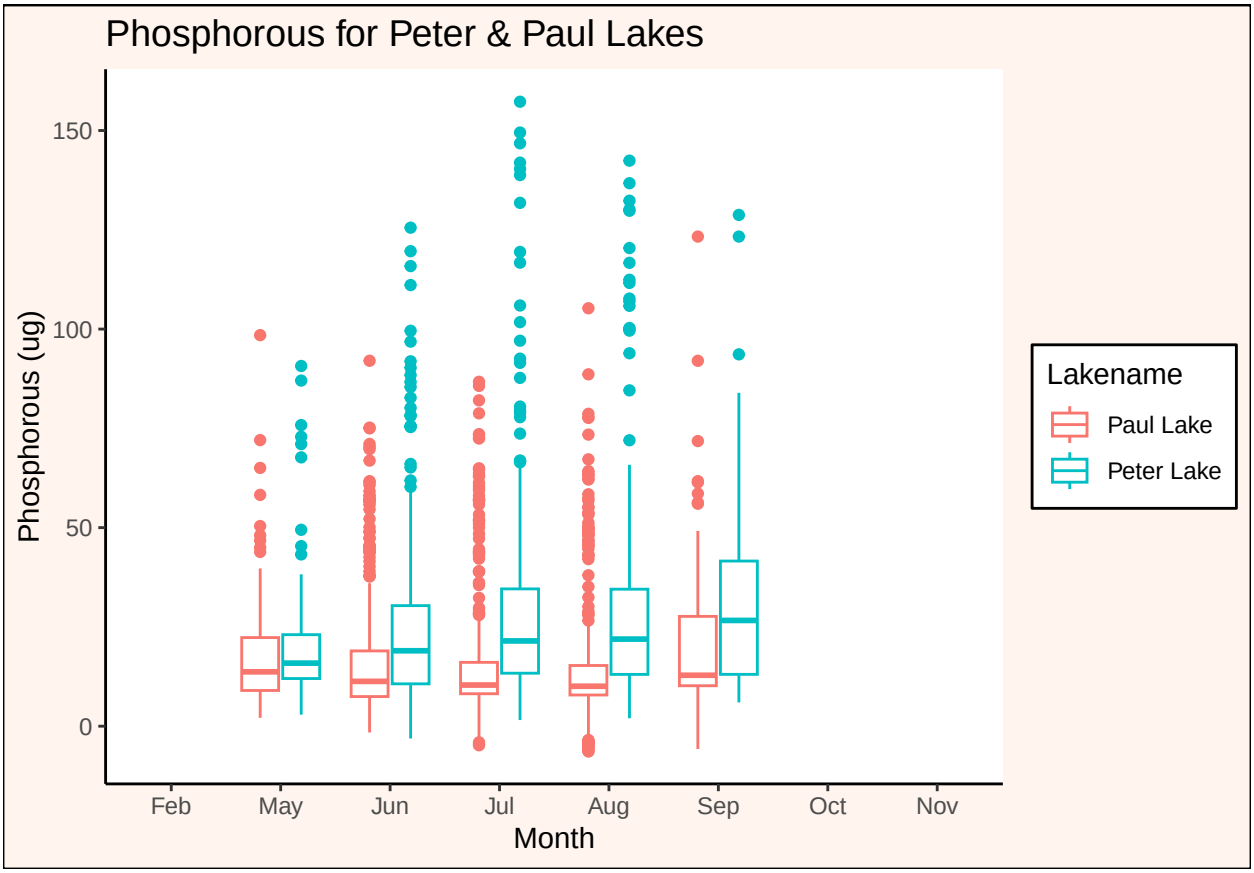
```
#5
#boxplot for temperature by month with lake name as the color aesthetic
temp.plot <-
  ggplot(NTL.chem.PeterPaul,
    aes(x=factor(month,levels=1:12,labels=month.abb),
    y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
labs(
  title="Temperature for Peter & Paul Lakes",
    x="Month",
    y="Temperature (C)",
    color="Lakename"
  )
print(temp.plot)
```

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



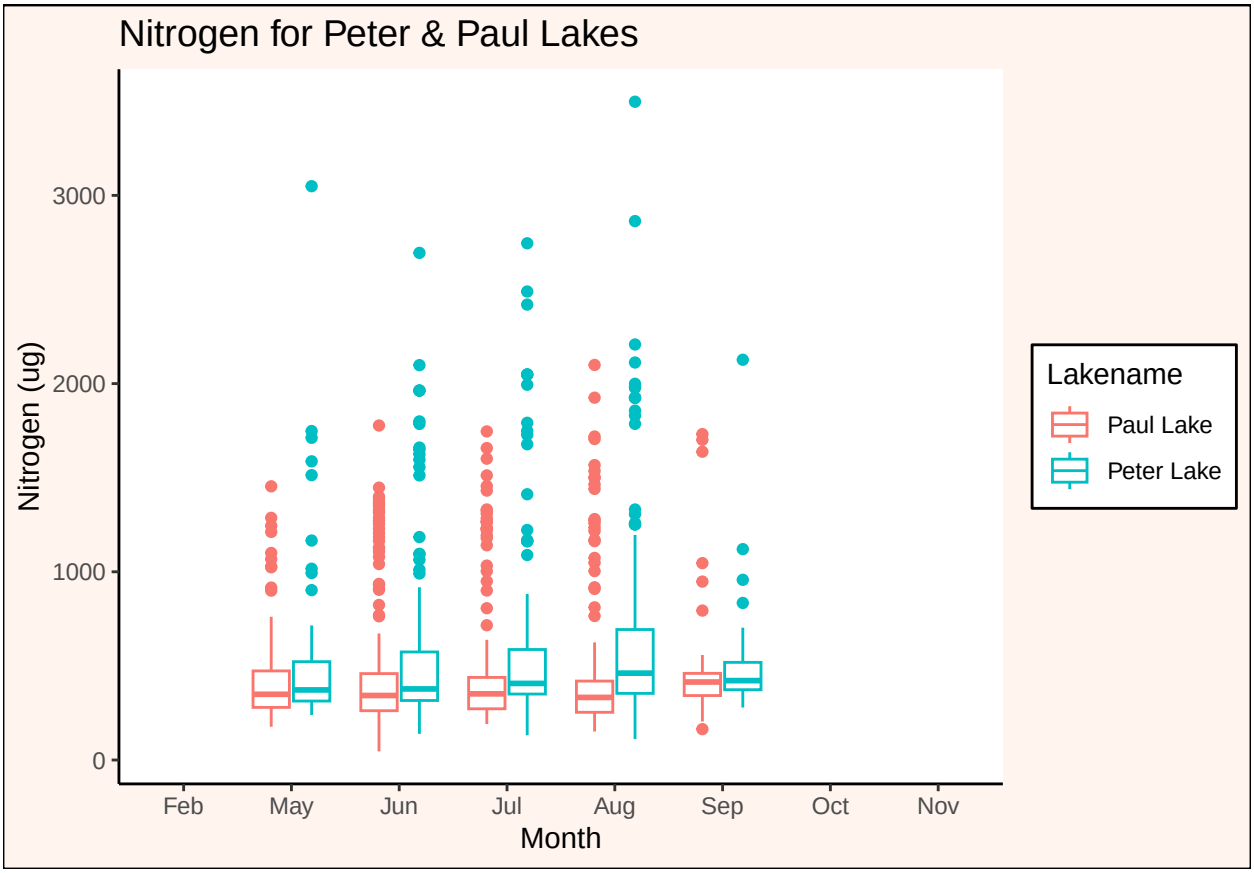

```
#boxplot for phosphorous by month with lake name as the color aesthetic
tp.plot <-
  ggplot(NTL.chem.PeterPaul,
  aes(x=factor(month,levels=1:12,labels=month.abb),
  y = tp_ug)) +
geom_boxplot(aes(color = lakename)) +
labs(
  title="Phosphorous for Peter & Paul Lakes",
    x="Month",
    y="Phosphorous (ug)",
    color="Lakename"
  )
print(tp.plot)
```

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




```
#boxplot for nitrogen by month with lake name as the color aesthetic
tn.plot <-
  ggplot(NTL.chem.PeterPaul,
  aes(x=factor(month,levels=1:12,labels=month.abb),
  y = tn_ug)) +
geom_boxplot(aes(color = lakename)) +
labs(
  title="Nitrogen for Peter & Paul Lakes",
    x="Month",
    y="Nitrogen (ug)",
    color="Lakename"
  )
print(tn.plot)
```

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

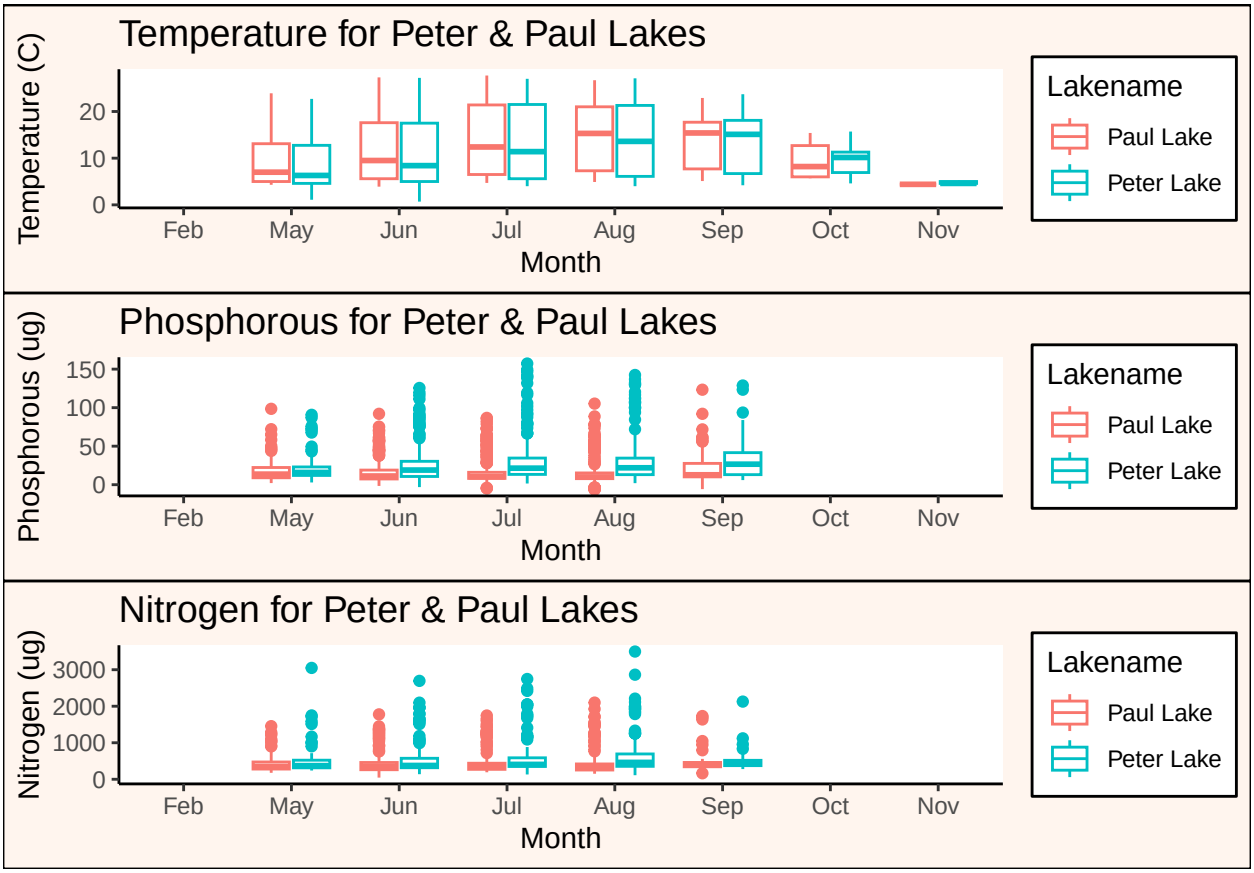



```
plot_grid(temp.plot, tp.plot, tn.plot, ncol= 1,
          align='v',axis='b')
```

```
## 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()‘).
```



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

Answer: The variables of interest increase in the spring, peak in summer and then decrease in winter.

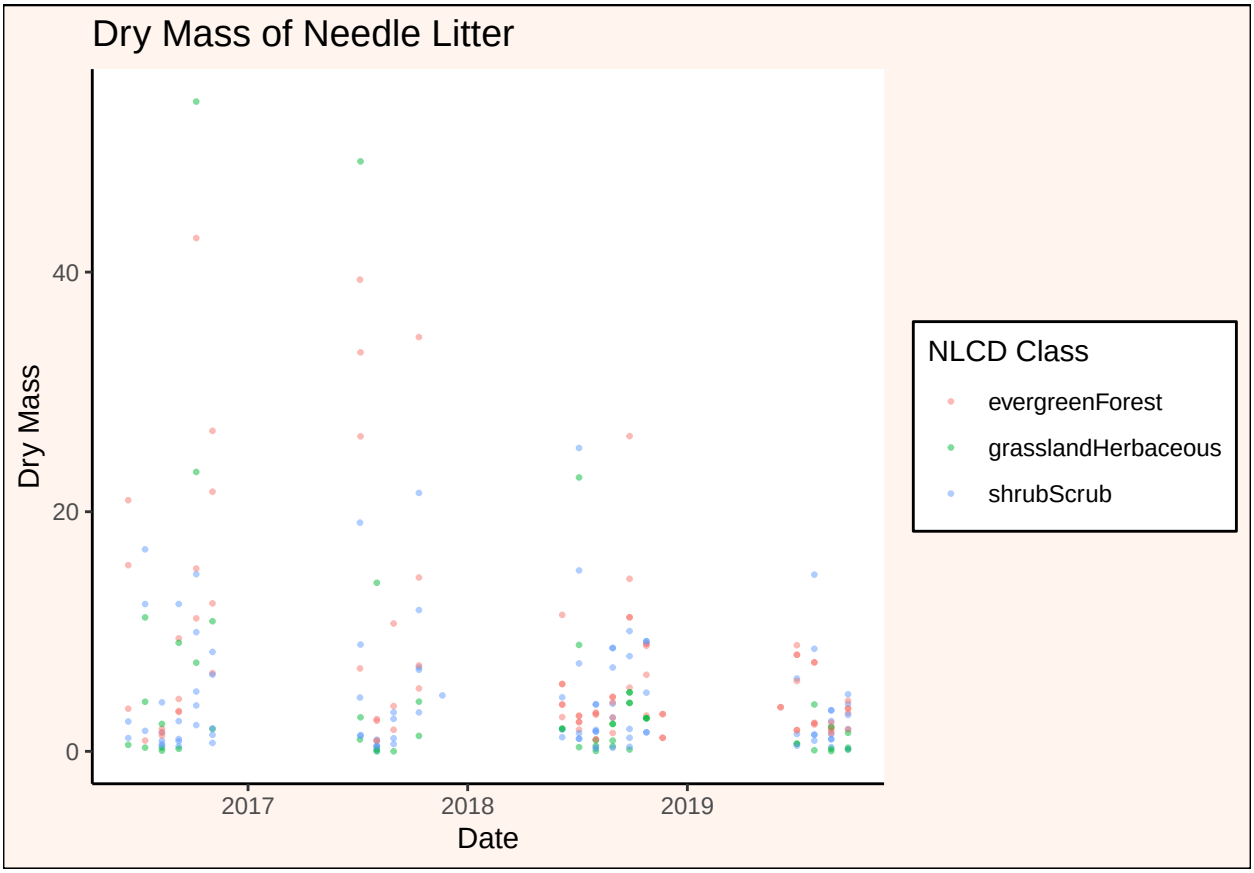
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

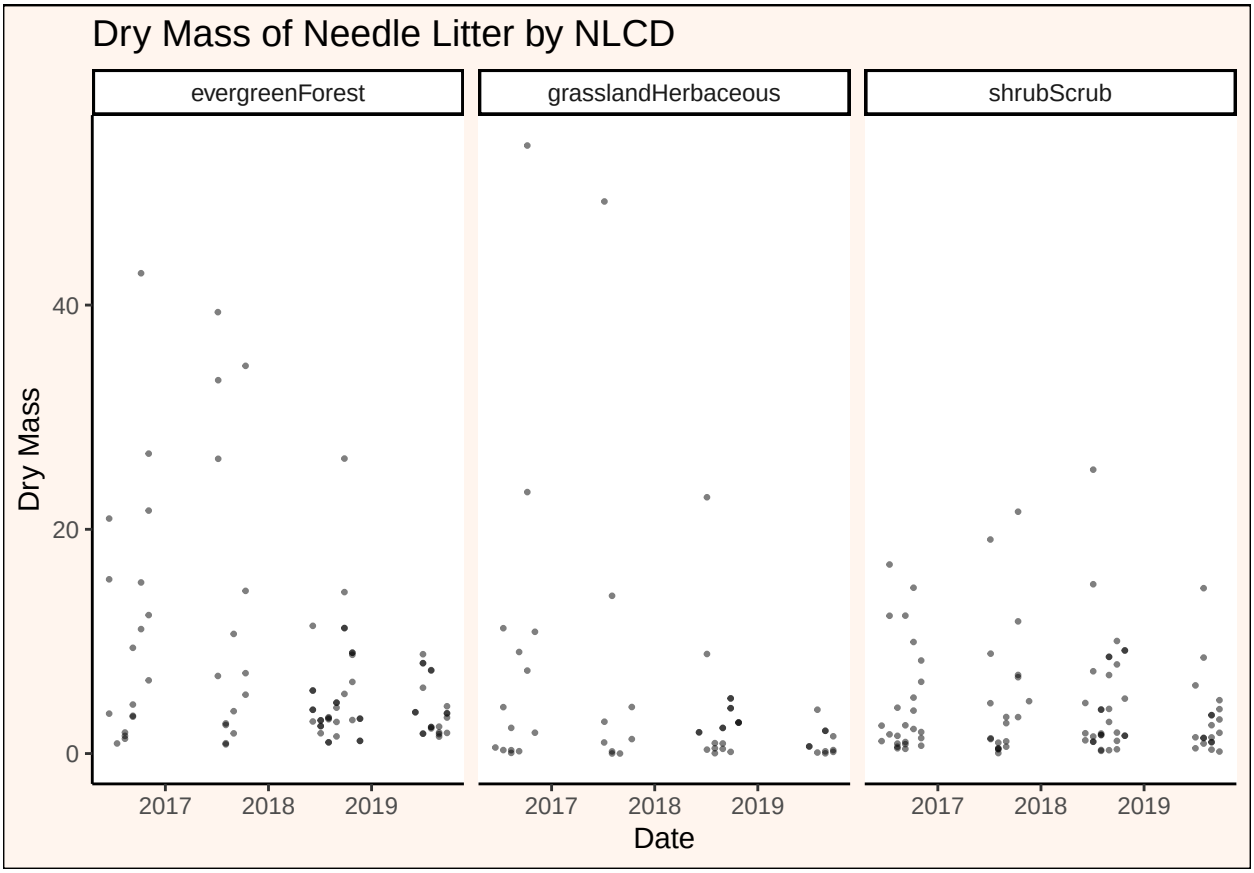
```
#6
#plot of the Needle dry mass by date
litter.plot <- NEON.litter %>%
  filter(functionalGroup=='Needles') %>%
  ggplot(aes(
      x=collectDate,
      y=dryMass,
      color=nlcdClass)
    ) +
  geom_point(size=0.5,alpha=0.5) +
    labs(
    title="Dry Mass of Needle Litter",
    x="Date",
    y="Dry Mass",
```

```
    color="NLCD Class"
    )

print(litter.plot)
```




```
#7
#plot of the Needle dry mass by date for each NLCD class
litter.facet <- NEON.litter %>%
  filter(functionalGroup=='Needles') %>%
  ggplot(aes(
      x=collectDate,
      y=dryMass,
      )
    ) +
  geom_point(size=0.5,alpha=0.5) +
    facet_wrap(vars(nlcdClass)) +
    labs(
    title="Dry Mass of Needle Litter by NLCD",
    x="Date",
    y="Dry Mass"
    )
print(litter.facet)
```



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

Answer: Plot 6 is more effective because it is easier to compare the different NLCD to each other since they are on the same graph and just different colors.