Assignment 5: Data Visualization

Jasmine Papas

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
#load packages and files
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.1      v purrr   1.0.0
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
## Loading required package: timechange
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(here)
```

```
## here() starts at /home/guest/R/EDA-Spring2023
```

```
library(cowplot)
```

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

```
getwd()
```

```
## [1] "/home/guest/R/EDA-Spring2023"
```

```
pp_chemistry_nutrients<- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processe
pp_chemistry_physics<- read.csv("./Data/Processed/NTL-LTER_Lake_ChemistryPhysics_PeterPaul_Processed.csv
niwot_litter_mass<- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",stringsAsFacto
#2
#checkto see if it is reading as Date
class(pp_chemistry_nutrients$sampledate)
```

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

```
class(pp_chemistry_physics$sampledate)
```

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

```
class(niwot_litter_mass$collectDate)
```

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

```
#rewrite so it reads as date
pp_chemistry_nutrients$sampledate<- ymd(pp_chemistry_nutrients$sampledate)
pp_chemistry_physics$sampledate<- ymd(pp_chemistry_physics$sampledate)
niwot_litter_mass$collectDate<-ymd(niwot_litter_mass$collectDate)

#double check dates
class(pp_chemistry_nutrients$sampledate)
```

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

```
class(pp_chemistry_physics$sampledate)
```

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

```
class(niwot_litter_mass$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
hw5_theme <- theme_classic(base_size = 14)+
  theme(axis.text = element_text(color = "black"),
        legend.position = "bottom")

#my theme for this hw assignment
```

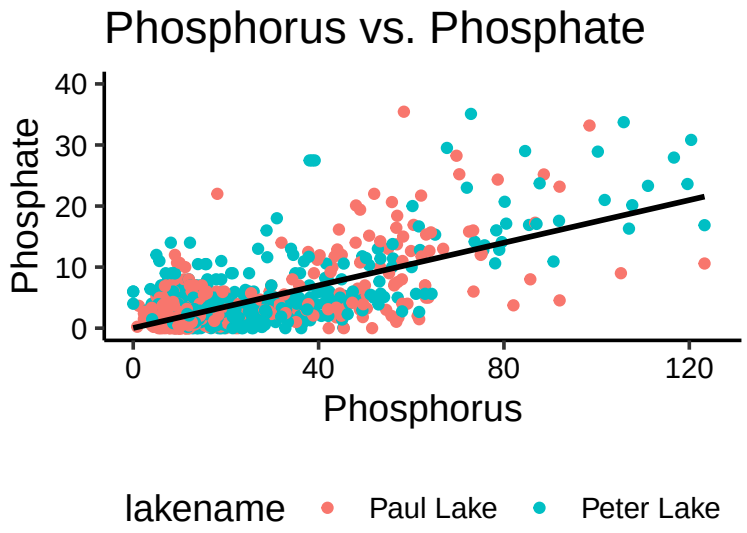
Create graphs

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
pp_phos_plot<-pp_chemistry_nutrients%>%
  ggplot(aes(x= tp_ug, y= po4,color = lakename))+
  geom_point()+
  geom_smooth(method = "lm", se= FALSE, color= "black")+
  hw5_theme+
  labs(x="Phosphorus", y= "Phosphate")+
  ggtitle("Phosphorus vs. Phosphate")+
  xlim(0,125)+
  ylim(0,40)

print(pp_phos_plot)
```

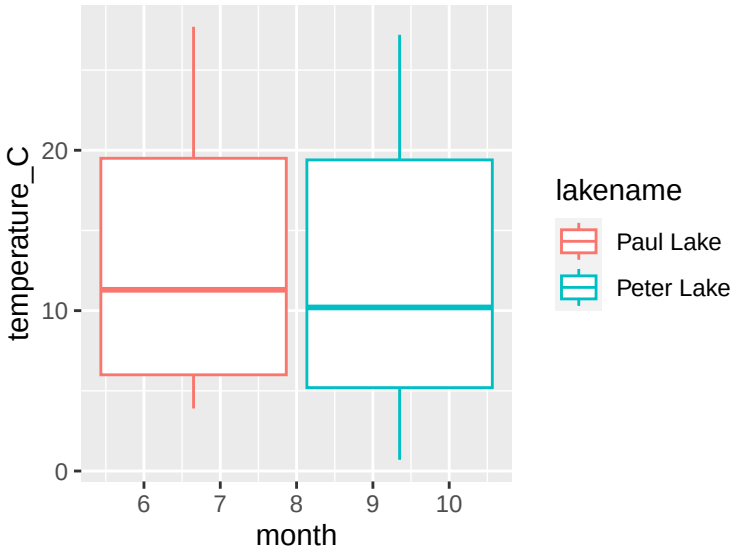
```
## `geom_smooth()` using formula = 'y ~ x'
```



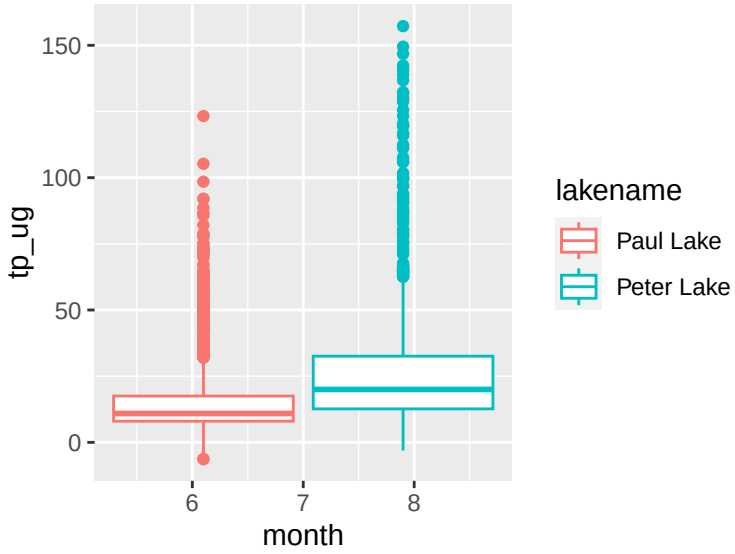
5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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

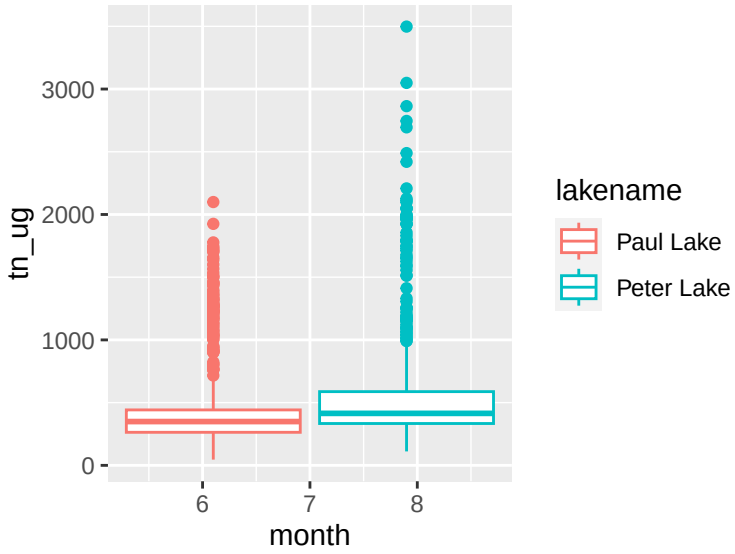
```
#5

#temperature
#class(pp_chemistry_nutrients$temperature_C)
temp<- ggplot(pp_chemistry_nutrients,aes(x= month, y= temperature_C))+
    geom_boxplot(aes(color= lakename))
print(temp)
```



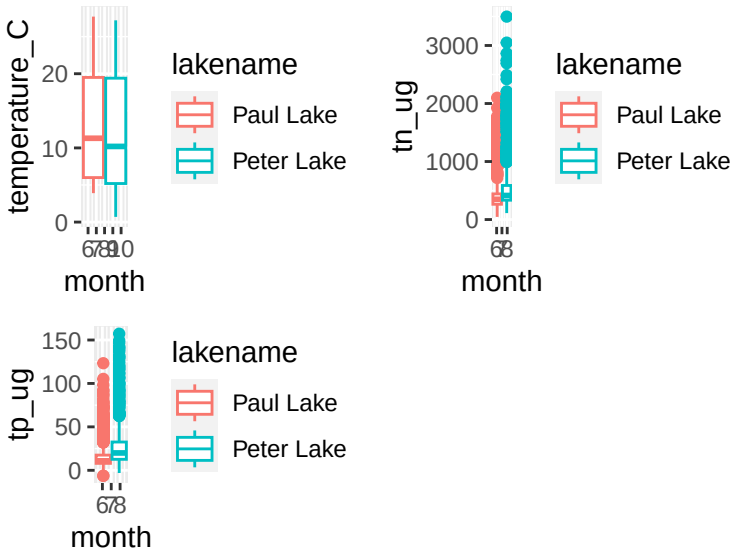
```
#TP
tp<-ggplot(pp_chemistry_nutrients,aes(x=month, y= tp_ug))+
  geom_boxplot(aes(color= lakename))
print(tp)
```




```
#TN
tn<-ggplot(pp_chemistry_nutrients,aes(x=month, y= tn_ug))+
  geom_boxplot(aes(color= lakename))
print(tn)
```




```
#All the plots together
plot_grid(temp, tn, tp, nrow = 2, align = 'h', rel_heights = c(1.25, 1))
```



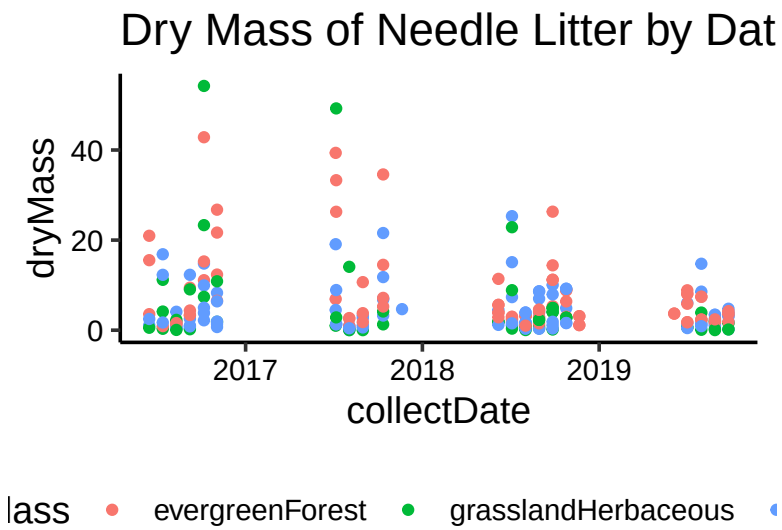
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Over the months, the average amounts of TN and TP are greater in Peter Lake than in Paul Lake. Meanwhile, the average temperature is slightly higher in Paul Lake than Peter Lake.

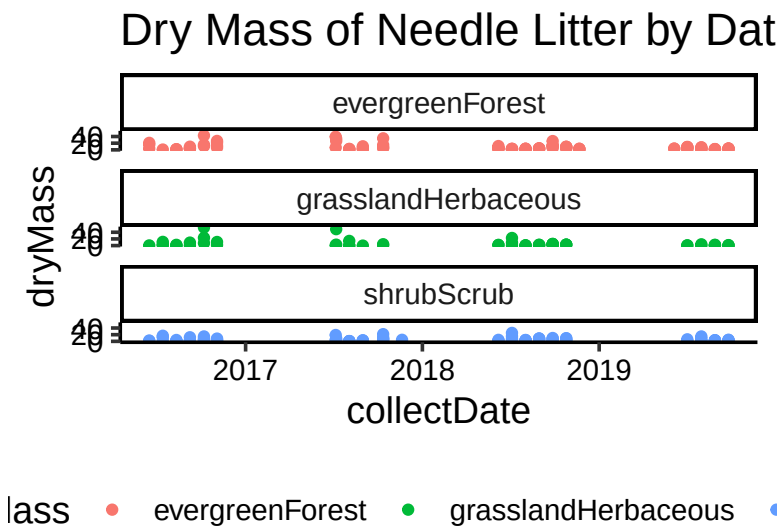
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
#6
needles<- niwot_litter_mass%>%
  filter(functionalGroup== "Needles")%>%
  ggplot(aes(x=collectDate, y=dryMass, color=nlcdClass))+
    geom_point()+
    hw5_theme+
    ggtitle("Dry Mass of Needle Litter by Date")

print(needles)
```




```
#7
needles_faceted<- niwot_litter_mass%>%
  filter(functionalGroup== "Needles")%>%
   ggplot(aes(x=collectDate, y=dryMass, color=nlcdClass))+
    geom_point()+
    hw5_theme+
    ggtitle("Dry Mass of Needle Litter by Date per NLCD Class")+
    facet_wrap(vars(nlcdClass), nrow = 3)
print(needles_faceted)
```



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

Answer: 7 is the more effective way to compare the different NLCD Classes to each other because it has the progression of the specific class over time. Meanwhile, 6 has all of the data on one graph which can be hard to read with all of the classes mixed together.