Assignment 5: Data Visualization

Laura Brockington

Spring 2023

#ask about plots in #5..why arent they showing all months? is there a way to have only 1 x-axis? #can’t get to knit graphs correctly

## 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   
library(tidyverse) #loading in necessary packages

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.0 ✔ purrr 1.0.1  
## ✔ tibble 3.1.8 ✔ dplyr 1.1.0  
## ✔ tidyr 1.3.0 ✔ stringr 1.5.0  
## ✔ readr 2.1.3 ✔ forcats 1.0.0  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(lubridate)

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

library(here)

## here() starts at /Users/laura/Desktop/EDA/EDA

library(cowplot)

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

here() #verifying my home directory

## [1] "/Users/laura/Desktop/EDA/EDA"

PeterPaul.chem.nutrients <-   
 read.csv(here("Data/Processed\_KEY/NTL-LTER\_Lake\_Chemistry\_Nutrients\_PeterPaul\_Processed.csv"), stringsAsFactors = TRUE) #loading in datasets  
Niwot.litter.mass <- read.csv(here("Data/Processed\_KEY/NEON\_NIWO\_Litter\_mass\_trap\_Processed.csv"), stringsAsFactors = TRUE)  
  
#2   
class(PeterPaul.chem.nutrients$sampledate) #checking format of date columns

## [1] "factor"

class(Niwot.litter.mass$collectDate)

## [1] "factor"

PeterPaul.chem.nutrients$sampledate <- ymd(PeterPaul.chem.nutrients$sampledate) #changing date columns to read as dates, not factors  
Niwot.litter.mass$collectDate <- ymd(Niwot.litter.mass$collectDate)

## Define your theme

1. 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  
mytheme <- theme\_gray(base\_size = 11) + #creating my custom plot theme  
 theme(axis.text = element\_text(family = "mono", color = "black"),   
 axis.title = element\_text(family = "mono"),  
 axis.ticks = element\_line(color = "lightblue3", linewidth = 0.5),  
 plot.title = element\_text(family = "mono", face = "bold", color = "black"),  
 panel.background = element\_rect(fill = "white"),  
 panel.grid.major = element\_line(color = "lightblue3",   
 linewidth = 0.5, linetype = "solid"),  
 panel.grid.minor = element\_line(color = "lightblue2",   
 linewidth = 0.25, linetype = "dashed"),  
 plot.background = element\_rect(color = "lightblue1"),  
 legend.position = "right",  
 legend.key = element\_rect(color = "white"),  
 legend.text = element\_text(family = "mono"),  
 legend.title = element\_text(family = "mono"),  
 legend.title.align = 0.5)  
theme\_set(mytheme)

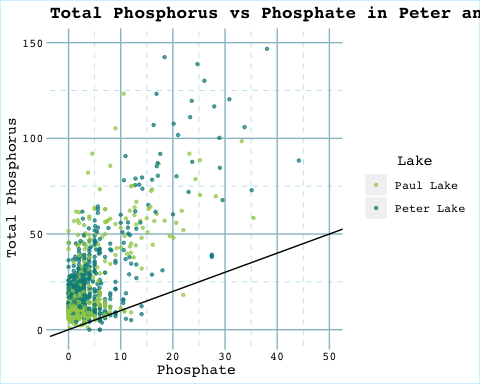
## Create graphs

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.

1. [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   
tp\_ug.po4 <- ggplot( #creating scatterplot of phosphorus by phosphate  
 PeterPaul.chem.nutrients,  
 aes(x=po4,  
 y=tp\_ug,  
 color=lakename)  
) +  
 geom\_point(size=0.85, alpha=0.7) +  
 xlim(-1, 50) +  
 ylim(-1, 150) +  
 geom\_abline(col='black') +  
 labs(  
 title = "Total Phosphorus vs Phosphate in Peter and Paul Lakes",  
 x = "Phosphate",  
 y = "Total Phosphorus",  
 col = "Lake") +  
 scale\_color\_manual(values=c("darkolivegreen3", "cyan4"))  
  
print(tp\_ug.po4)

## Warning: Removed 21948 rows containing missing values (`geom\_point()`).

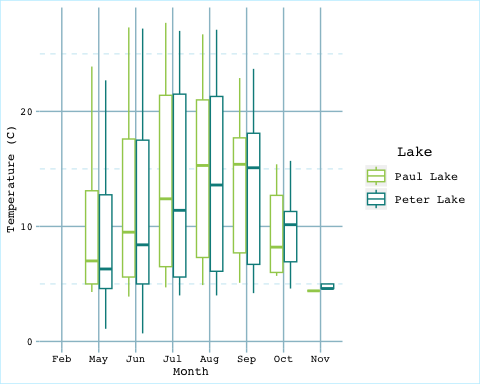


1. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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>

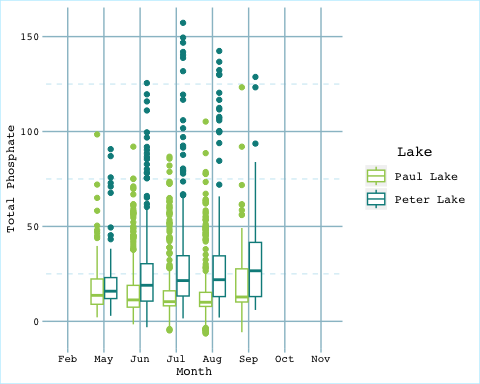
#5   
temp\_month <- #boxplot of temperature by month  
 ggplot(  
 PeterPaul.chem.nutrients,   
 aes(x = (factor(month, levels = 1:12, labels = month.abb)),   
 y = temperature\_C,  
 color = lakename)) +  
 geom\_boxplot() +  
 labs(x = "Month",  
 y = "Temperature (C)",  
 col = "Lake") +  
 scale\_color\_manual(values = c("darkolivegreen3", "cyan4")) +  
 theme(axis.text = element\_text(size = 8),  
 axis.title = element\_text(size = 9))  
print(temp\_month)

## Warning: Removed 3566 rows containing non-finite values (`stat\_boxplot()`).



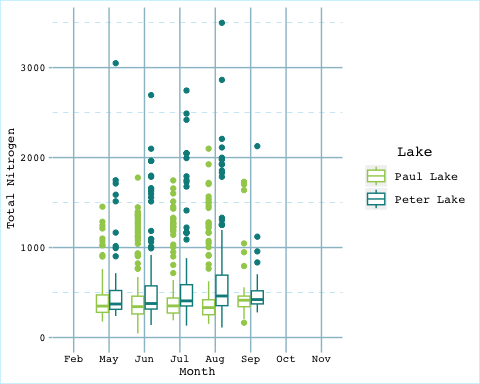
tp\_month <- #boxplot of total phosphorus by month  
 ggplot(  
 PeterPaul.chem.nutrients,   
 aes(x = (factor(month, levels = 1:12, labels = month.abb)),   
 y = tp\_ug,  
 color = lakename)) +  
 geom\_boxplot() +  
 labs(x = "Month",  
 y = "Total Phosphate",  
 col = "Lake") +  
 scale\_color\_manual(values = c("darkolivegreen3", "cyan4")) +  
 theme(axis.text = element\_text(size = 8),  
 axis.title = element\_text(size = 9))  
print(tp\_month)

## Warning: Removed 20729 rows containing non-finite values (`stat\_boxplot()`).



tn\_month <- #boxplot of total nitrogen by month  
 ggplot(  
 PeterPaul.chem.nutrients,   
 aes(x = (factor(month, levels = 1:12, labels = month.abb)),   
 y = tn\_ug,  
 color = lakename)) +  
 geom\_boxplot() +  
 labs(x = "Month",  
 y = "Total Nitrogen",  
 col = "Lake") +  
 scale\_color\_manual(values = c("darkolivegreen3", "cyan4")) +  
 theme(axis.text = element\_text(size = 8),  
 axis.title = element\_text(size = 9))  
print(tn\_month)

## Warning: Removed 21583 rows containing non-finite values (`stat\_boxplot()`).



#creating cowplot with all 3 boxplots  
cowplot\_no\_legend <- plot\_grid(temp\_month + theme(legend.position="none"),   
 tp\_month + theme(legend.position="none"),   
 tn\_month + theme(legend.position="none"),   
 nrow = 3,   
 align = 'hv')

## Warning: Removed 3566 rows containing non-finite values (`stat\_boxplot()`).

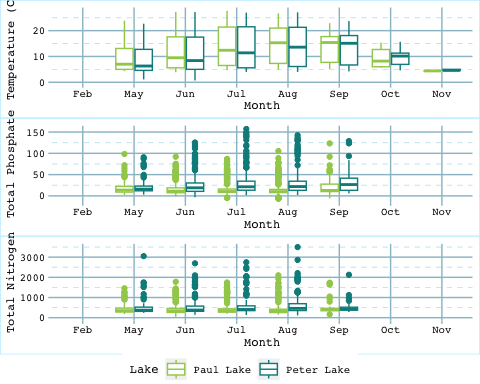
## Warning: Removed 20729 rows containing non-finite values (`stat\_boxplot()`).

## Warning: Removed 21583 rows containing non-finite values (`stat\_boxplot()`).

legend <- get\_legend(  
 tp\_month +  
 theme(legend.position = "bottom",  
 legend.text = element\_text(size = 8),  
 legend.title = element\_text(size = 9)))

## Warning: Removed 20729 rows containing non-finite values (`stat\_boxplot()`).

cowplot <- plot\_grid(cowplot\_no\_legend, legend, ncol = 1, rel\_heights = c(1.2, 0.1)) #creating cowplot of all 3 boxplots with only 1 legend  
print(cowplot)

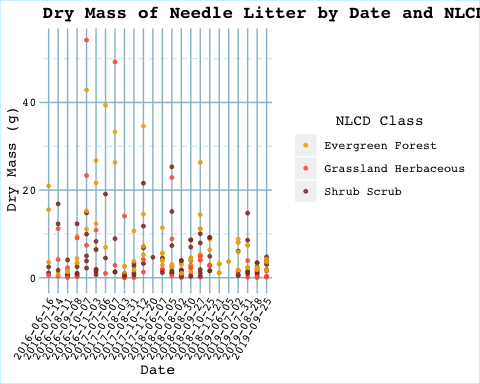


Question: What do you observe about the variables of interest over seasons and between lakes?

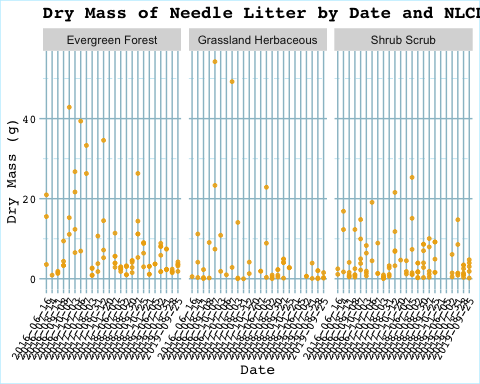
Answer:

1. [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)
2. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

#6  
needles\_drymass <- Niwot.litter.mass %>%  
 filter(functionalGroup == "Needles") %>%  
 ggplot(  
 aes(x = factor(collectDate),  
 y = dryMass,  
 color = nlcdClass)) +   
 geom\_point(size = 1, alpha = 0.85) +   
 labs(  
 title = "Dry Mass of Needle Litter by Date and NLCD Class",  
 x = "Date",  
 y = "Dry Mass (g)",  
 col = "NLCD Class") +  
 scale\_color\_manual(values = c("darkgoldenrod2", "tomato1", "coral4"), labels = c("Evergreen Forest", "Grassland Herbaceous", "Shrub Scrub")) +  
 theme(axis.text.x = element\_text(angle=60, hjust=1))  
print(needles\_drymass)



#7  
needles\_drymass\_faceted <- Niwot.litter.mass %>%  
 filter(functionalGroup == "Needles") %>%  
 ggplot(  
 aes(x = factor(collectDate),  
 y = dryMass)) +   
 geom\_point(size = 1, alpha = 0.85, col = "darkgoldenrod2") +   
 facet\_wrap(vars(nlcdClass), labeller = as\_labeller(c("evergreenForest" = "Evergreen Forest","grasslandHerbaceous" = "Grassland Herbaceous","shrubScrub" = "Shrub Scrub"))) +  
 labs(  
 title = "Dry Mass of Needle Litter by Date and NLCD Class",  
 x = "Date",  
 y = "Dry Mass (g)") +  
 theme(axis.text.x = element\_text(angle=60, hjust=1))  
print(needles\_drymass\_faceted)

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

Answer: I find the plot in #6 to be more effective. It is much easier to compare the dry mass of litter on a certain date between NLCD classes when they are on the same line. I also find it less crowded and easier to read than the graph in #7. #7 has an unneccesary repition of axis.