Assignment 5: Data Visualization

Jessalyn Chuang

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.

```
#1
#Load packages
library(tidyverse);library(lubridate);library(here); library(cowplot)
```

```
## -- 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
## here() starts at /home/guest/EDE_Fall2024
##
##
## Attaching package: 'cowplot'
##
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```
here()
```

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

```
#2
#Load in data
PeterPaul.chem.nutrients <-
read.csv(here("Data/Processed_KEY/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"),
         stringsAsFactors = TRUE)

Niwot_Ridge_litter <-
  read.csv(here("Data/Processed_KEY/NEON_NIWO_Litter_mass_trap_Processed.csv"),
           stringsAsFactors = TRUE)

#turn date columns into date format using lubridate
PeterPaul.chem.nutrients$sampledate <- ymd(PeterPaul.chem.nutrients$sampledate)
Niwot_Ridge_litter$collectDate <- ymd(Niwot_Ridge_litter$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

```
#3
#I chose theme_light() as the base, axis text as set to being black,
#axis titles were set to size 14 with the color of gray, and the legend
#was set to be on the right side in mytheme
mytheme <- theme_light() +
  theme(axis.text = element_text(color = "black"),
        axis.title.x = element_text(size = 14, color = "gray"),
        axis.title.y = element_text(size = 14, color = "gray"),
        legend.position = "right")
```

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.

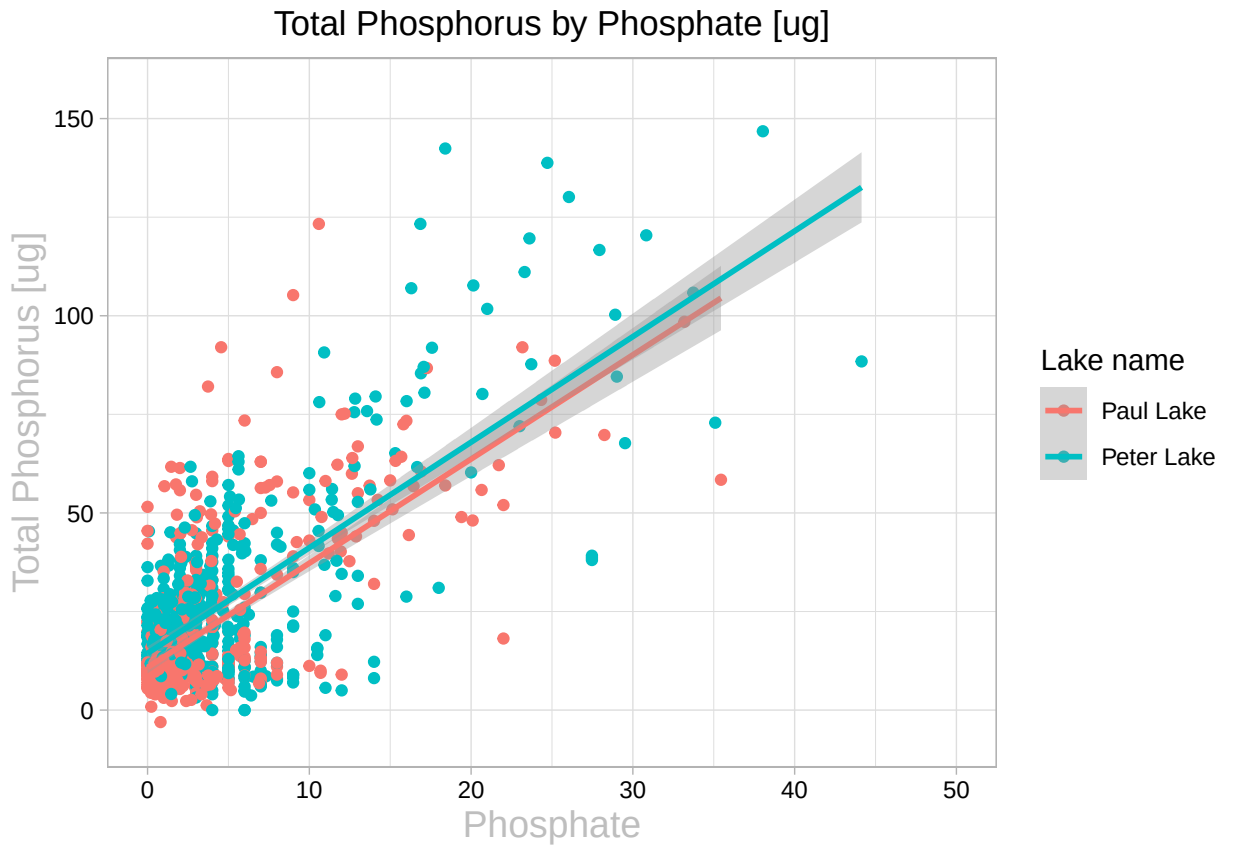
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add line(s) of best fit using the `lm` method. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```
#4
#From the PeterPaul dataset, plot po4 as the x axis and tp_ug as the y axis
#with the color based on the lake name.
#geom_smooth was used to draw the lines of best fit
q4 <- PeterPaul.chem.nutrients %>%
  ggplot(mapping = aes(x = po4, y = tp_ug, color = lakename)) +
  geom_point() +
  xlim(0,50) +
  geom_smooth(method = lm, se = TRUE) +
  mytheme +
  labs(color="Lake name",
       title = "Total Phosphorus by Phosphate [ug]",
       x = "Phosphate",
       y = "Total Phosphorus [ug]") +
  theme(plot.title = element_text(hjust = 0.5))

print(q4)
```

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

```
## Warning: Removed 21947 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 21947 rows containing missing values or values outside the scale range
## (`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.

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`.

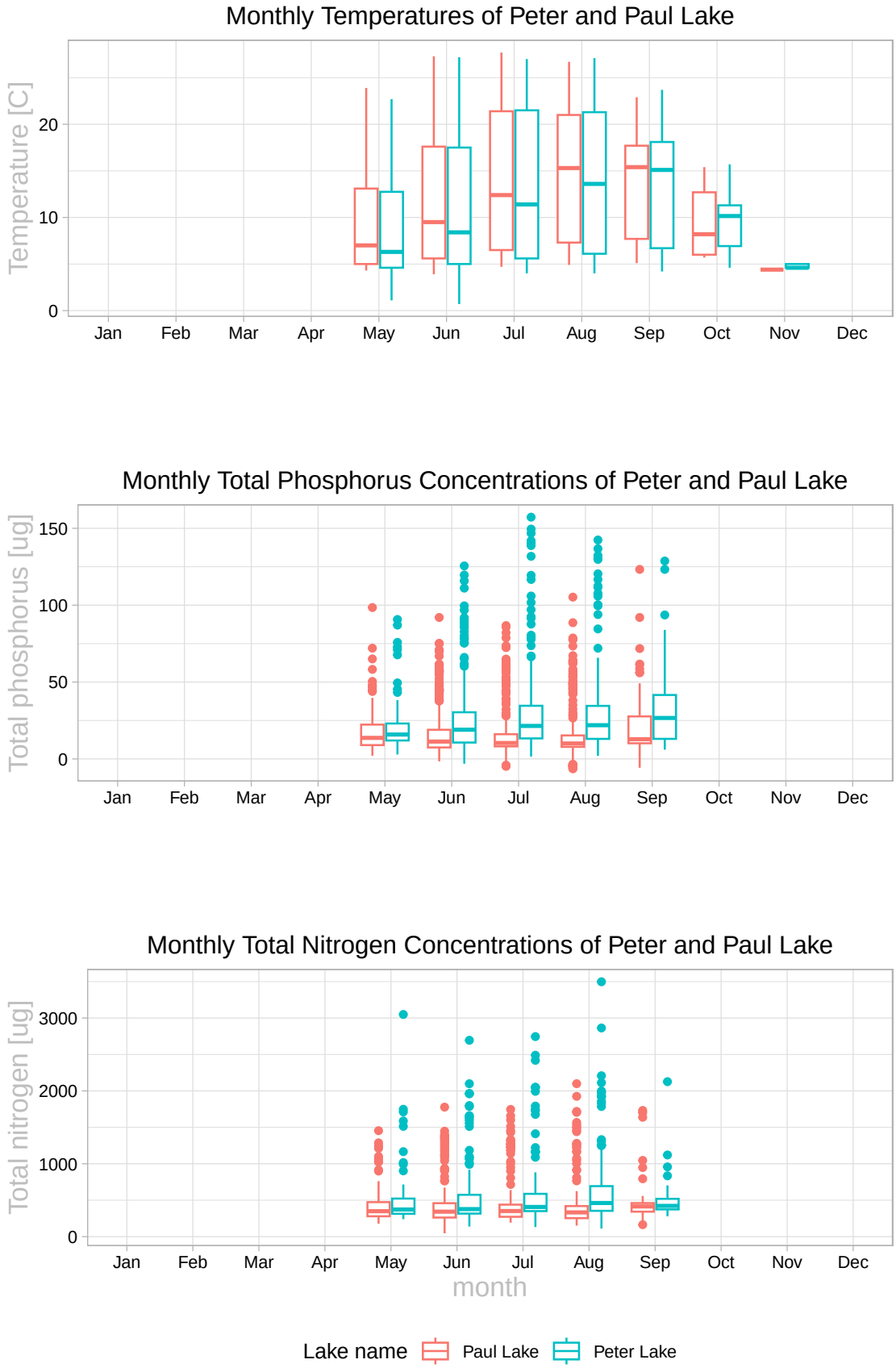
```
#5

#Convert month to a factor -- with 12 levels, labelled with month names
PeterPaul.chem.nutrients$month <- factor(PeterPaul.chem.nutrients$month,
                                         levels = 1:12,
                                         labels = month.abb)

#month vs temperature boxplot created
boxplot_temp <- ggplot(PeterPaul.chem.nutrients, aes(x = month,
                       y = temperature_C,
                       color=lakename)) +
                geom_boxplot() +
                scale_x_discrete(drop = F) +
                labs(title = "Monthly Temperatures of Peter and Paul Lake",
                     color="Lake name",
                     y = "Temperature [C]") +
```

```
                mytheme+
                theme(axis.title.x = element_blank(),
                      legend.position = "none",
                      plot.title = element_text(hjust = 0.5))

#month vs tp_ug boxplot created, the x axis is hidden for this graph
boxplot_TP <- ggplot(PeterPaul.chem.nutrients, aes(x = month,
                      y = tp_ug,
                      color=lakename)) +
              geom_boxplot() +
              scale_x_discrete(drop = F) +
              mytheme +
              labs(title = "Monthly Total Phosphorus Concentrations of Peter and Paul Lake",
                   color="Lake name",
                   y = "Total phosphorus [ug]") +
              theme(axis.title.x = element_blank(),
                    legend.position = "none",
                    plot.title = element_text(hjust = 0.5))

#month vs tn_ug boxplot created, the x axis is hidden for this graph, and
#the legend was added at the bottom of this graph
boxplot_TN <- ggplot(PeterPaul.chem.nutrients, aes(x = month,
                      y = tn_ug,
                      color=lakename)) +
              geom_boxplot() +
              scale_x_discrete(drop = F) +
              mytheme +
              labs(title = "Monthly Total Nitrogen Concentrations of Peter and Paul Lake",
                   color="Lake name",
                   y = "Total nitrogen [ug]") +
              theme(legend.position = "bottom",
                    plot.title = element_text(hjust = 0.5))

#use plot_grid from the cowplot package to arrange plots into three rows
plot_grid(boxplot_temp, boxplot_TP, boxplot_TN,
          nrow = 3, align = 'h', rel_heights = c(1, 1, 1))
```

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

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

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


Monthly Temperatures of Peter and Paul Lake
Temperature [C]
0
10
20
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Monthly Total Phosphorus Concentrations of Peter and Paul Lake
Total phosphorus [ug]
0
50
100
150
Monthly Total Nitrogen Concentrations of Peter and Paul Lake
Total nitrogen [ug]
0
1000
2000
3000
month
Lake name
Paul Lake
Peter Lake

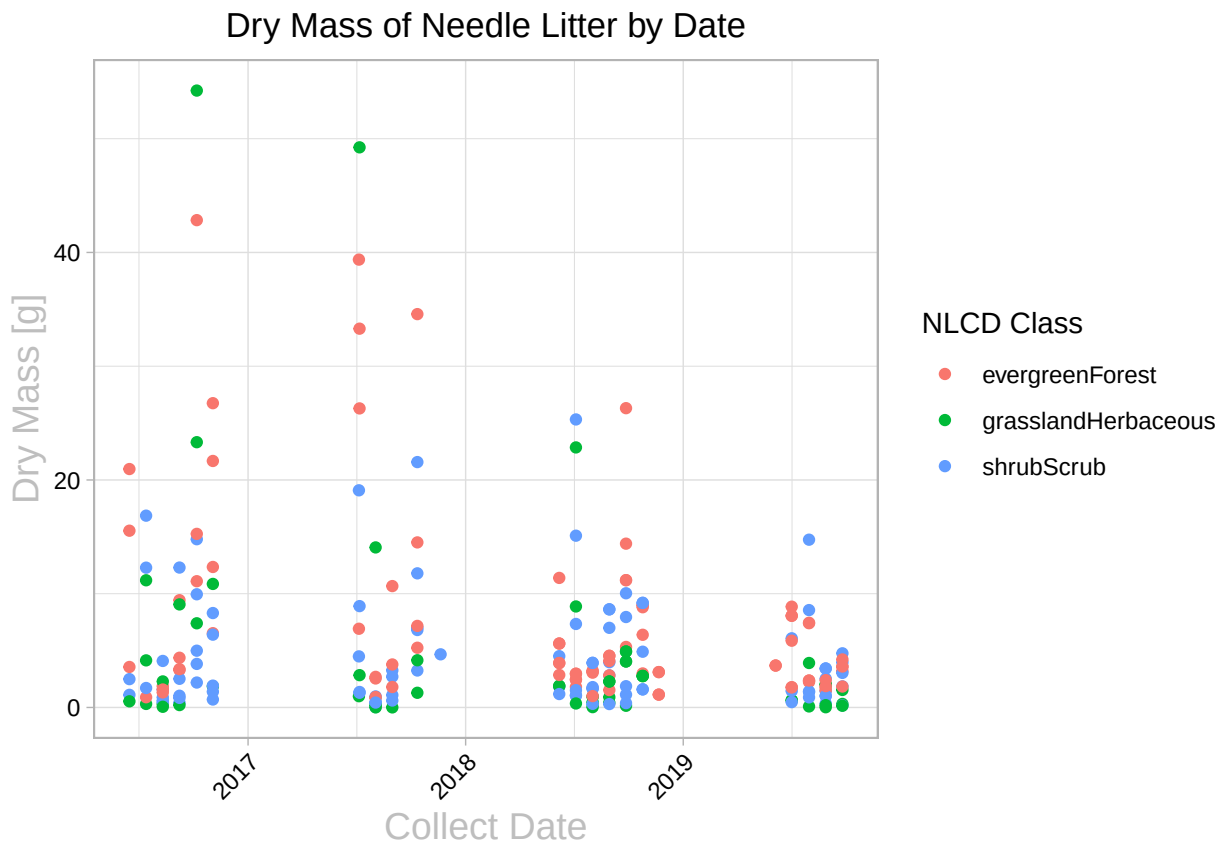
Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Both lakes have similar temperature trends, and temperature increases from May, peaks in the summer months, then drops again by October. Peter Lake has higher phosphorus concentrations than Paul Lake, while Paul Lake has lower variability in phosphorus concentrations compared to Peter Lake. Total Phosphorus concentrations seem to peak in July - Sep. Peter Lake also has higher nitrogen concentrations compared to Paul Lake. The variability in nitrogen concentrations is greater in Peter Lake considering the larger spread of the outlier points for Peter Lake compare to Paul Lake's outlier points. Total nitrogen concentrations also increase starting in May and remain elevated throughout the summer until September.

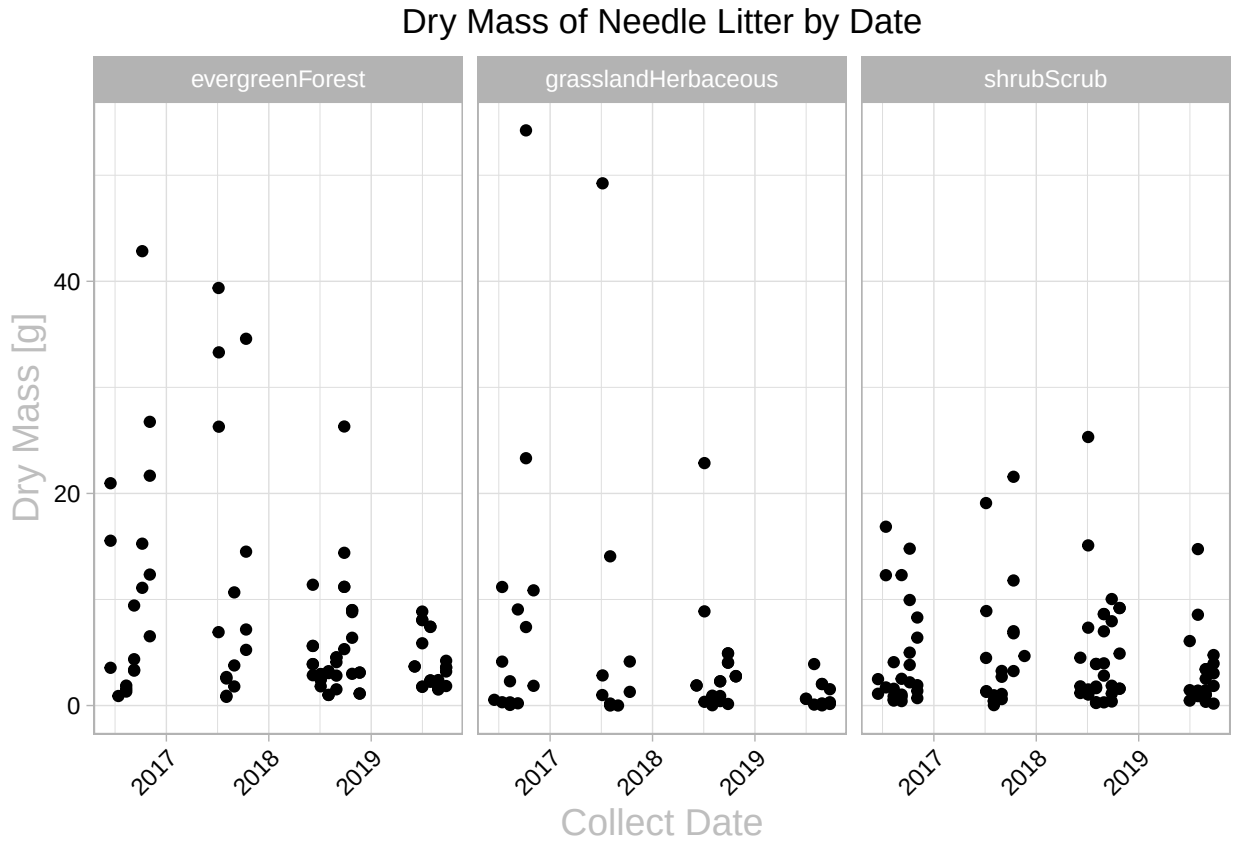
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```
#6
#from the Niwot_Ridge_litter data, filter to keep only the Needles functional
#group, plot collectDate on the x axis, dryMass on the y axis, and color points
#by nlcdClass
litter_Needles <- Niwot_Ridge_litter %>%
                  filter(functionalGroup == "Needles") %>%
                  ggplot(aes(x=collectDate, y=dryMass, color = nlcdClass))+
                  geom_point()+
                  mytheme+
                  theme(axis.text.x = element_text(angle = 45, hjust = 1),
                  plot.title = element_text(hjust = 0.5))+
                  labs(title = "Dry Mass of Needle Litter by Date",
                       x = "Collect Date",
                       y = "Dry Mass [g]",
                       color = "NLCD Class")

print(litter_Needles)
```




```
#7
#This shows the same data above, but now the data is faceted with facet_wrap
#to show each nlcdClass on a separate graph
litter_Needles_2 <- Niwot_Ridge_litter %>%
                  filter(functionalGroup == "Needles") %>%
                  ggplot(aes(x=collectDate, y=dryMass))+
                  geom_point()+
                  facet_wrap(facets = vars(nlcdClass), nrow=1,ncol=3) +
                  mytheme+
                  theme(axis.text.x = element_text(angle = 45, hjust = 1),
                        plot.title = element_text(hjust = 0.5))+
                  labs(title = "Dry Mass of Needle Litter by Date",
                       x = "Collect Date",
                       y = "Dry Mass [g]",
                       color = "NLCD Class")

print(litter_Needles_2)
```



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

Answer: It depends on what you are trying to study. If the goal is to compare the trends of dry mass between the three land cover types directly and with each other, then plot 6 is more effective as the colors allow for a direct comparison in a unified space. If the goal is to focus on the internal distribution of dry mass for each land cover type independently, plot 7 is more effective, as each category is isolated, making it easier to see patterns or anomalies within that specific class.