Assignment 5: Data Visualization

Maia Griffith

Fall 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. 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 Loading packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## 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 error
```

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

```
## here() starts at /home/guest/ENV872/maia-g/EDE_Fall2023
```

```r
library(cowplot)
```

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

```r
library(ggthemes)
```

```
##
## Attaching package: 'ggthemes'
##
## The following object is masked from 'package:cowplot':
##
##     theme_map
```

```r
# Verifying home working directory
here()
```

```
## [1] "/home/guest/ENV872/maia-g/EDE_Fall2023"
```

```r
#Assign a variable to the processed data folder location
processed_data = "Data/Processed_KEY"

# Reading in data files
PetPau.chem.nut <- read.csv(
  here(processed_data,"NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"),
  stringsAsFactors = TRUE)
Litter <- read.csv(
  here(processed_data,"NEON_NIWO_Litter_mass_trap_Processed.csv"),
  stringsAsFactors = TRUE)

#2  I know they are factors because of the `stringAsFactors = TRUE`

# Changing them to Date format
PetPau.chem.nut$sampledate <- ymd(PetPau.chem.nut$sampledate)
Litter$collectDate <- ymd(Litter$collectDate)

# Check the class. They are Date format.
class(PetPau.chem.nut$sampledate)
```

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

```
class(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 Custom theme
MG_theme <- theme_base()  +
  theme(
    plot.title = element_text(
      color = 'maroon',
      size = '14',
      hjust = 0.5),
    axis.title.x = element_text(
      color = 'maroon'),
    axis.title.y = element_text(
      color = 'maroon'),
    plot.background =   element_rect(
      fill = 'white'),
    legend.position = 'bottom',
    )
```

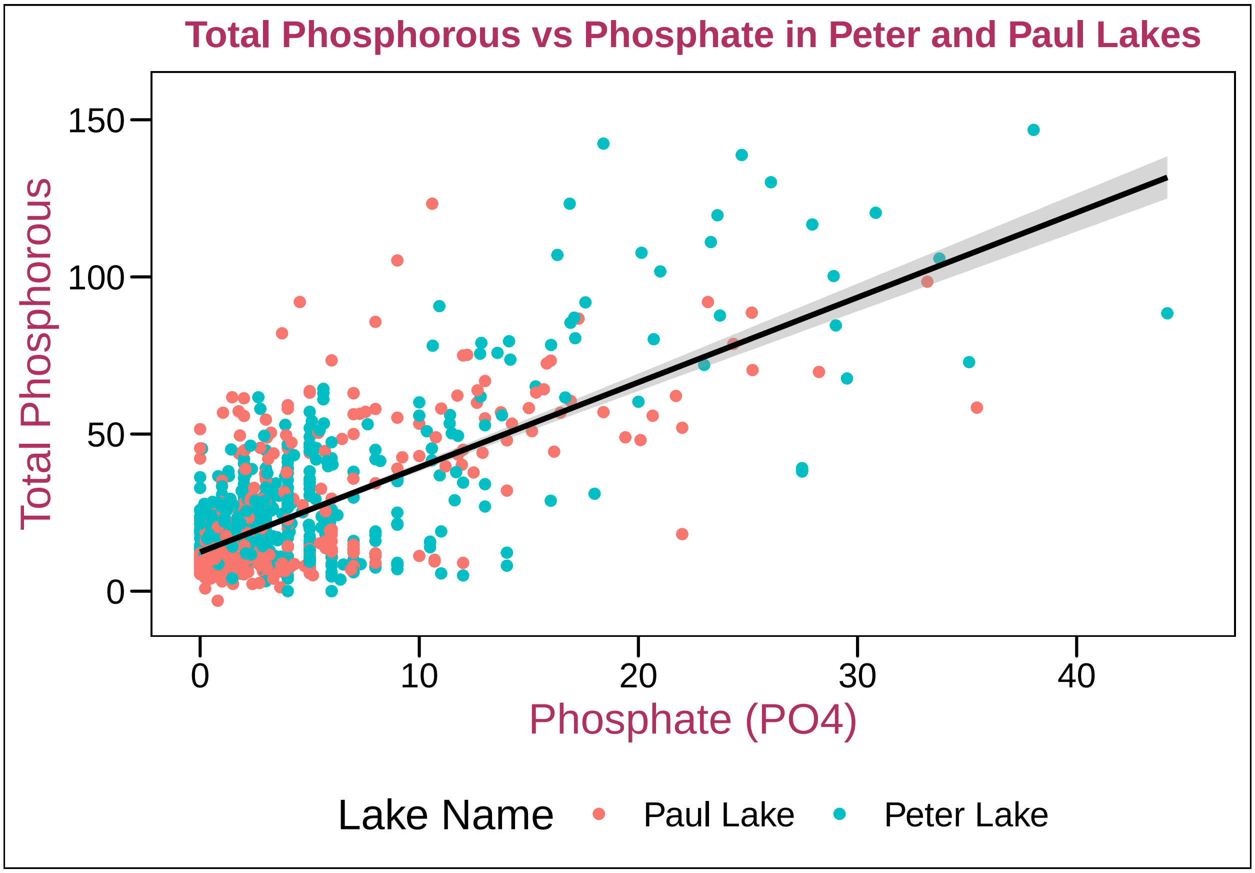
Create graphs

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
theme_set(MG_theme)

tp_po4_plot <-
  ggplot(PetPau.chem.nut,
         aes(x = po4,
             y = tp_ug,
             color = lakename)) +
  geom_point() +
  geom_smooth(method = "lm", color = 'black') +
  xlim(0,45) +
  labs(title = "Total Phosphorous vs Phosphate in Peter and Paul Lakes",
       x = "Phosphate (PO4)",
       y = "Total Phosphorous",
       color = "Lake Name")

tp_po4_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: * Recall the discussion on factors in the previous section as it may be helpful here. * R has a built-in variable called `month.abb` that returns a list of months;see https://r-lang.com/month-abb-in-r-with-example

```
#5

#5a: temperature
```

```r
temp_box <-
  ggplot(PetPau.chem.nut,
         aes(x = factor(month,
                        levels = 1:12,
                        labels = month.abb),
             y = temperature_C,
             color = lakename)) +
  geom_boxplot() +
  labs(title = "Lake Temperature by Month",
       y = "Temperature (C)",
       color = "Lake Name") +
  scale_x_discrete(name = "", drop = FALSE) +
  theme(axis.text.x = element_text(angle = 45,  hjust = 1))

#temp_box

#5b: Total Phosphorous (TP)
TP_box <-
  ggplot(PetPau.chem.nut,
         aes(x = factor(month,
                        levels = 1:12,
                        labels = month.abb),
             y = tp_ug,
             color = lakename)) +
  geom_boxplot() +
  labs(title = "Total Phosphorous by Month",
       y = "Total Phosphorous",
       x = "",
       color = "Lake Name") +
  scale_x_discrete(name = "", drop = FALSE) +
  theme(axis.text.x = element_text(angle = 45,  hjust = 1),
        plot.margin = unit(c(1,1,1,1), "cm"))

#TP_box

#5c: Total Nitrogen (TN)
TN_box <-
  ggplot(PetPau.chem.nut,
         aes(x = factor(month,
                        levels = 1:12,
                        labels = month.abb),
             y = tn_ug,
             color = lakename)) +
  geom_boxplot() +
  labs(title = "Total Nitrogen by Month",
       y = "Total Nitrogen",
       x = "Month",
       color = "Lake Name") +
  scale_x_discrete(name = "Month", drop = FALSE) +
  theme(axis.text.x = element_text(angle = 45,  hjust = 1))

#TN_box
```

```
#Getting the legend only in cowplot
my_legend <- get_legend(TN_box)
```

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

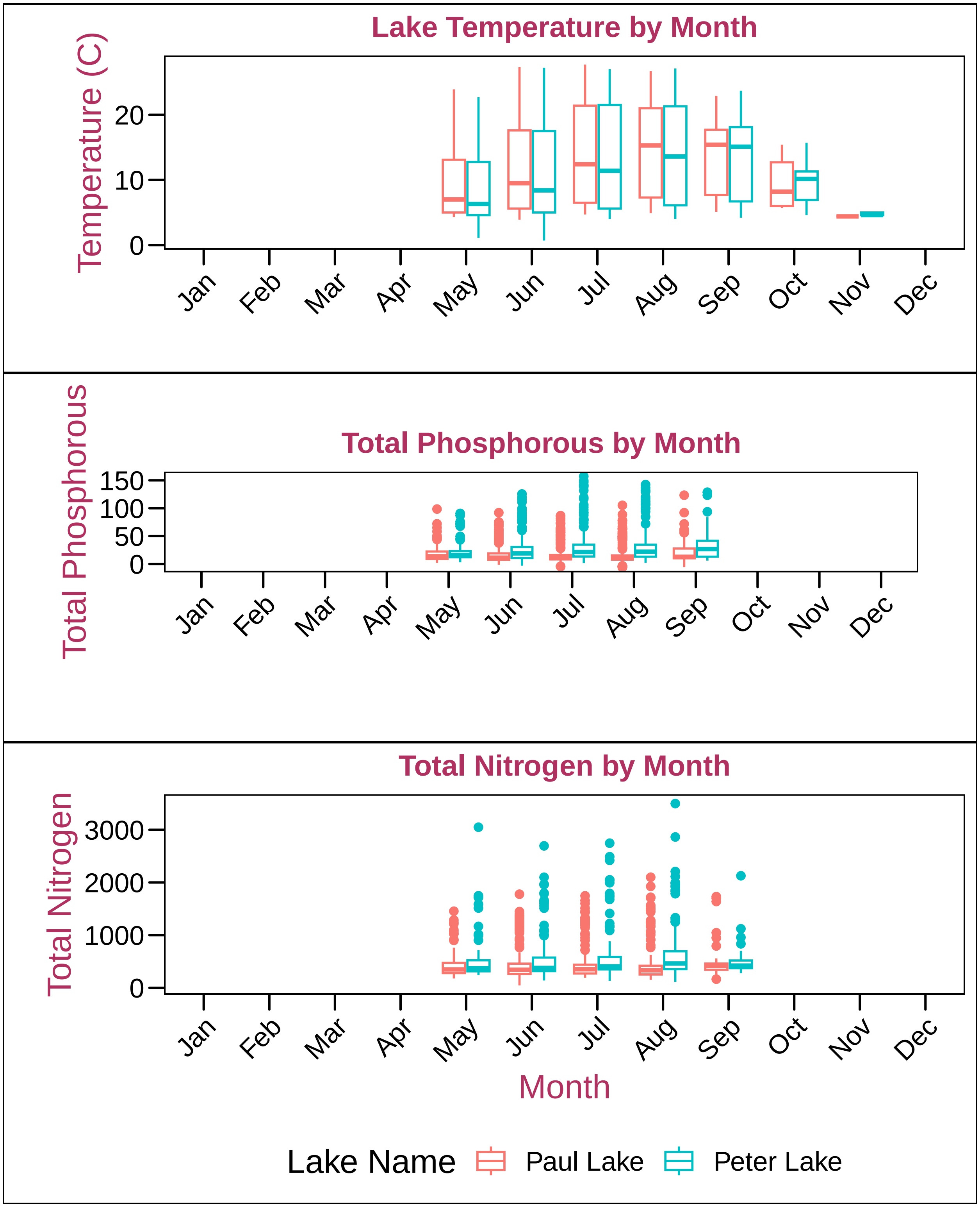
```
  # create some space to the left of the legend


# Creating a cowplot with all 3 plots.
combo_Q5 <-
  plot_grid(
            temp_box + theme(legend.position = "none"),
            TP_box + theme(legend.position = "none"),
            TN_box + theme(legend.position = "bottom"),
            ncol = 1, nrow = 3, align = 'v', axis = 'l', rel_heights = c(1,1,1.25))
```

```
## 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()`).
```

```
combo_Q5
```



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

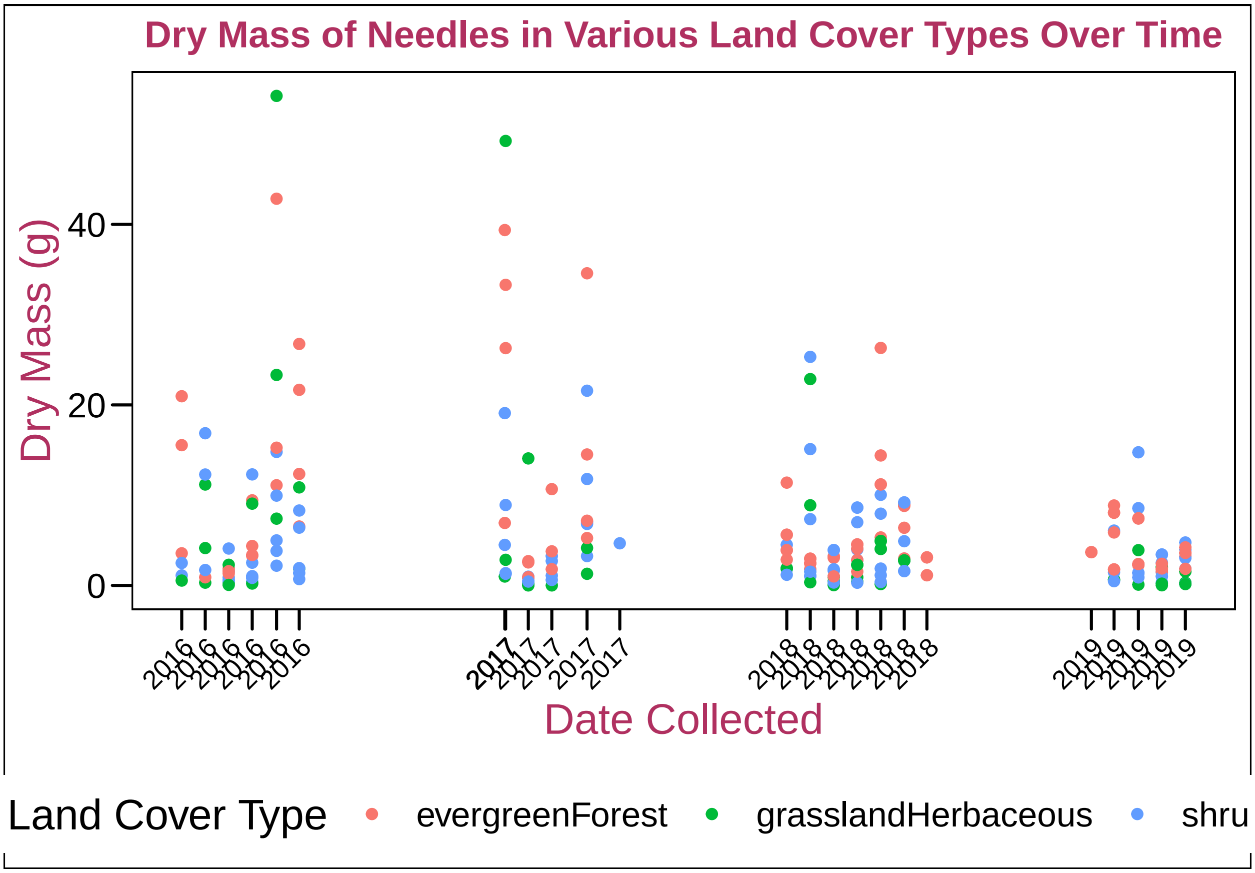
> Answer: Temperature has a more clear distribution of cooler in both lakes before summer begins, with increasing temps during the summer months of June, July, and August. Then the temps in both lakes start to decline again heading into fall and winter. The total phosphorous in Paul Lake

did not seem to have as big of changes as Peter Lake, especially in July and August. Similarly, the Nitrogen content of Peter Lake had larger variation and higher values than Paul Lake throughout the months.

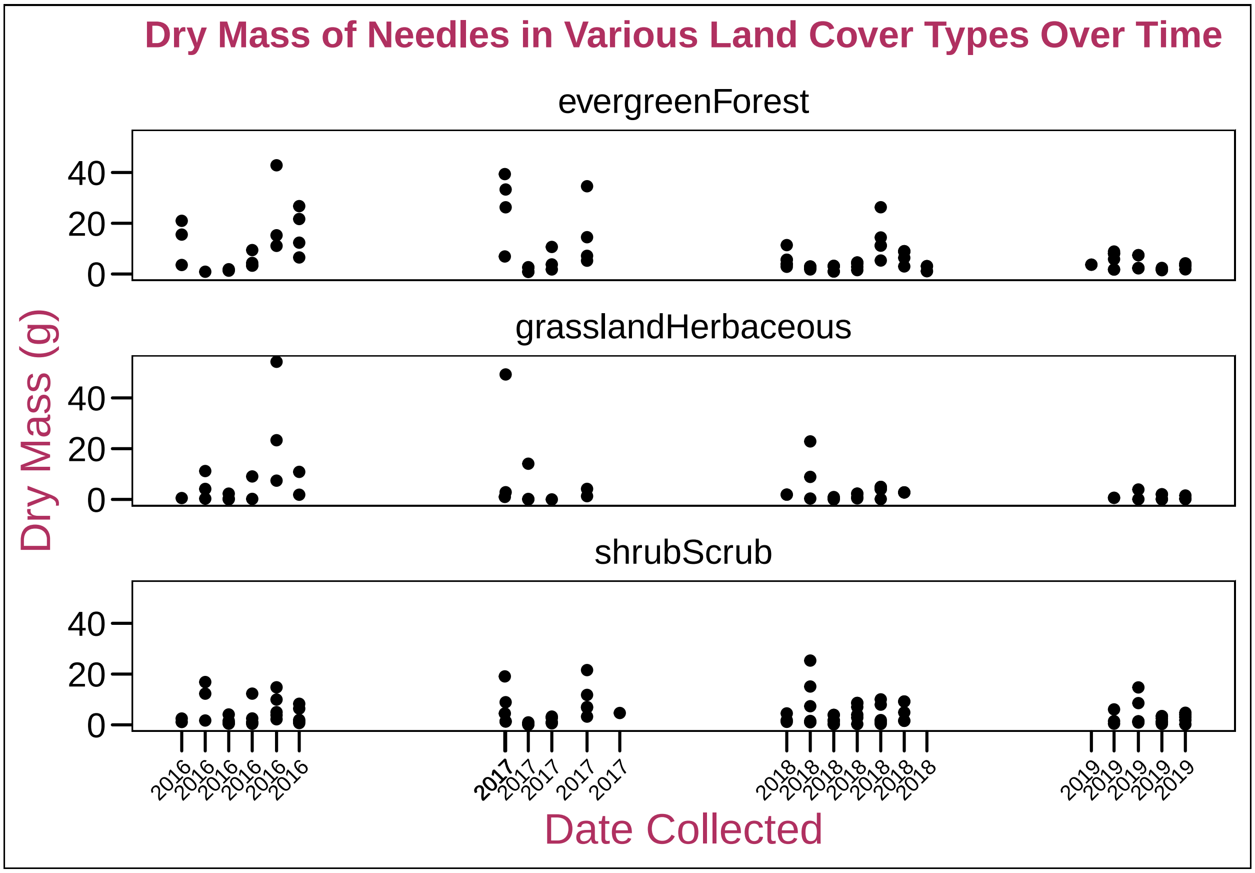
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

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

```r
#6 Plotting subset of Litter by date.
needles_Q6_plot <- Litter %>%
  filter(functionalGroup == "Needles") %>% #filtering by Needles
  ggplot(aes(x = collectDate, #x-axis is date
             y = dryMass, #y-axis is dry mass of needles
             color = nlcdClass)) + #separate by NLCD using color
  geom_point() +
  labs(title = "Dry Mass of Needles in Various Land Cover Types Over Time",
       y = "Dry Mass (g)",
       x = "Date Collected",
       color = "Land Cover Type") +
  scale_x_date(name = "Date Collected",
               date_labels = "%Y",
               breaks = unique(Litter$collectDate)) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1,
                                   size = 10))

needles_Q6_plot
```



```
#7
needles_Q7_facet <- Litter %>%
  filter(functionalGroup == "Needles") %>% #filtering by Needles
  ggplot(aes(x = collectDate, #x-axis is date
             y = dryMass)) + #y-axis is dry mass of needles
  geom_point() +
  labs(title = "Dry Mass of Needles in Various Land Cover Types Over Time",
       y = "Dry Mass (g)",
       x = "Date Collected",
       color = "Land Cover Type") +
  scale_x_date(name = "Date Collected",
               date_labels = "%Y",
               breaks = unique(Litter$collectDate)) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1,
                                   size = 8)) +
  facet_wrap(vars(nlcdClass), nrow = 3)

needles_Q7_facet
```



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

Answer: I think the plot for Q7 (`needles_Q7_facet`) is more effective because it allows the viewer to really compare the differences across land cover types very clearly. The plot from Q6 was pretty good, but because of the amount of points it made comparing the drymass per land cover type a bit more difficult, as so many points overlapped each other. Overall, the plot in Q6 was just too busy, so being able to separate them and view each side by side in Q7 was much more useful.