

Assignment 5: Data Visualization

Atalie Fischer

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Visualization

Directions

1. Change “Student Name” on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., “Fay_A05_DataVisualization.Rmd”) prior to submission.

The completed exercise is due on Monday, February 14 at 7:00 pm.

Set up your session

1. Set up your session. Verify your working directory and load the tidyverse and cowplot packages. 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
getwd()

## [1] "/Users/ataliefischer/Desktop/EDA/Environmental_Data_Analytics_2022/Assignments"
setwd("/Users/ataliefischer/Desktop/EDA/Environmental_Data_Analytics_2022")

library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --
## v ggplot2 3.3.5      v purrr  0.3.4
## v tibble  3.1.4      v dplyr  1.0.7
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   2.0.1      v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(cowplot)

PeterPaul_data <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")
```

```

      stringsAsFactors = TRUE)
Litter_data <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
      stringsAsFactors = TRUE)

#2
PeterPaul_data$sampldate <- as.Date(PeterPaul_data$sampldate,
      format = "%Y-%m-%d")
Litter_data$collectDate <- as.Date(Litter_data$collectDate,
      format = "%Y-%m-%d")

```

Define your theme

3. Build a theme and set it as your default theme.

```

#3
mytheme <- theme_classic(base_size = 12) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")

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.

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 ylim()).

```

#4
ggplot(PeterPaul_data, aes(x = tp_ug, y = po4, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, color = "black") +
  xlim(0, 150) + xlab(expression("Total Phosphorus (ug)")) +
  ylim(0, 50) + ylab(expression("Phosphate")) +
  labs(color="Lakes")

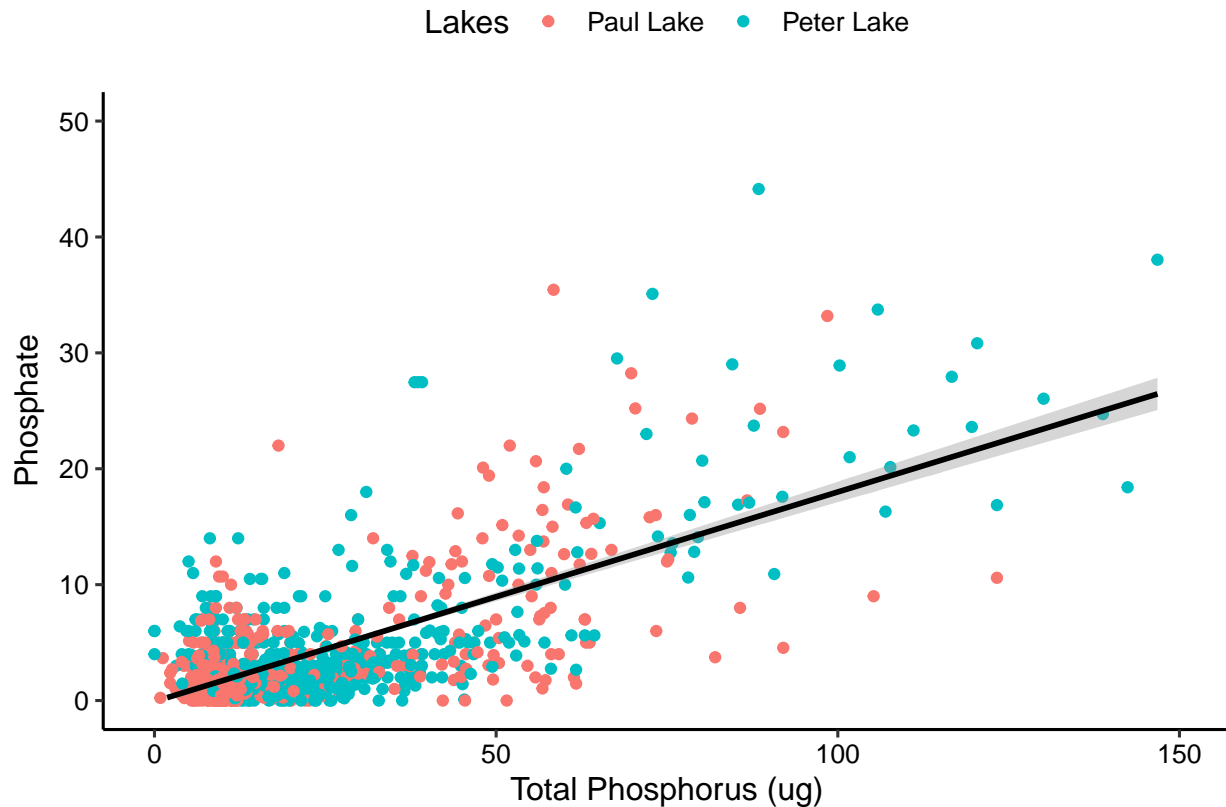
```

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

```
## Warning: Removed 21948 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 21948 rows containing missing values (geom_point).
```

```
## Warning: Removed 1 rows containing missing values (geom_smooth).
```

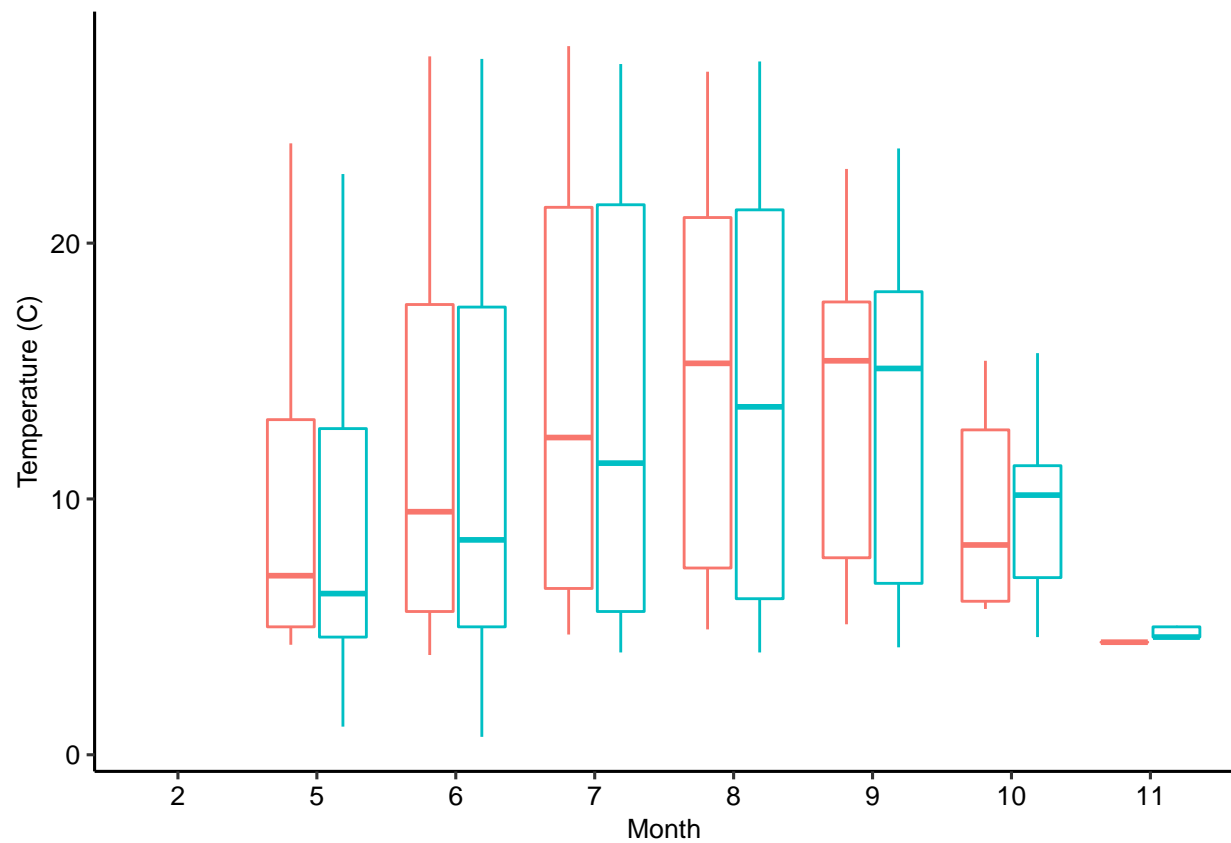


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

```
#5
PeterPaul_data$month <- as.factor(PeterPaul_data$month)

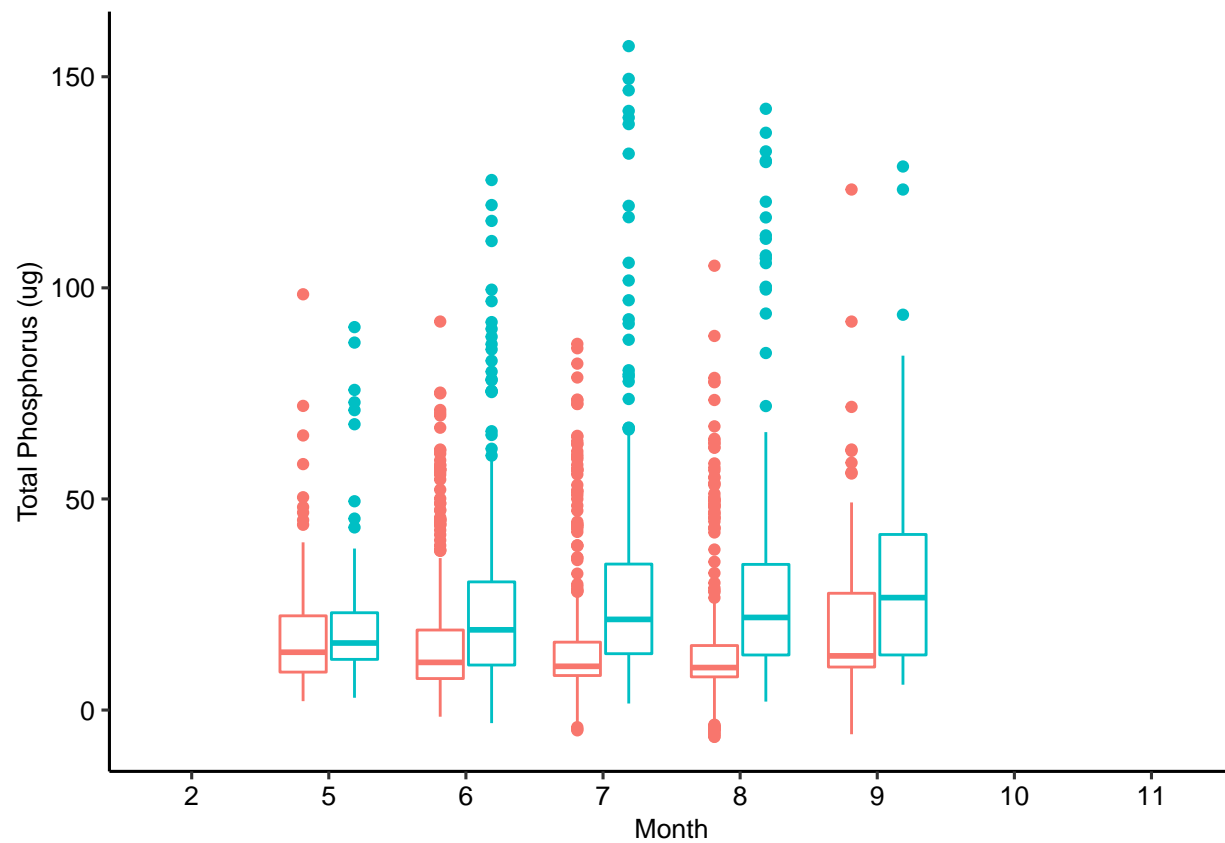
temperature.plot<-
  ggplot(PeterPaul_data, aes(color = lakename)) +
    geom_boxplot(aes(x = month, y = temperature_C)) +
    ylab("Temperature (C)") +
    xlab("Month") +
    theme(legend.position = "none",
          axis.title = element_text(size = 10))
print(temperature.plot)
```

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



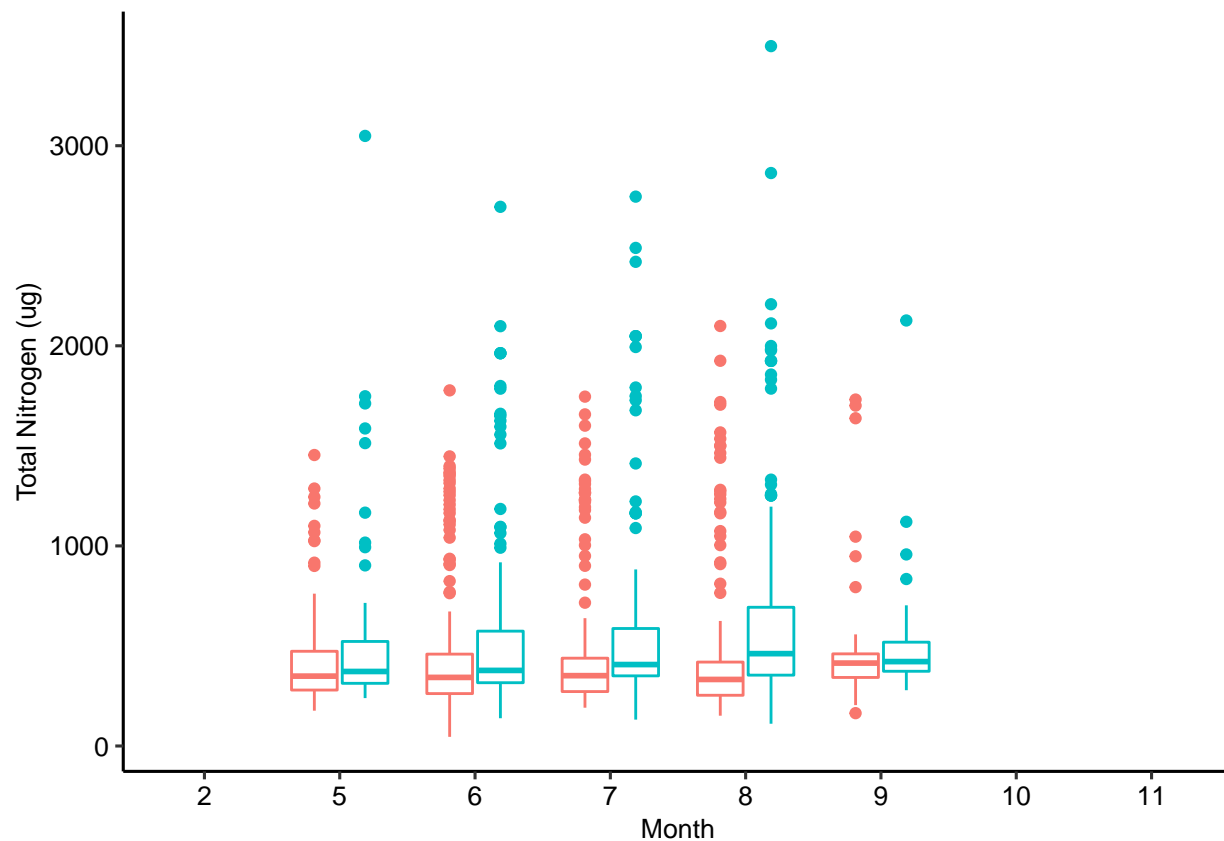
```
TP.plot<-
  ggplot(PeterPaul_data, aes(color = lakename)) +
    geom_boxplot(aes(x = month, y = tp_ug)) +
    ylab("Total Phosphorus (ug)") +
    xlab("Month") +
    theme(legend.position = "none",
          axis.title = element_text(size = 10))
print(TP.plot)
```

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



```
TN.plot<-
  ggplot(PeterPaul_data, aes(color = lakename)) +
    geom_boxplot(aes(x = month, y = tn_ug)) +
    ylab("Total Nitrogen (ug)") +
    xlab("Month") +
    theme(legend.position = "none",
          axis.title = element_text(size = 10))
print(TN.plot)
```

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



```
legend <- get_legend(temperature.plot +
  guides(color = guide_legend(nrow = 1)) +
  theme(legend.position = "top"))
```

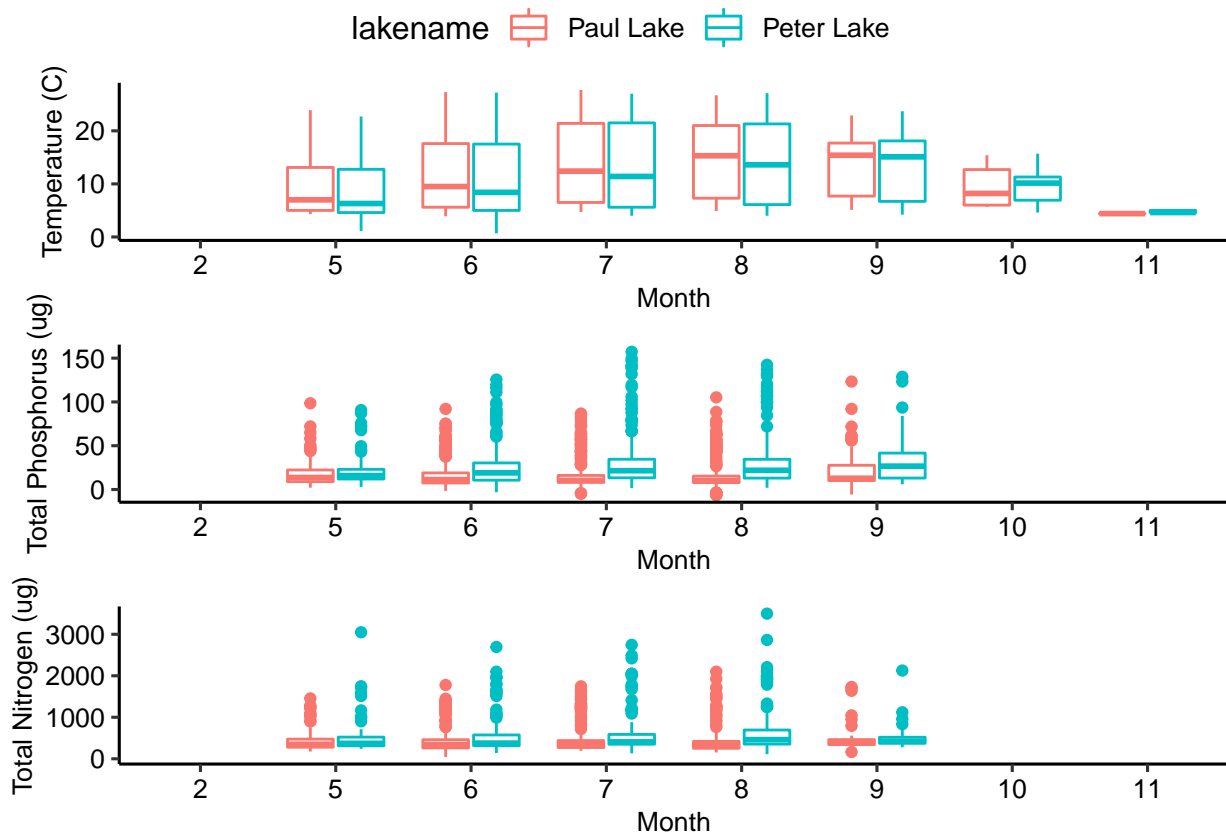
```
## Warning: Removed 3566 rows containing non-finite values (stat_boxplot).
```

```
plot_grid(legend, temperature.plot, TP.plot, TN.plot,
  nrow = 4, ncol = 1,
  align = 'v', axis = "l", rel_heights = c(0.3, 1, 1, 1))
```

```
## 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?

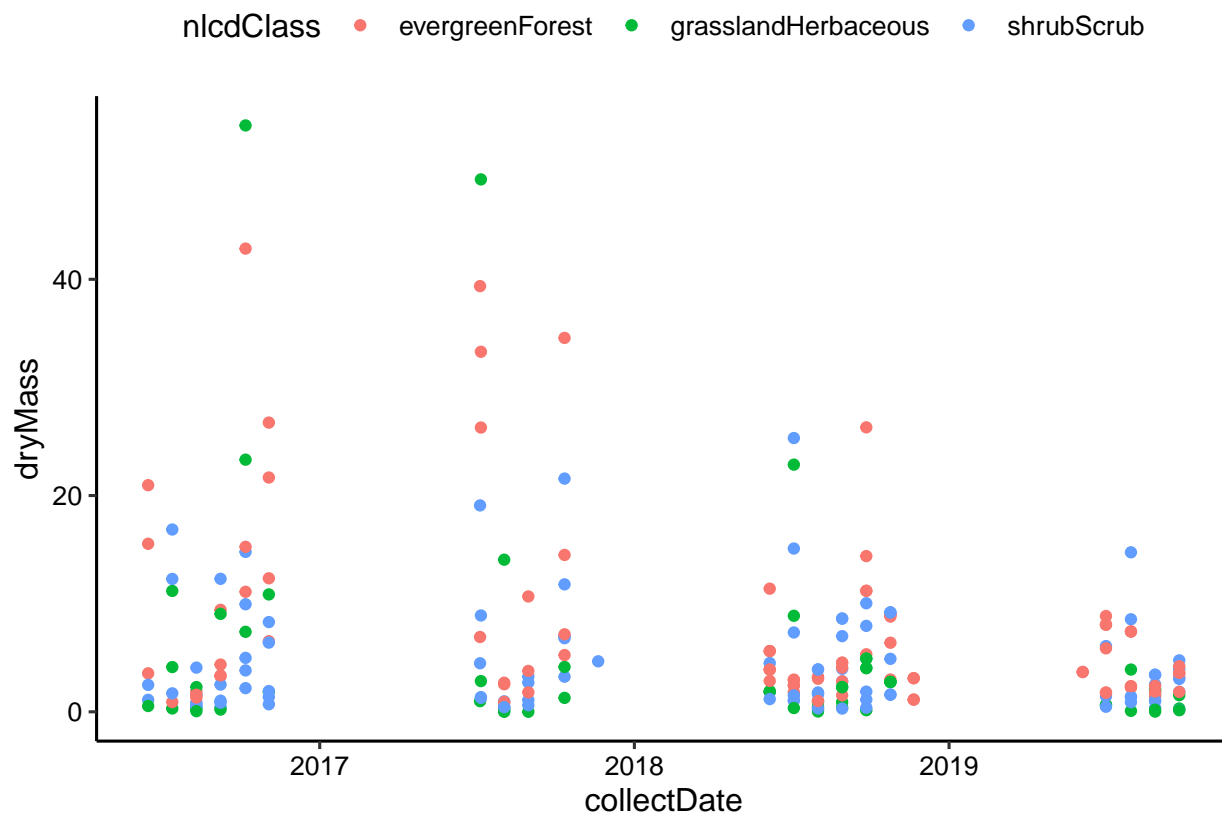
Answer: The median temperature of Paul Lake is generally higher than the median temperature of Peter Lake. The temperatures of both lakes increase from May through August, and decrease from August through November, with highest temperatures of both lakes occurring in August.

Peter Lake has greater median total phosphorus and total nitrogen. Total phosphorus in Peter Lake increases at a faster rate than Paul Lake from May through September. Total nitrogen in Peter Lake peaks in August.

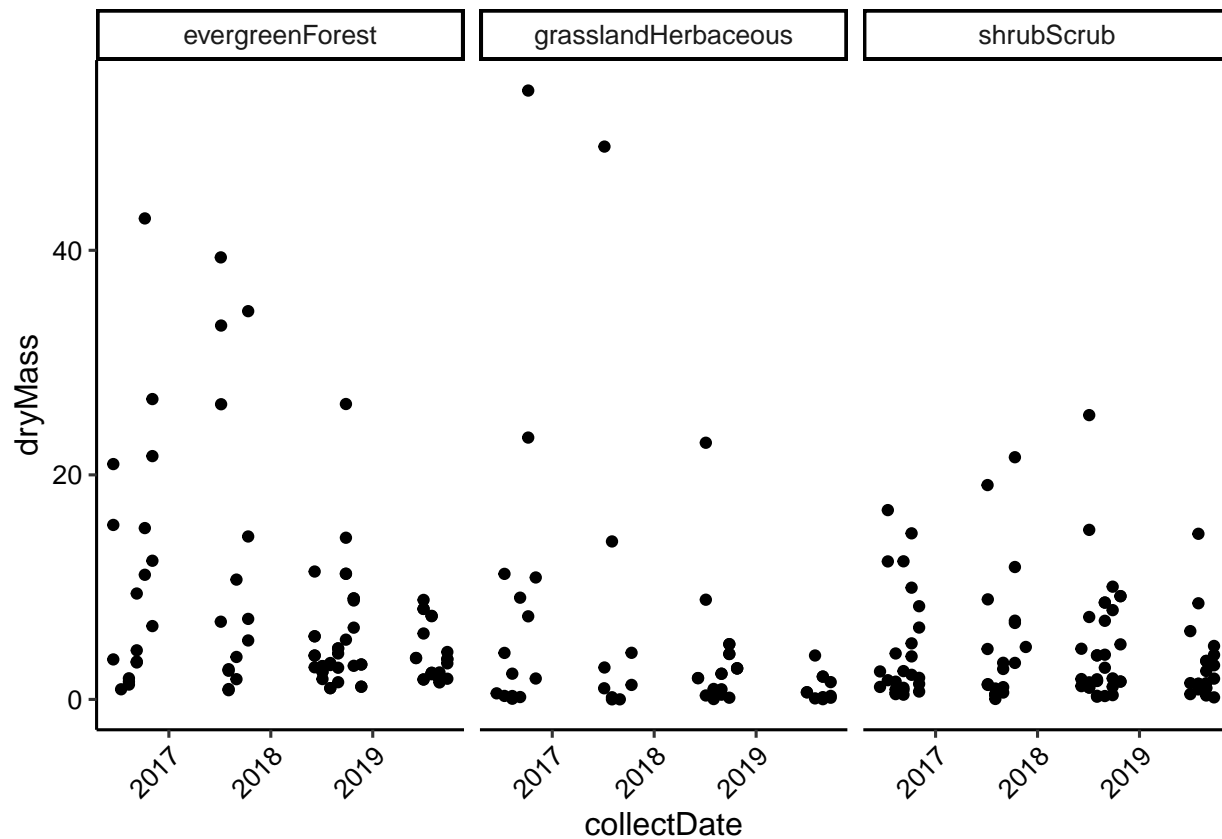
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the “Needles” functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than separated by color.

#6

```
ggplot(subset(Litter_data, functionalGroup == "Needles")) +
  geom_point(aes(x = collectDate, y = dryMass, color = nlcdClass))
```



```
#7
ggplot(subset(Litter_data, functionalGroup == "Needles"),
  aes(x = collectDate, y = dryMass)) +
  geom_point() +
  facet_wrap(vars(nlcdClass)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

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

Answer: I think plot #7 is more effective because it is easier to see the spread of dry mass for each NLCD class over time. For example, it is easy to see that the dry mass of evergreen forest is decreasing over time while the dry mass of shrubs is relatively constant. This is more difficult to see in plot #6.