Assignment 5: Data Visualization

Yanxi Peng

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A02_CodingBasics.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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

The completed exercise is due on Friday, Oct 14th @ 5:00pm.

Set up your session

1. Set up your session. Verify your working directory and load the tidyverse, lubridate, & 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.c`
version).

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

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

```
library(cowplot)
```

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

```
getwd()
```

```
## [1] "C:/Users/16920/Documents/R/EDA-Fall2022/Assignments"
```

```
setwd("C:/Users/16920/Documents/R/EDA-Fall2022")
NTL_LTER <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
    stringsAsFactors = TRUE)
Niwot_Ridge <- read_csv("Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")
```

```
## Rows: 1692 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): plotID, trapID, functionalGroup, qaDryMass, nlcdClass, plotType, g...
## dbl  (5): dryMass, subplotID, decimalLatitude, decimalLongitude, elevation
## date (1): collectDate
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

2. Make sure R is reading dates as date format; if not change the format to date.

```
class(NTL_LTER$sampledate)
```

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

```
class(Niwot_Ridge$collectDate)
```

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

```
NTL_LTER$sampledate <- ymd(NTL_LTER$sampledate)
Niwot_Ridge$collectDate <- ymd(Niwot_Ridge$collectDate)
```

Define your theme

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

```
mytheme <- theme_classic(base_size = 14) + 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.

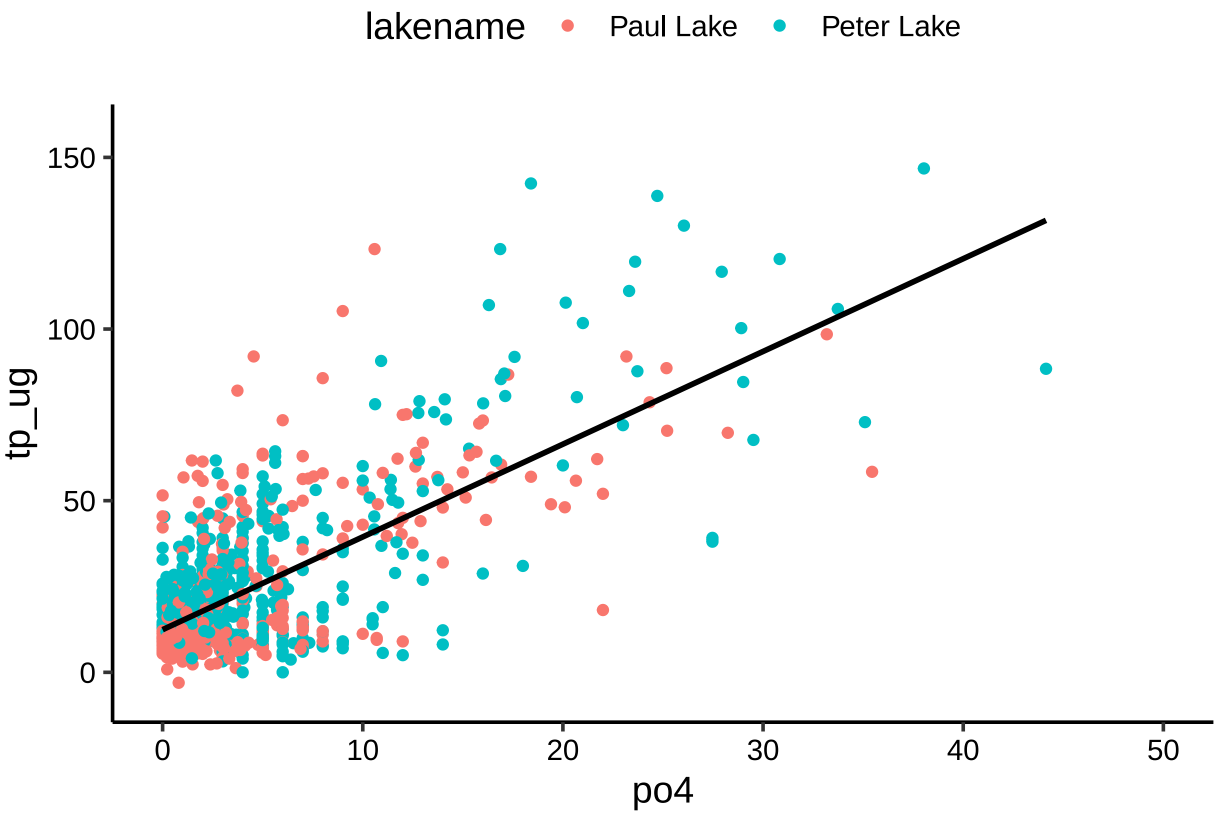
4. [NTL-LTER] Plot total phosphorus (`tp_ug`) by phosphate (`po4`), with separate aesthetics for Peter and Paul lakes. Add a line of best fit and color it black. Adjust your axes to hide extreme values (hint: change the limits using `xlim()` and/or `ylim()`).

```
plot4 <- ggplot(NTL_LTER, aes(x = po4, y = tp_ug, color = lakename)) + geom_point() +
    geom_smooth(method = lm, se = FALSE, color = "black") + xlim(0, 50)
plot4
```

```
## `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).
```

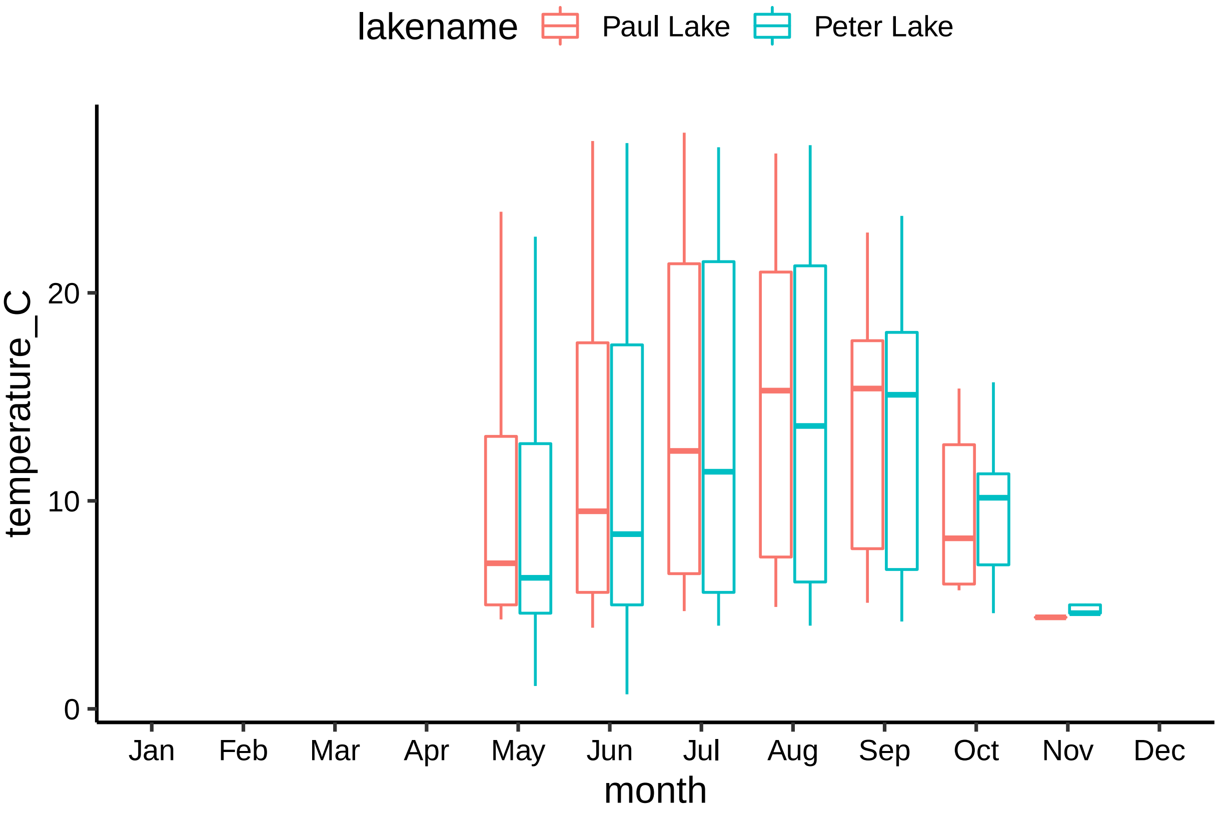


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

Tip: 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

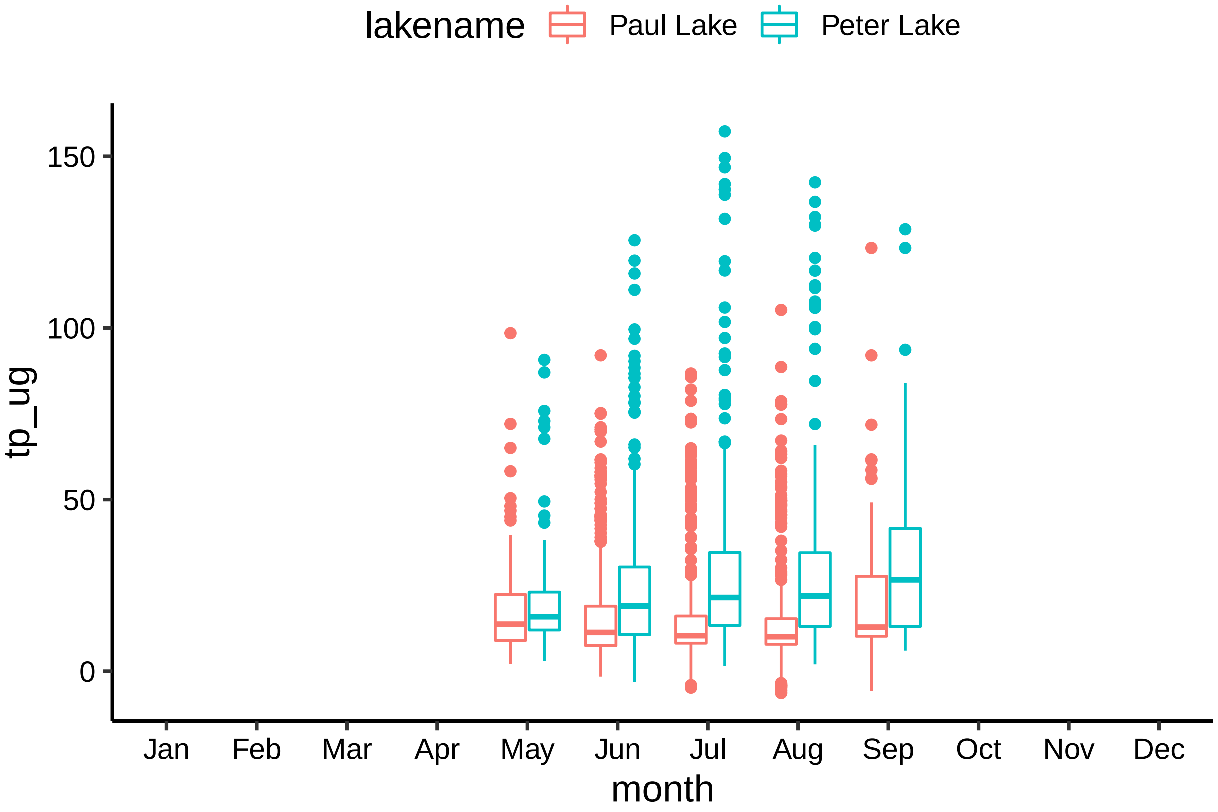
```
NTL_LTER$month <- month.abb[NTL_LTER$month]

NTL_LTER$month <- factor(NTL_LTER$month, levels = month.abb)

# a) temperature
plot5_temp <- ggplot(NTL_LTER, aes(x = month, y = temperature_C, color = lakename)) +
    geom_boxplot() + scale_x_discrete(drop = F)
plot5_temp
```

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



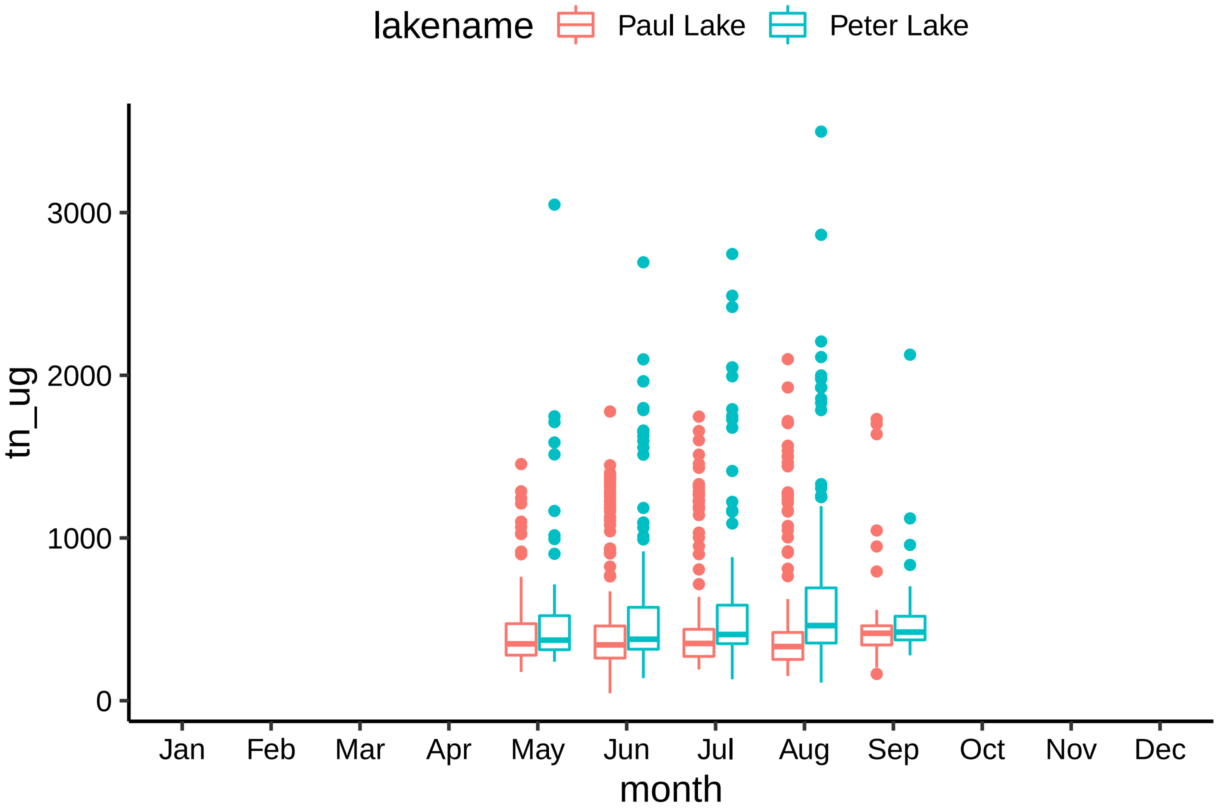
```
# b) TP
plot5_TP <- ggplot(NTL_LTER, aes(x = month, y = tp_ug, color = lakename)) + geom_boxplot() +
    scale_x_discrete(drop = F)
plot5_TP
```

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



```
# c) TN
plot5_TN <- ggplot(NTL_LTER, aes(x = month, y = tn_ug, color = lakename)) + geom_boxplot() +
    scale_x_discrete(drop = F)
plot5_TN
```

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



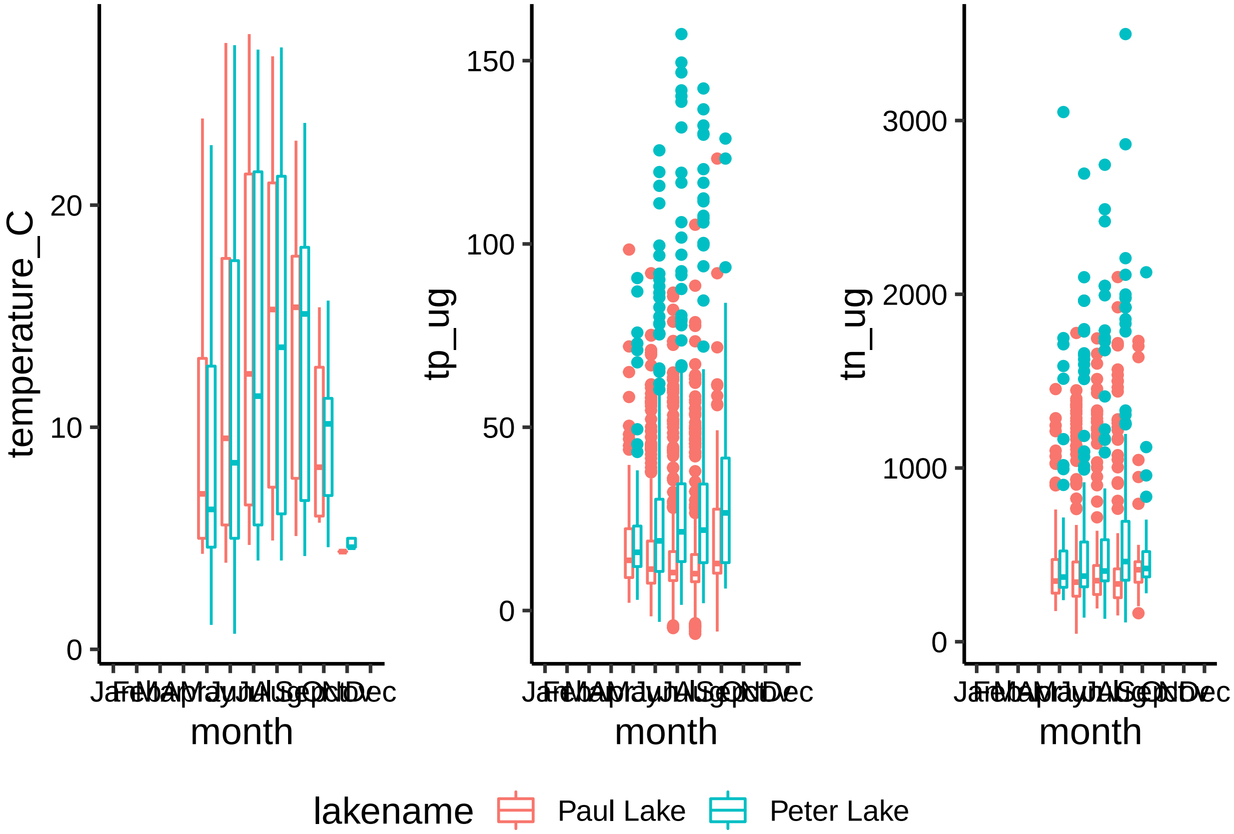
```
legend <- get_legend(plot5_TP)
```

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

```
threeplots <- plot_grid(plot5_temp + theme(legend.position = "none"), plot5_TP +
    theme(legend.position = "bottomright"), plot5_TN + theme(legend.position = "none"),
    align = "h", axis = "tb", nrow = 1, rel_widths = c(3, 3))
```

```
## 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).
```

```
plot_grid(threeplots, legend, nrow = 2, rel_heights = c(3, 0.3))
```

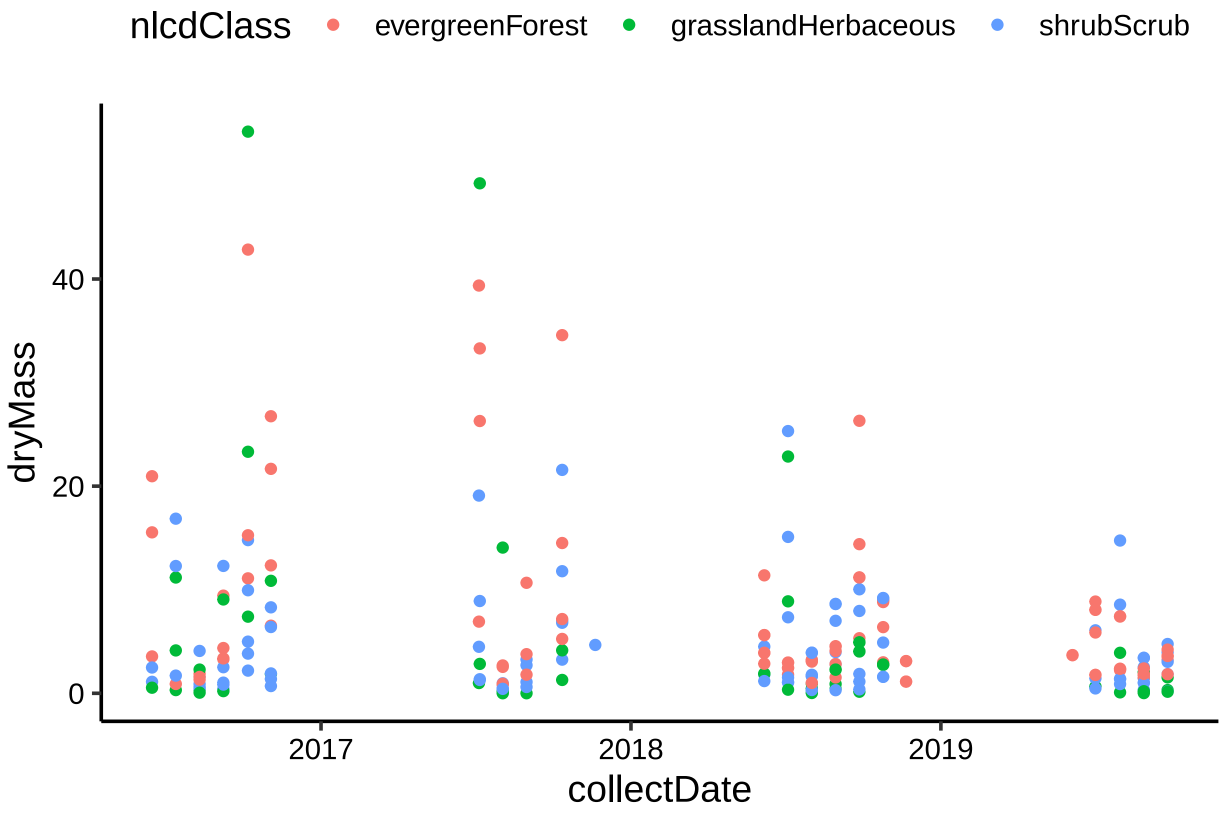


Question: What do you observe about the variables of interest over seasons and between lakes?

> Answer: Between lakes, Paul Lake has higher median temperature value in May, Jun, Jul, Aug and Sep than Peter Lake. For the median tp_ug and tn_ug, Paul Lake has higher value than Peter. Between seasons, the temperature increase from May to Aug and then decrease till Nov. The median tp_ug and tn_ug value did not change much in different seasons.

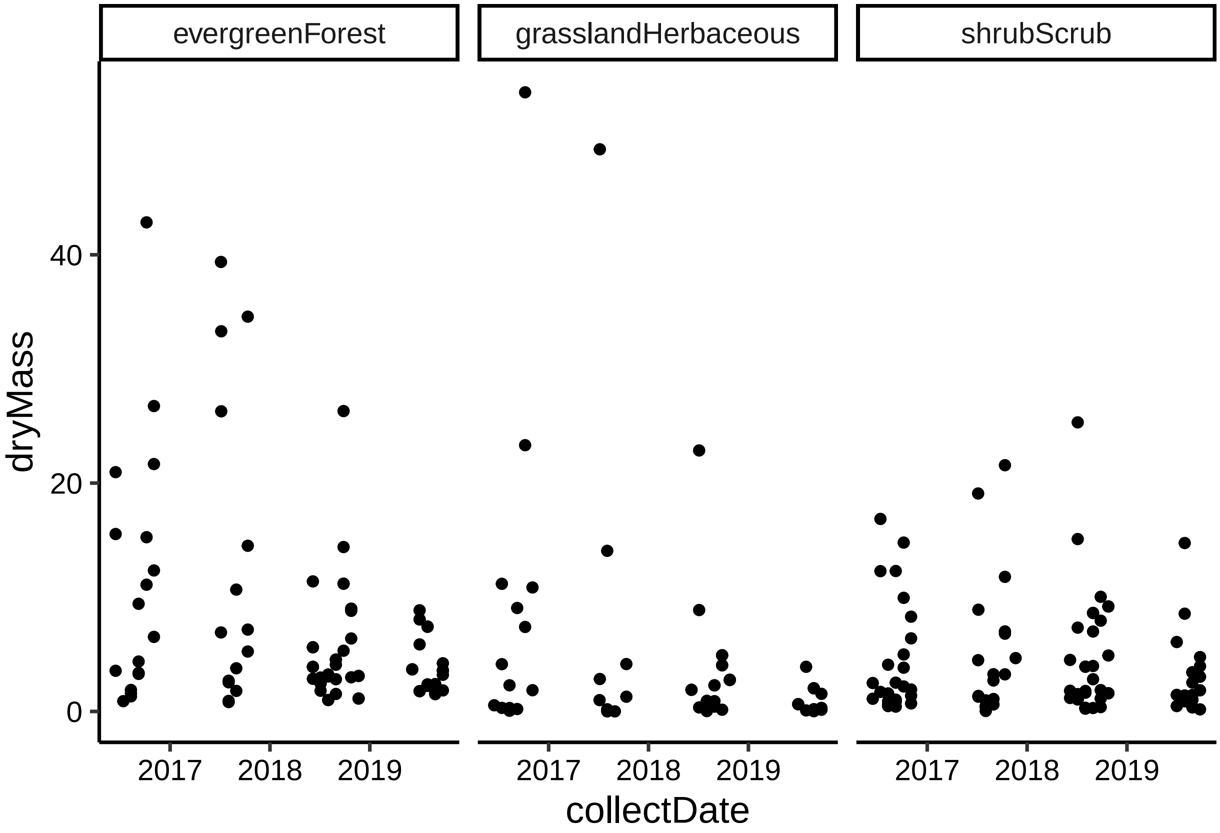
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

```
Needles_NR <- subset(Niwot_Ridge, functionalGroup %in% c("Needles"))
plot6 <- ggplot(Needles_NR, aes(x = collectDate, y = dryMass, color = nlcdClass)) +
    geom_point()

plot6
```



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

```
plot7 <- ggplot(Needles_NR, aes(x = collectDate, y = dryMass)) + geom_point() + facet_wrap(~nlcdClass)
plot7
```



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

Answer:Plot in question 7 is more effective. Because plot 7 is more clear which separate the different classes into different sub-plot. Rather than use color difference in one plot, separated into three facets are more clear and all the point would not gather together and easy to see the difference between each classes or amoung same class.