Assignment 5: Data Visualization

Emma DeAngeli

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] "C:/Users/Emma DeAngeli/Documents/Environmental Data Analytics/Assignments"
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5

## Warning: package 'tibble' was built under R version 4.0.5

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'readr' was built under R version 4.0.5

## Warning: package 'purrr' was built under R version 4.0.5

## Warning: package 'dplyr' was built under R version 4.0.5

## Warning: package 'stringr' was built under R version 4.0.5

## Warning: package 'forcats' was built under R version 4.0.5

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

```
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.0.5
```

```
PeterPaul.chem.nutrients <-
  read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv")
NiwotRidge.litter <-
  read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv")

#2
PeterPaul.chem.nutrients$sampledate <- as.Date(PeterPaul.chem.nutrients$sampledate, format = "%Y-%m-%d")
NiwotRidge.litter$collectDate <- as.Date(NiwotRidge.litter$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 = "purple"),
        legend.position = "right")
theme_set(mytheme)
```

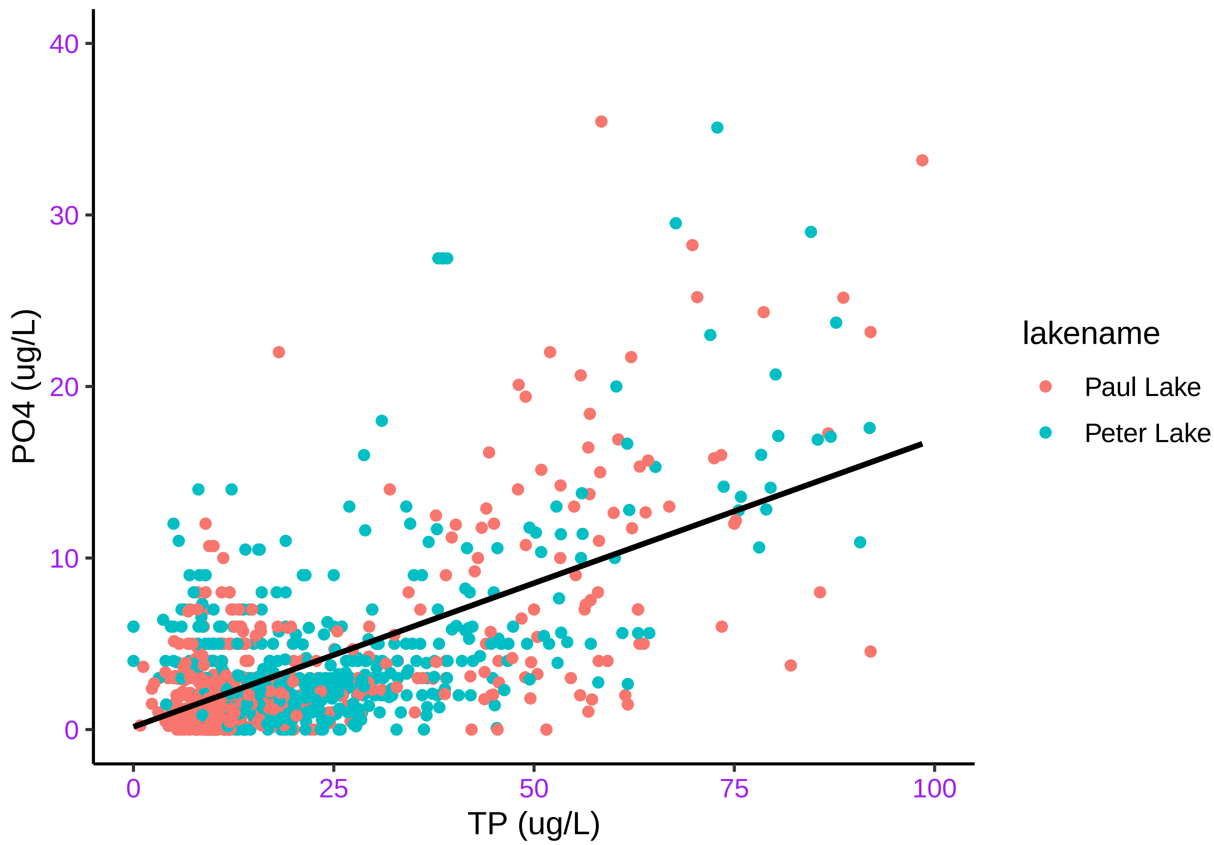
Create graphs

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()`).

```r
#4
phos <-
  ggplot(PeterPaul.chem.nutrients, aes(x = tp_ug, y = po4, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, col = "black") +
  labs(x = "TP (ug/L)", y = "PO4 (ug/L)") +
  xlim(0, 100) +
  ylim(0, 40)
print(phos)
```

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

## Warning: Removed 21965 rows containing non-finite values (stat_smooth).

## Warning: Removed 21965 rows containing missing values (geom_point).
```

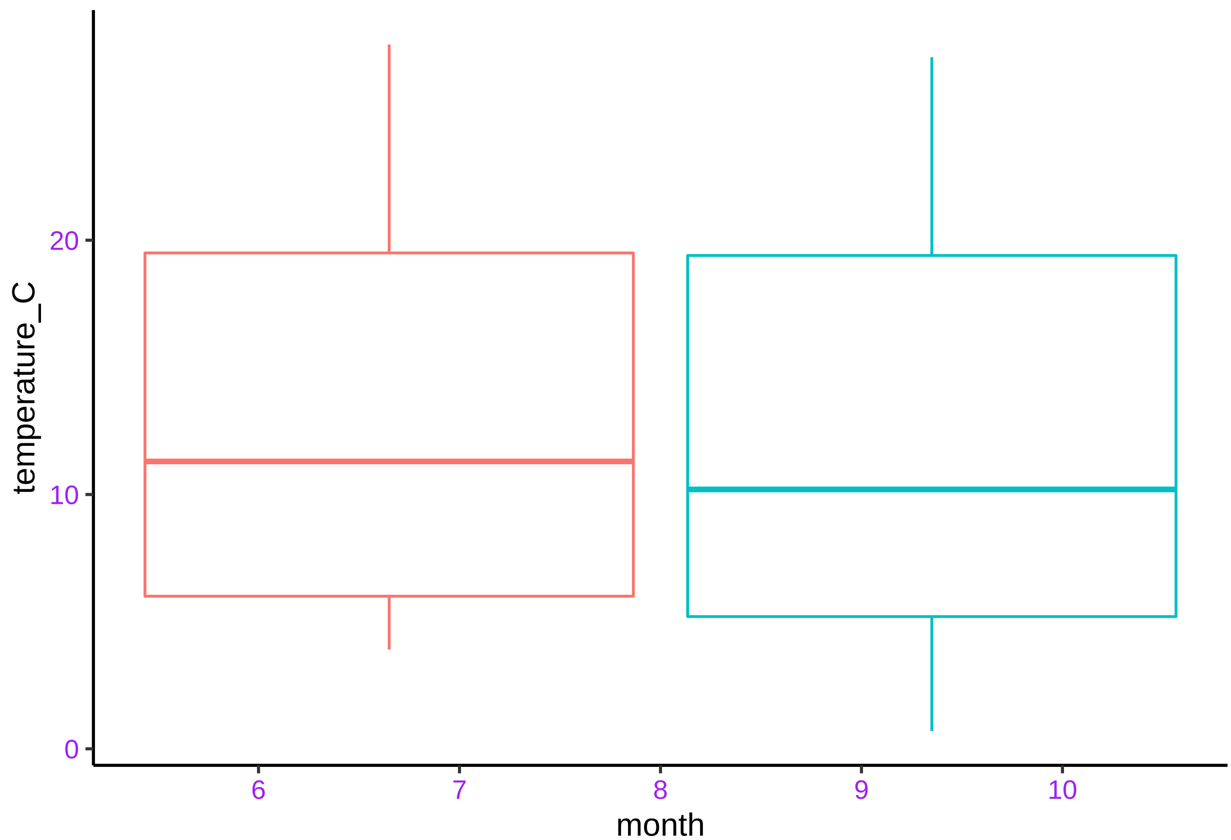


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

```r
#5
dummyplot <-
    ggplot(PeterPaul.chem.nutrients, aes(x = month, y = temperature_C, color = lakename)) +
```

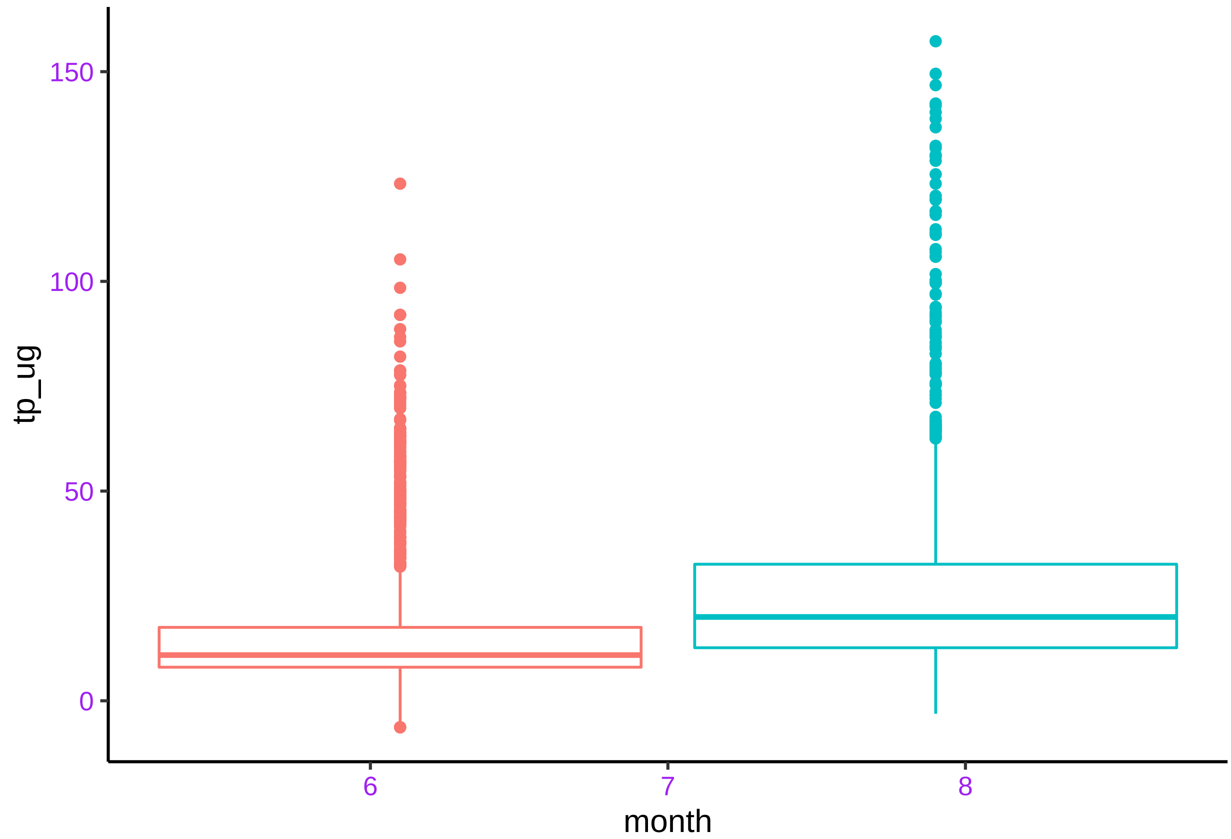
```
  geom_boxplot()

tempplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = temperature_C, color = lakename)) +
  geom_boxplot() +
  theme(legend.position = "none")
print(tempplot)
```

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



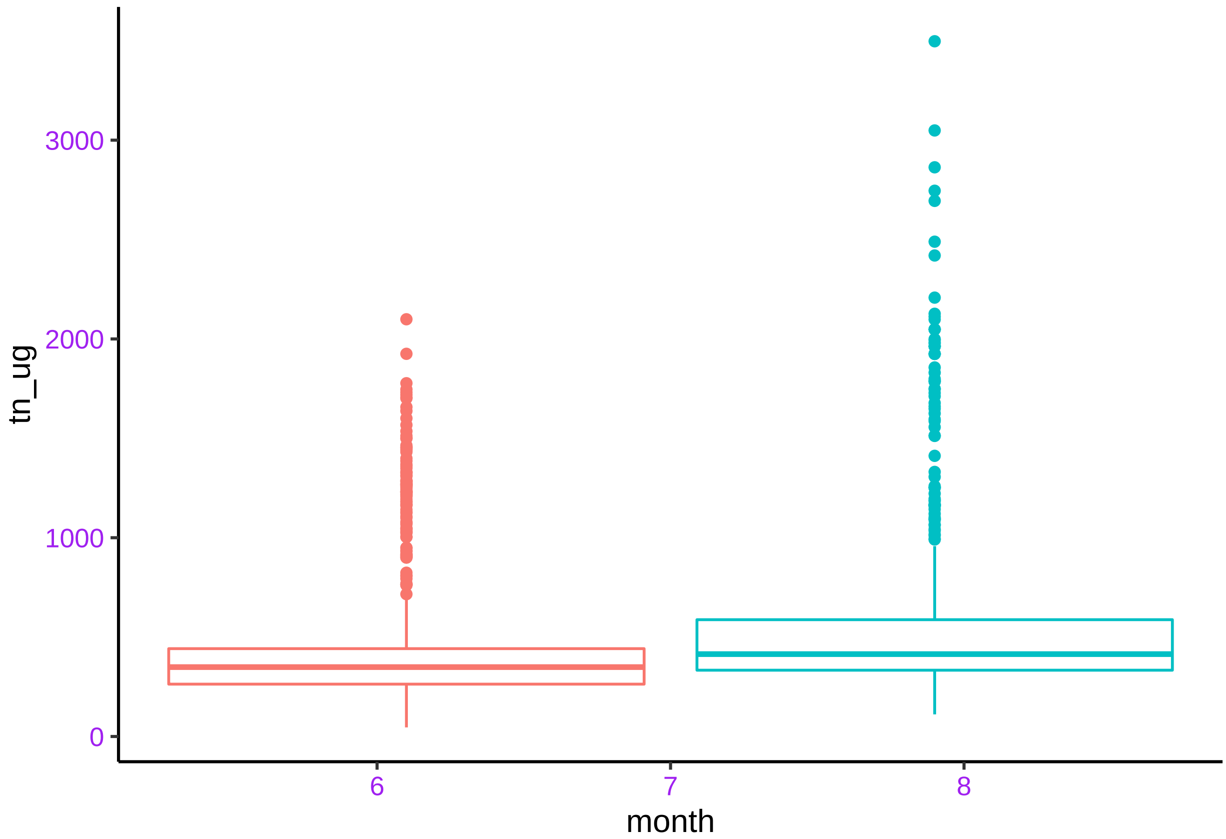

```
TPplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tp_ug, color = lakename)) +
  geom_boxplot()+
  theme(legend.position = "none")
print(TPplot)
```

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




```r
TNplot <-
  ggplot(PeterPaul.chem.nutrients, aes(x = month, y = tn_ug, color = lakename)) +
  geom_boxplot() +
  theme(legend.position = "none")
print(TNplot)
```

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




```
legend <- get_legend(dummyplot)
```

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

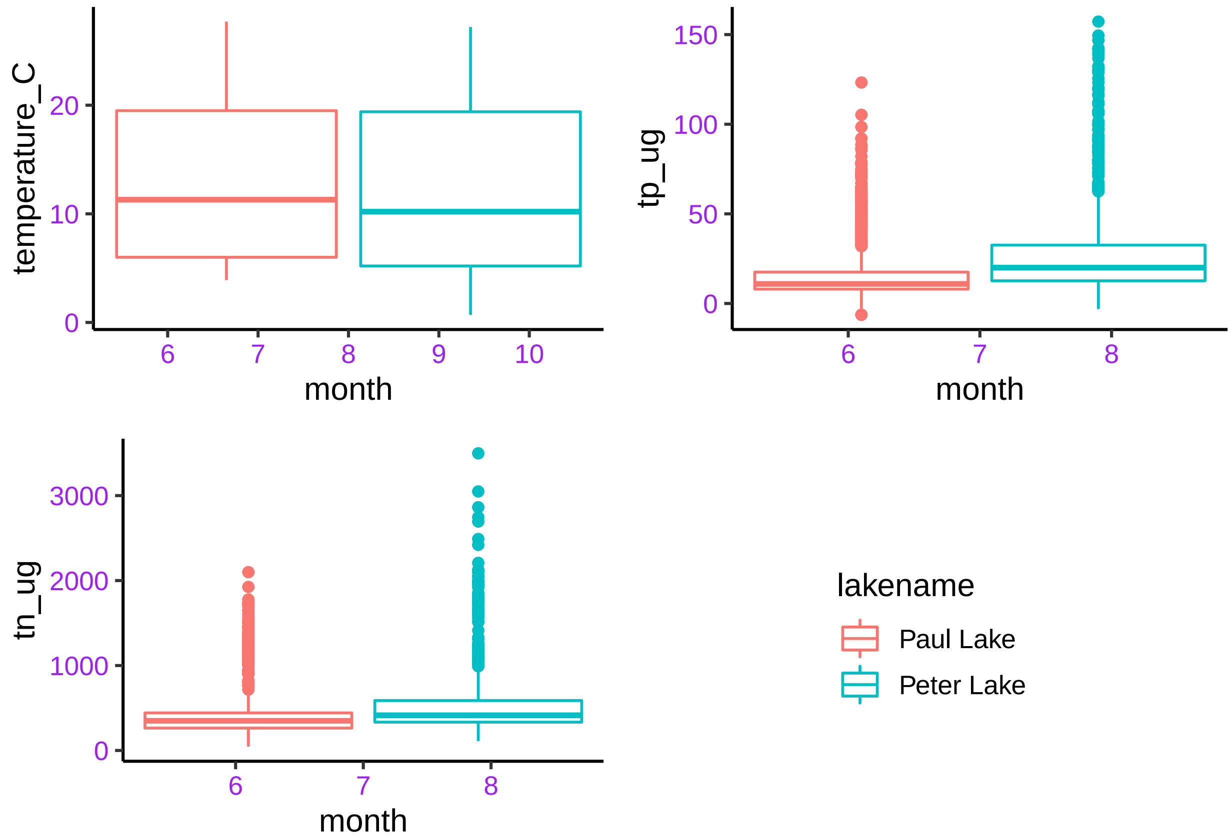
```
plot_grid(tempplot, TPplot, TNplot, legend, nrow = 2, align = 'h')
```

```
## 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).
```

```
## Warning: Graphs cannot be horizontally aligned unless the axis parameter is set.
## Placing graphs unaligned.
```

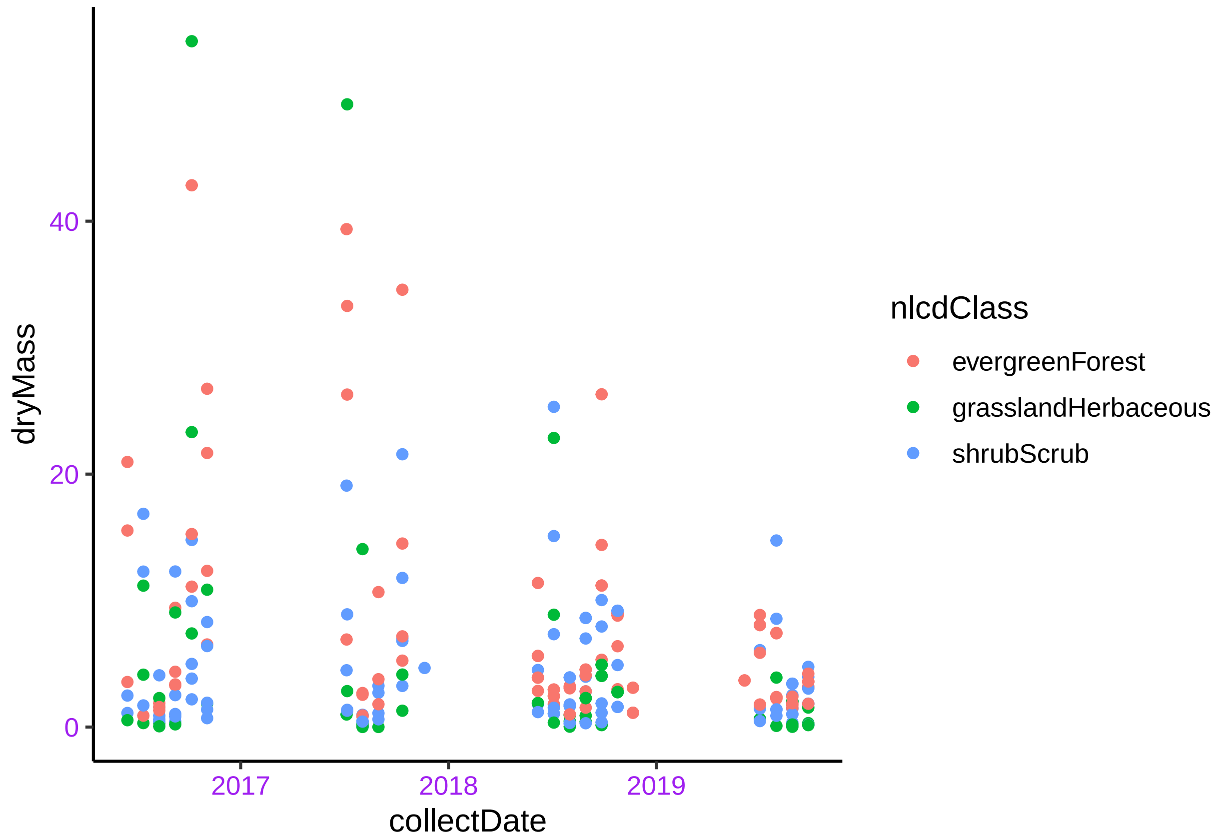


Question: What do you observe about the variables of interest over seasons and between lakes?

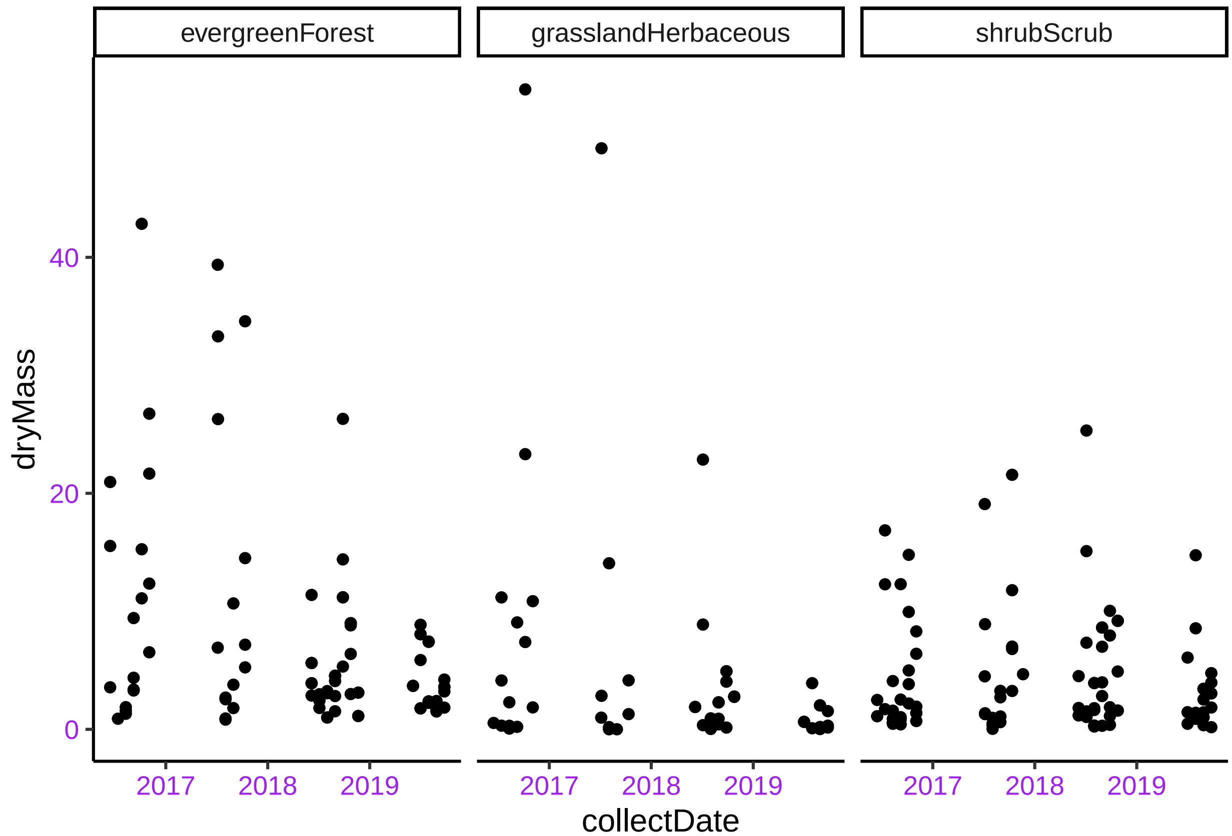
> Answer: The mean temperature of Paul Lake is slightly warmer than Peter Lake, but they're comparable in terms of their distributions. Peter Lake also has more nutrient loads with levels slightly above that of Paul Lake. From these boxplots, it is hard to gauge the variability over seasons and between lakes for each variable because, for example, it would seem that there are no measurements for Paul Lake past July.

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
NWT.litter <-
  ggplot(subset(NiwotRidge.litter, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point()
print(NWT.litter)
```




```
#7
NWT.litter <-
  ggplot(subset(NiwotRidge.litter, functionalGroup == "Needles"),
         aes(x = collectDate, y = dryMass)) +
  geom_point() +
  facet_wrap(vars(nlcdClass))
print(NWT.litter)
```



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

> Answer: I think the second plot is more effective because there is too much going on in the first one for me to discern trends among the classes. When they are faceted, it is much easier to see what data belongs to which class.