

Assignment 5: Data Visualization

Jess Ozog

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_NIW0_Litter_mass_trap_Processed.csv] version).
2. Make sure R is reading dates as date format; if not change the format to date.

```
#1
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=60), tidy=TRUE, echo=TRUE)

getwd()
```

```
## [1] "C:/ENV872/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   1.1.4      v stringr 1.4.0
## v readr   2.0.2      v forcats 0.5.1
```

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

library(cowplot)
lakeChemistry<-read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
                        stringsAsFactors = TRUE)
litter <- read.csv("./Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
                  stringsAsFactors = TRUE)

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

Define your theme

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

```
# 3
theme1 <- theme_bw(base_size = 12) + theme(panel.border = element_rect(fill = "transparent",
                                color = "gray", size = 1), plot.title = element_text(hjust = 0.5),
                                legend.position = "right")
theme_set(theme1)
```

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()).

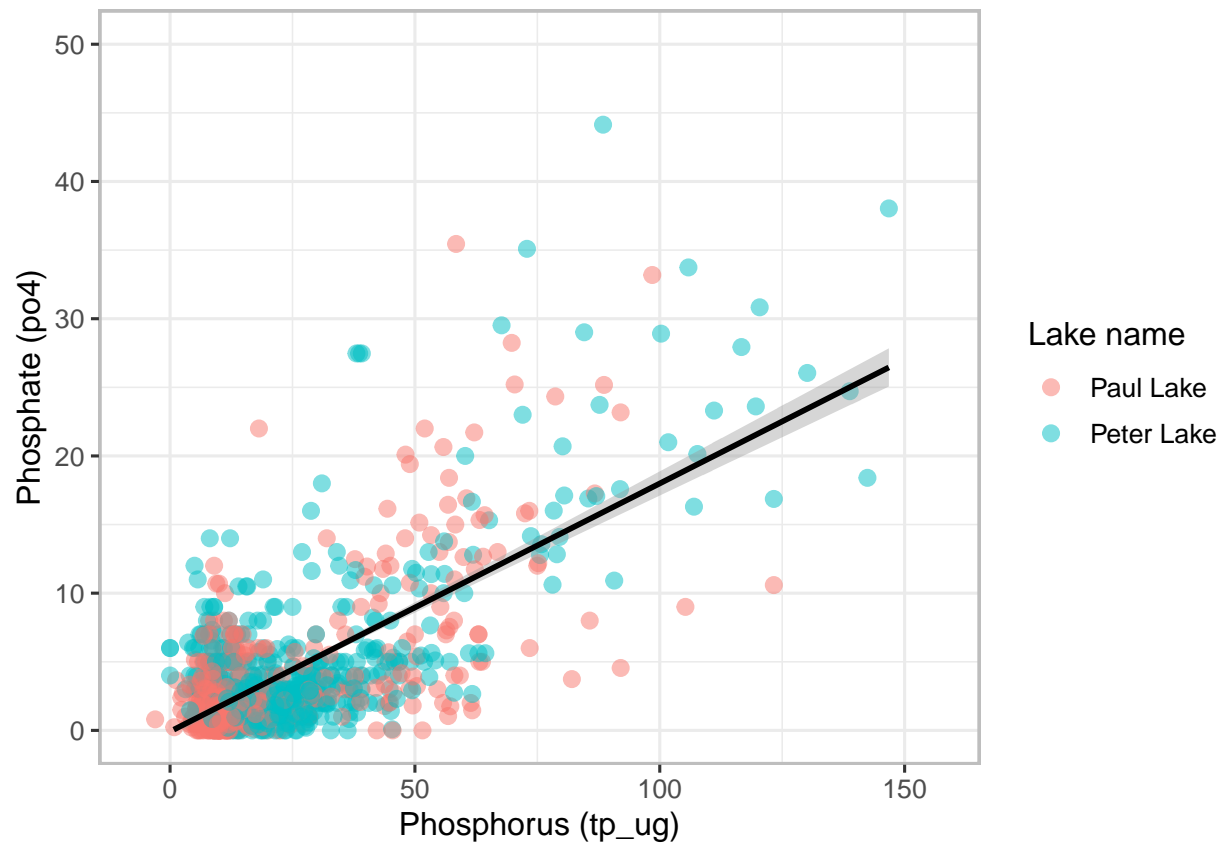
```
# 4
plot1 <- ggplot(lakeChemistry, aes(x = tp_ug, y = po4, color = lakename)) +
  geom_point(size = 2.5, alpha = 0.5) + ylim(0, 50) + xlab("Phosphorus (tp_ug)") +
  ylab("Phosphate (po4)") + labs(color = "Lake name") + geom_smooth(method = lm,
                                color = "black")
print(plot1)
```

```
## '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).
```

```
## Warning: Removed 2 rows containing missing values (geom_smooth).
```

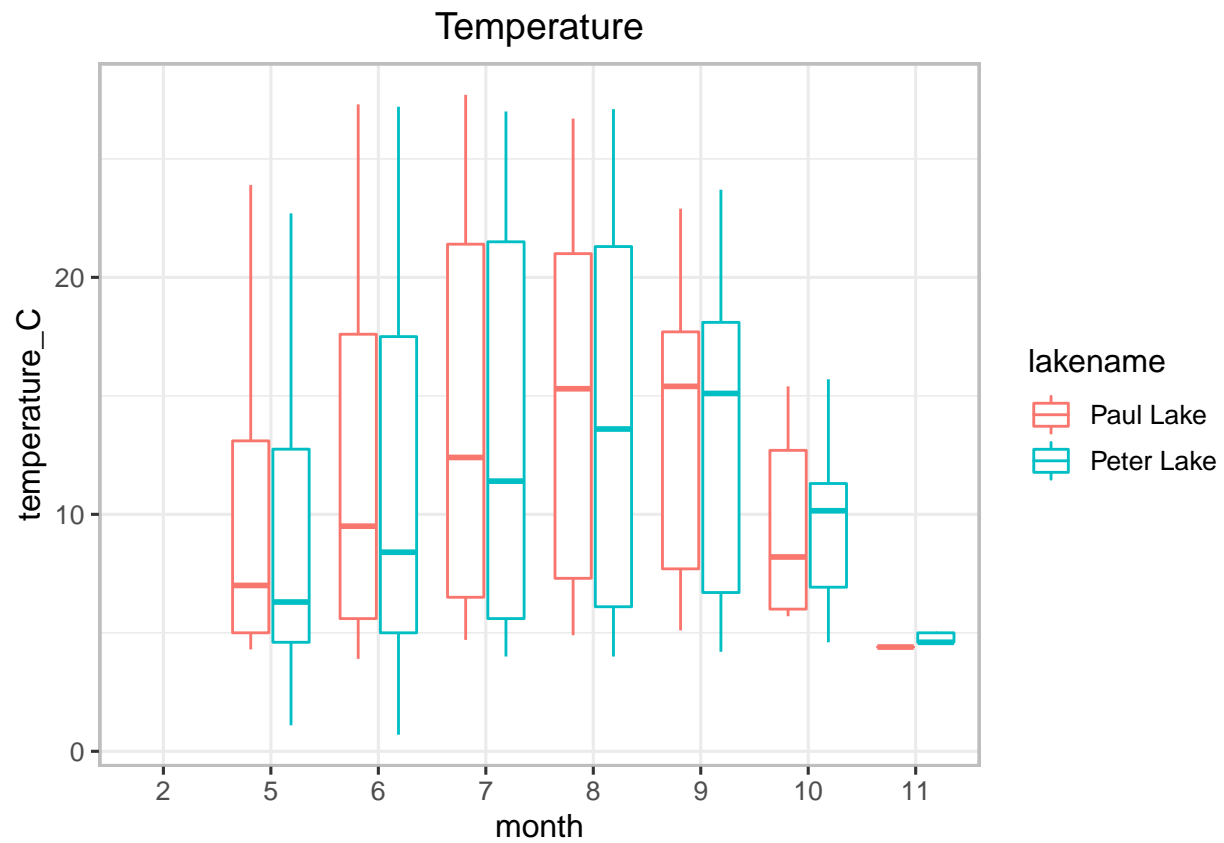


5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and (c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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

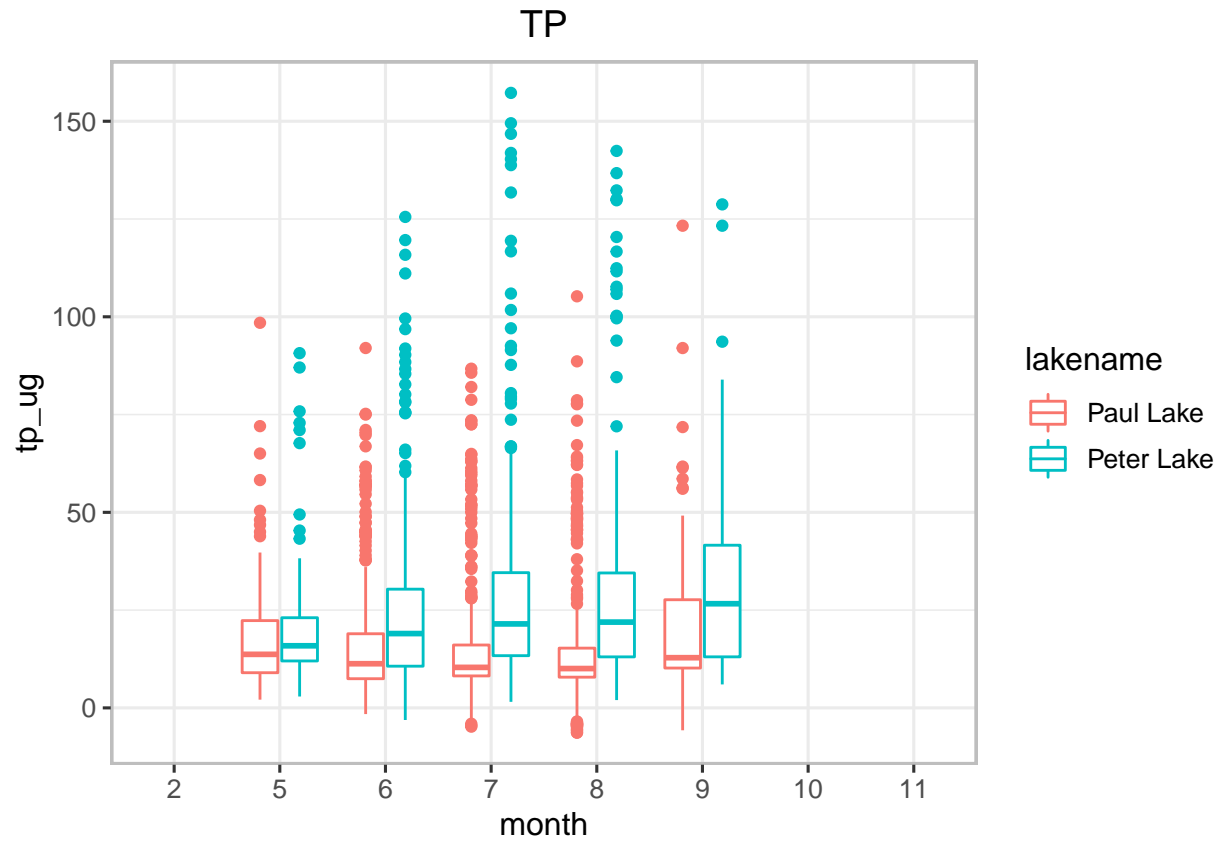
# temp month plot
temp <- ggplot(lakeChemistry) + geom_boxplot(aes(x = month, y = temperature_C,
  color = lakename)) + ggtitle("Temperature")
print(temp)
```

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



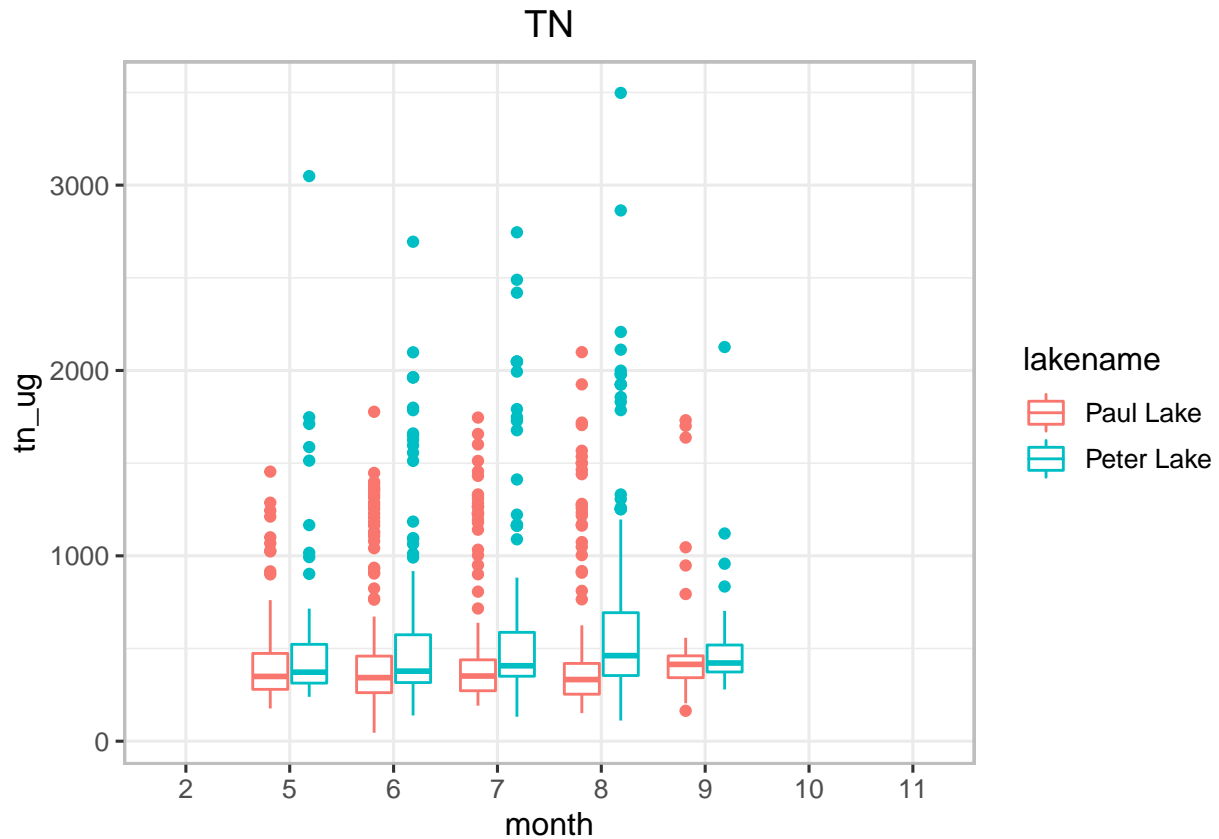
```
# tp month plot
tp <- ggplot(lakeChemistry) + geom_boxplot(aes(x = month, y = tp_ug,
  color = lakename)) + ggtitle("TP")
print(tp)
```

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



```
# tn month plot
tn <- ggplot(lakeChemistry) + geom_boxplot(aes(x = month, y = tn_ug,
  color = lakename)) + ggtitle("TN")
print(tn)
```

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



```
# put all plots into one row
plots <- plot_grid(temp + theme(legend.position = "none"), tp +
  theme(legend.position = "none"), tn + theme(legend.position = "none"),
  nrow = 1)
```

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

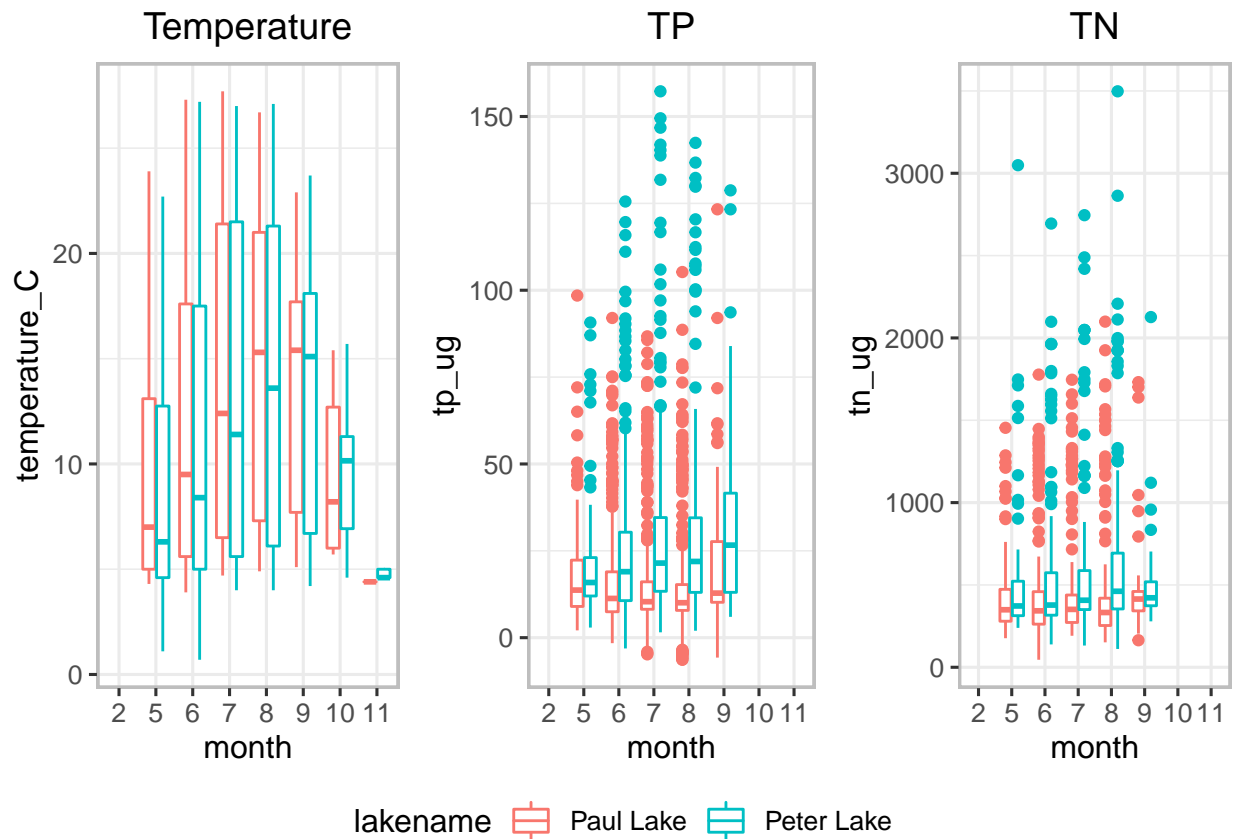
```
## Warning: Removed 20729 rows containing non-finite values (stat_boxplot).
```

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

```
# extract out legend from one of the plots
legend <- get_legend(temp + theme(legend.position = "bottom"))
```

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

```
# plot all the plots with one legend
plot_grid(plots, legend, ncol = 1, rel_heights = c(1, 0.1))
```



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

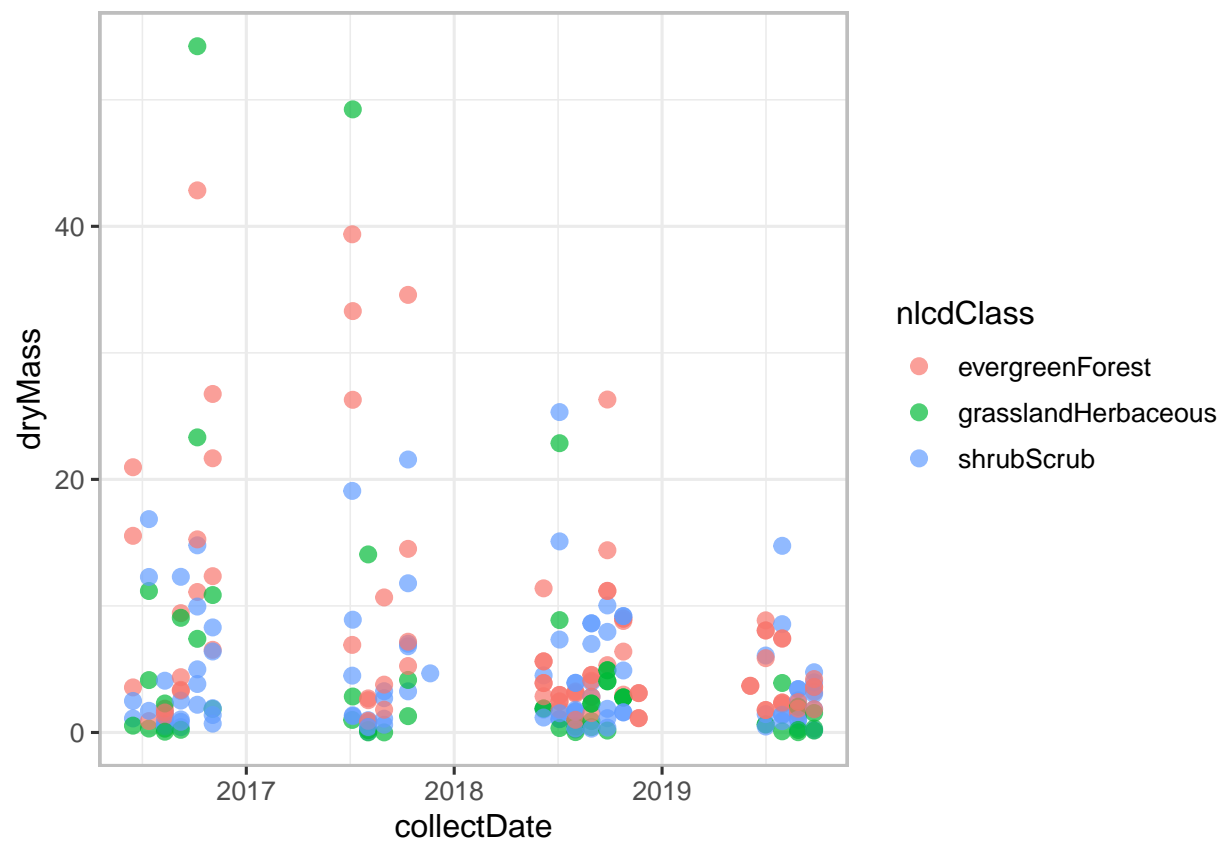
Answer: Temperature was consistently similar throughout all the sampled months for both Peter Lake and Paul Lake. TP for Peter Lake had a larger interquartile range compared to Paul Lake, as well as more extreme outliers. TN had similar results as for TP, with Peter Lake having slightly larger IQRs and more extreme outliers.

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.

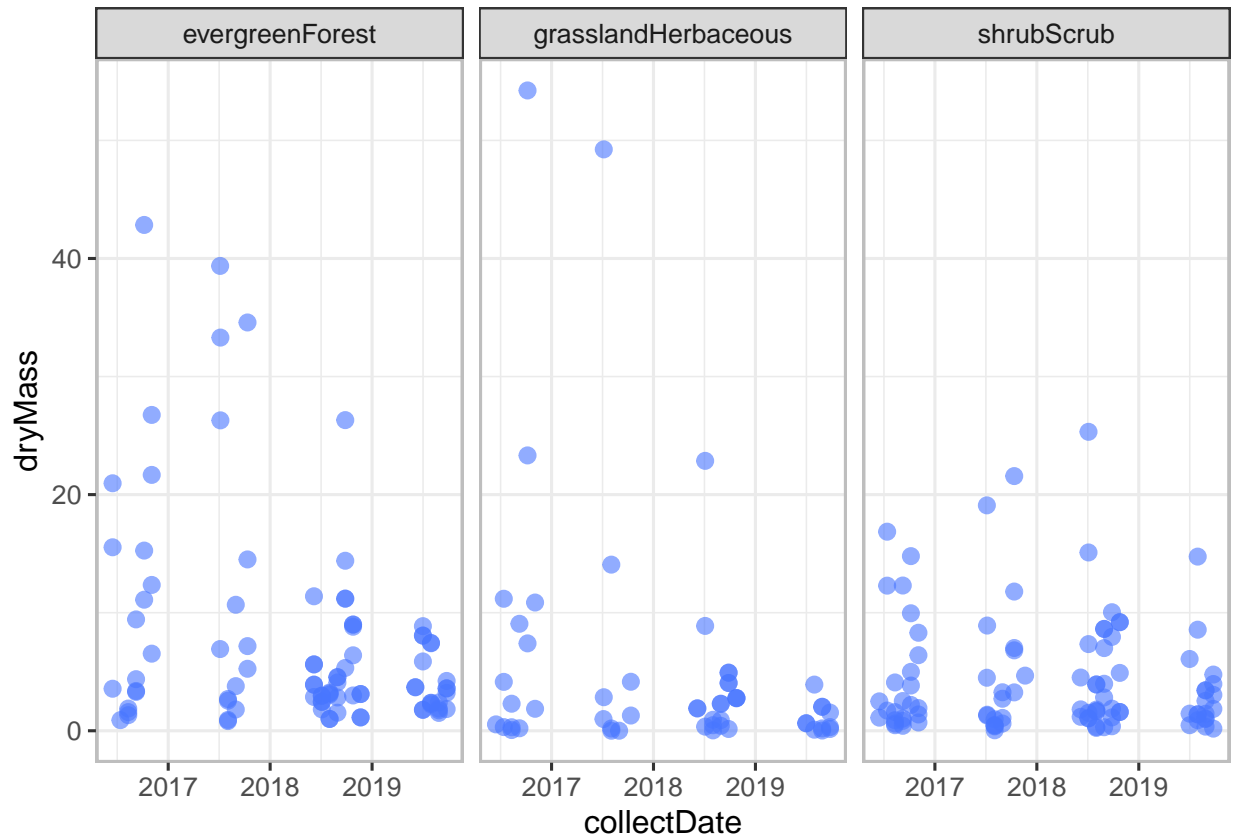
```
# 6

# get just data with needles functional group
needles <- filter(litter, functionalGroup == "Needles")

plot2 <- ggplot(needles, aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point(size = 2.5, alpha = 0.7)
print(plot2)
```



```
# 7
plot3 <- ggplot(needles, aes(x = collectDate, y = dryMass)) +
  geom_point(size = 2.5, alpha = 0.6, color = "royal blue1") +
  facet_wrap(vars(nlcdClass), nrow = 1)
print(plot3)
```

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

Answer: I think plot 7 is more effective at showing how the dry mass of needle litter changes over time. Plot 7 makes it easier to see dry mass for each NLCD class specifically, rather than all on top of each other in graph 6. The points are also mostly concentrated around the same dry mass values for each NLCD class, therefore, plot 7 allows for easier visualization.