Assignment 5: Data Visualization

Molly Bruce

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 Tuesday, February 23 at 11:59 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 (both the tidy [`NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv`] and the gathered [`NTL-LTER_Lake_Nutrients_PeterPaulGathered_Processed.csv`] versions) and the processed data file for the Niwot Ridge litter dataset.

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

```
#1

getwd()
```

```
## [1] "C:/Users/mmb88/Desktop/Environmental_Data_Analytics_2021/Assignments"
```

```
#install.packages("tidyverse")
#install.packages("ggplot2")
#install.packages("cowplot")

library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.3
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
```

```
## Warning: package 'tibble' was built under R version 4.0.3
```

```
## Warning: package 'tidyr' was built under R version 4.0.3
## Warning: package 'readr' was built under R version 4.0.3
## Warning: package 'purrr' was built under R version 4.0.3
## Warning: package 'dplyr' was built under R version 4.0.3
## Warning: package 'stringr' was built under R version 4.0.3
## Warning: package 'forcats' was built under R version 4.0.3
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.0.3
```

```
library(scales)
```

```
## Warning: package 'scales' was built under R version 4.0.3
##
## Attaching package: 'scales'
## The following object is masked from 'package:purrr':
##
##     discard
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
# Again, I'm providing full paths because
# RStudio on my virtual machine won't process the relative paths
# I recognize that relative paths are preferable

NTL_LTER_nutrients_processed <-
  read.csv("C:/Users/mmb88/Desktop/Environmental_Data_Analytics_2021/Data/Processed/NTL-LTER_Lake_Chemis

NTL_LTER_gathered_processed <-
  read.csv("C:/Users/mmb88/Desktop/Environmental_Data_Analytics_2021/Data/Processed/NTL-LTER_Lake_Nutri

NiwotRidge_Litter <-
  read.csv("C:/Users/mmb88/Desktop/Environmental_Data_Analytics_2021/Data/Processed/NEON_NIWO_Litter_mas

#2

# Verified after in Environment tab that sampledate column is read as factor
NTL_LTER_nutrients_processed$sampledate <-
  as.Date(NTL_LTER_nutrients_processed$sampledate, format = "%Y-%m-%d")
NTL_LTER_gathered_processed$sampledate <-
  as.Date(NTL_LTER_gathered_processed$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.

```
# I could add more to this theme but will leave it more simple for now
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top",
        panel.grid.major = element_line(colour = "gray"))
theme_set(mytheme)
```

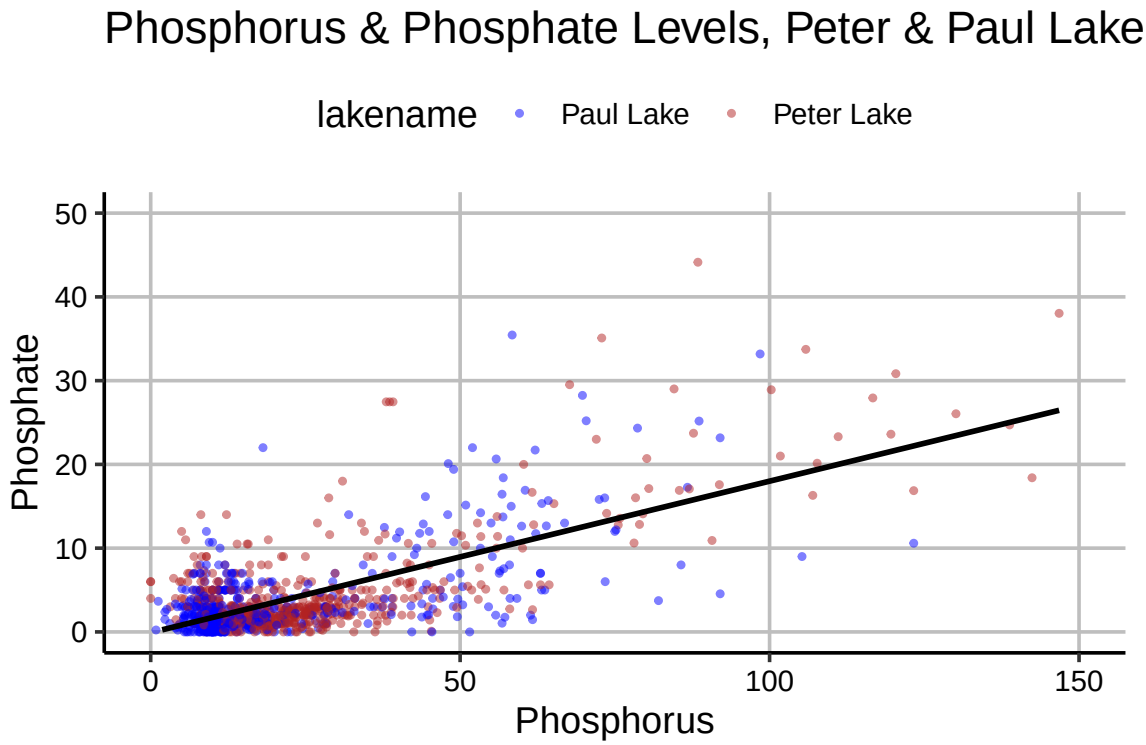
Create graphs

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.

```
NTL_LTER_Scatter <- ggplot(NTL_LTER_nutrients_processed,
                           aes(x = tp_ug,
                               y = po4,
                               color = lakename)) +
  geom_point(size = .9, alpha = .5) +
  scale_color_manual(values = c("blue", "firebrick")) +
  xlim(0, 150) + ylim(0, 50) +
  geom_smooth(method = lm, se = FALSE, color = "black") +
  labs(title = "Phosphorus & Phosphate Levels, Peter & Paul Lakes",
       y = "Phosphate",
       x = "Phosphorus") +
  mytheme
print(NTL_LTER_Scatter)
```

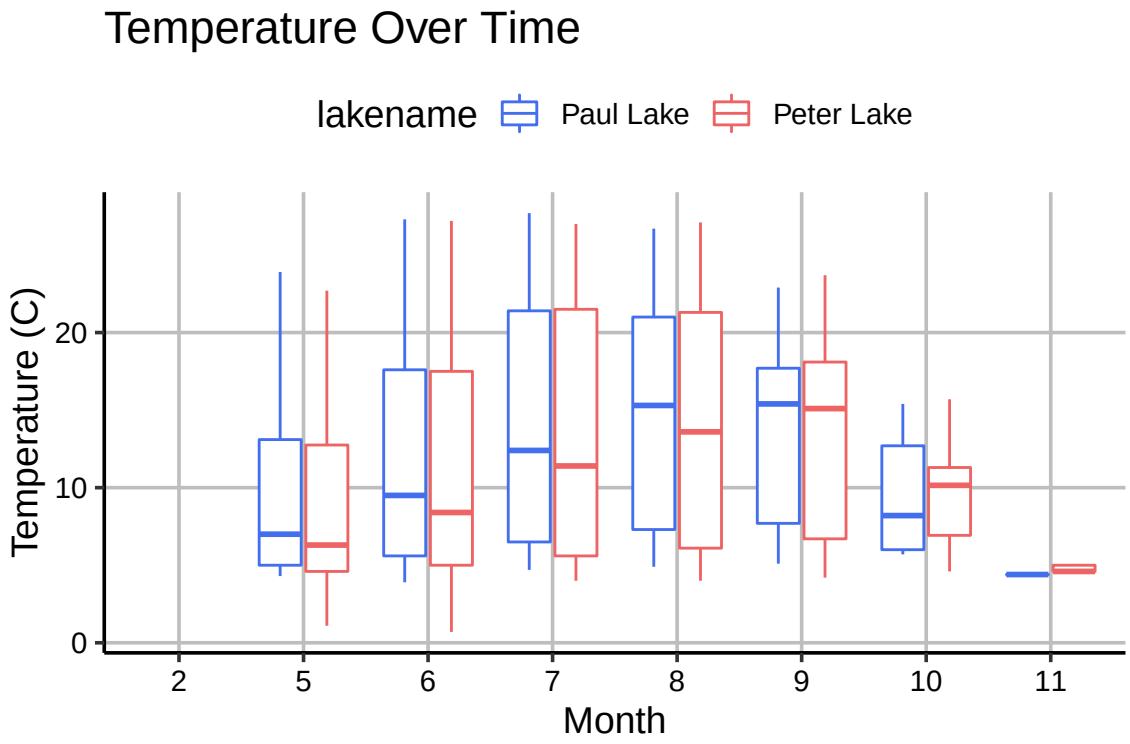
```
## `geom_smooth()` using formula 'y ~ x'

## 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.

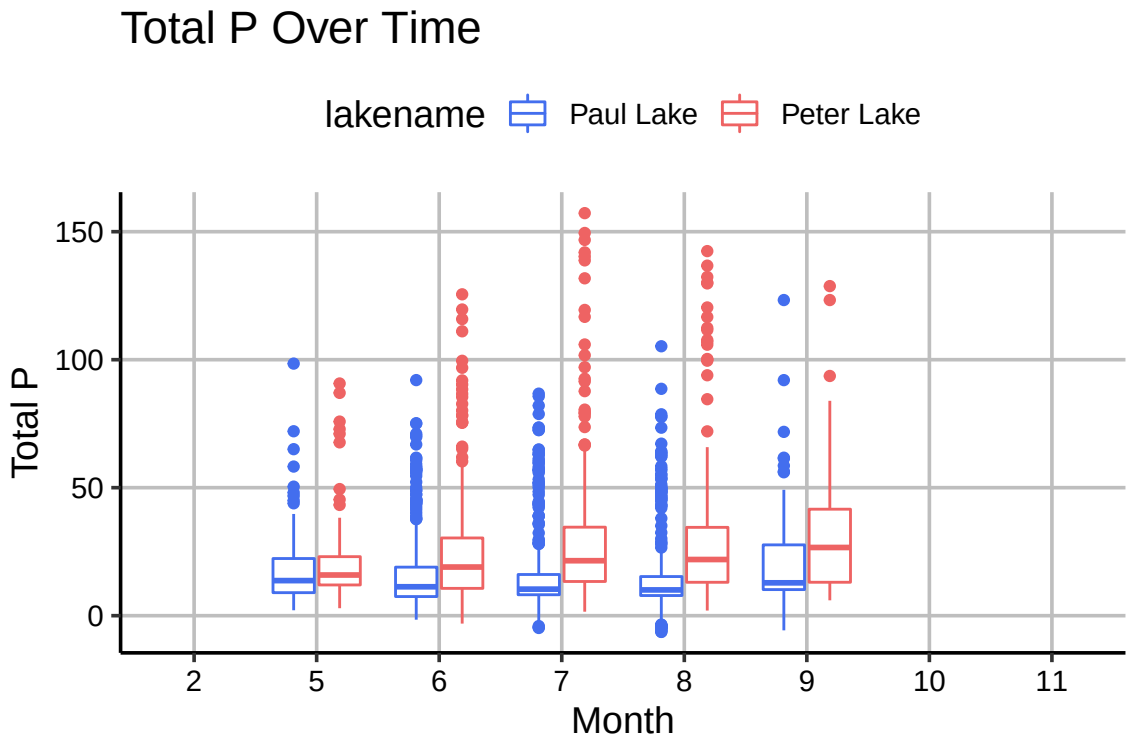
```
# (a) Temperature Boxplot
NTL_LTER_Box_Temp <- ggplot(NTL_LTER_nutrients_processed,
                            aes(x = as.factor(month),
                                y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  labs(title = "Temperature Over Time",
       y = "Temperature (C)",
       x = "Month")
print(NTL_LTER_Box_Temp)
```

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



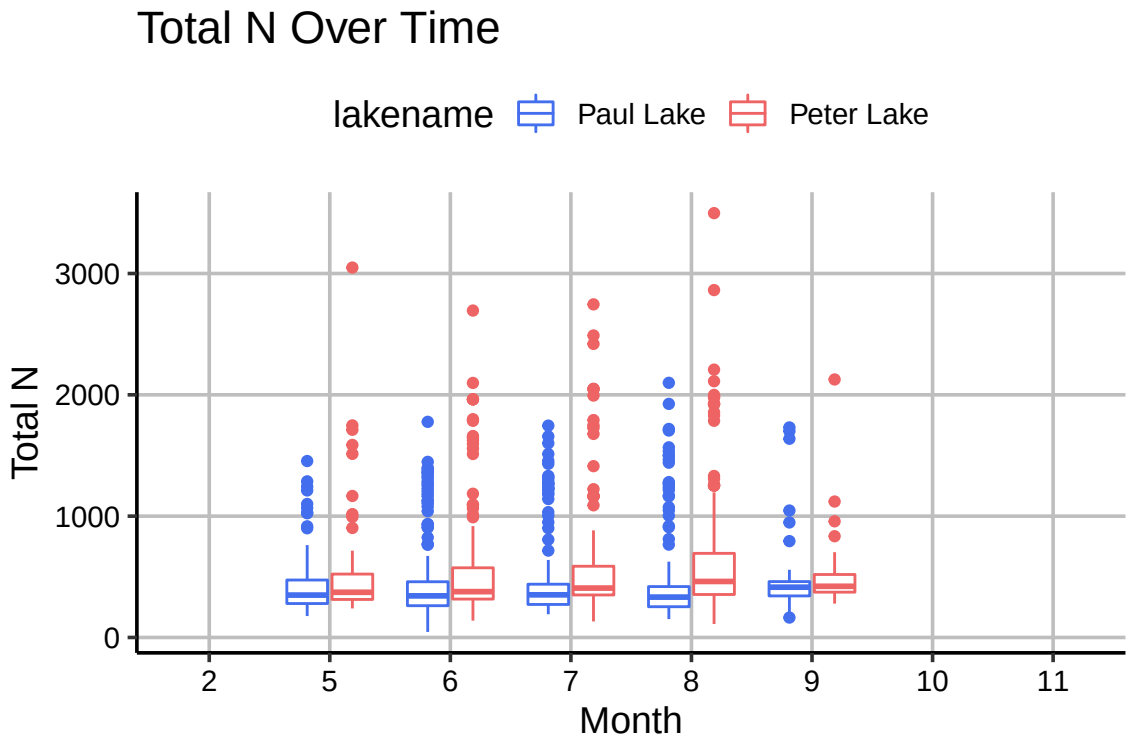

```
# (b) TP Boxplot
NTL_LTER_Box_TP <- ggplot(NTL_LTER_nutrients_processed,
                          aes(x = as.factor(month),
                              y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  labs(title = "Total P Over Time",
       y = "Total P",
       x = "Month")
print(NTL_LTER_Box_TP)
```

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




```
# (c) TN Boxplot
NTL_LTER_Box_TN <- ggplot(NTL_LTER_nutrients_processed,
                          aes(x = as.factor(month),
                              y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  labs(title = "Total N Over Time",
       y = "Total N",
       x = "Month")
print(NTL_LTER_Box_TN)
```

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



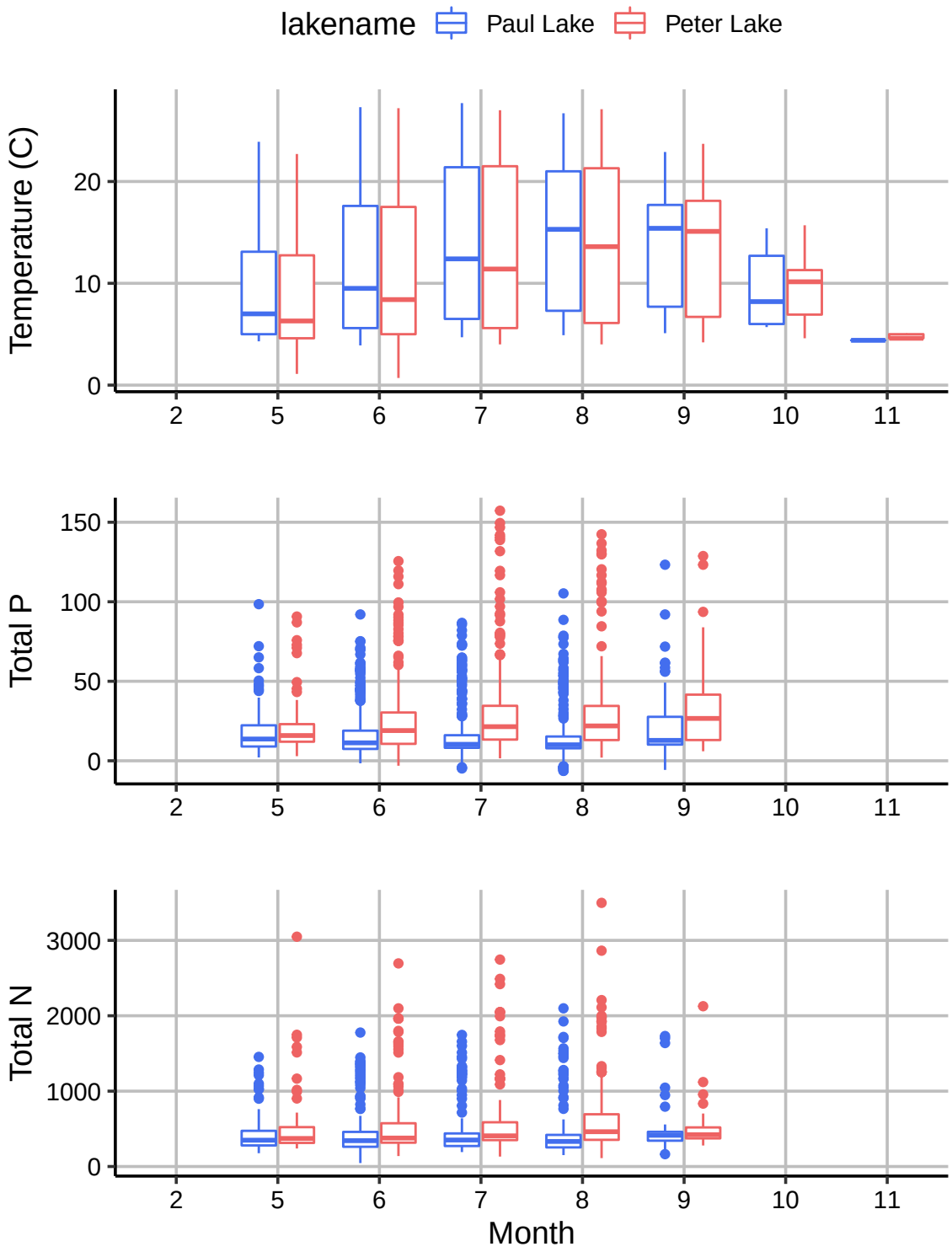

```
# Temp boxplot without x label (for cowplot cleaning)
NTL_LTER_Box_Temp <- ggplot(NTL_LTER_nutrients_processed,
                            aes(x = as.factor(month),
                                y = temperature_C)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  ylab("Temperature (C)") +
  xlab(" ") # adding this for part below

# TP Boxplot without legend (for cowplot cleaning)
NTL_LTER_Box_TP_NL <- ggplot(NTL_LTER_nutrients_processed,
                            aes(x = as.factor(month),
                                y = tp_ug)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  theme(legend.position = "none") + # adding this for part below
  ylab("Total P") +
  xlab(" ") # adding this for part below

# TN Boxplot without legend (for cowplot cleaning)
NTL_LTER_Box_TN_NL <- ggplot(NTL_LTER_nutrients_processed,
                            aes(x = as.factor(month),
                                y = tn_ug)) +
  geom_boxplot(aes(color = lakename)) +
  scale_color_manual(values = c("royalblue2", "indianred2")) +
  theme(legend.position = "none") + # adding this for part below
  ylab("Total N") +
  xlab("Month")
```

```r
# cowplot combining TN, TP, and Temp & only 1 legend
plot_grid(NTL_LTER_Box_Temp, NTL_LTER_Box_TP_NL, NTL_LTER_Box_TN_NL,
          nrow = 3, # 3 rows
          align = 'v', # vetically aligned
          rel_heights = c(1.3, 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).
```

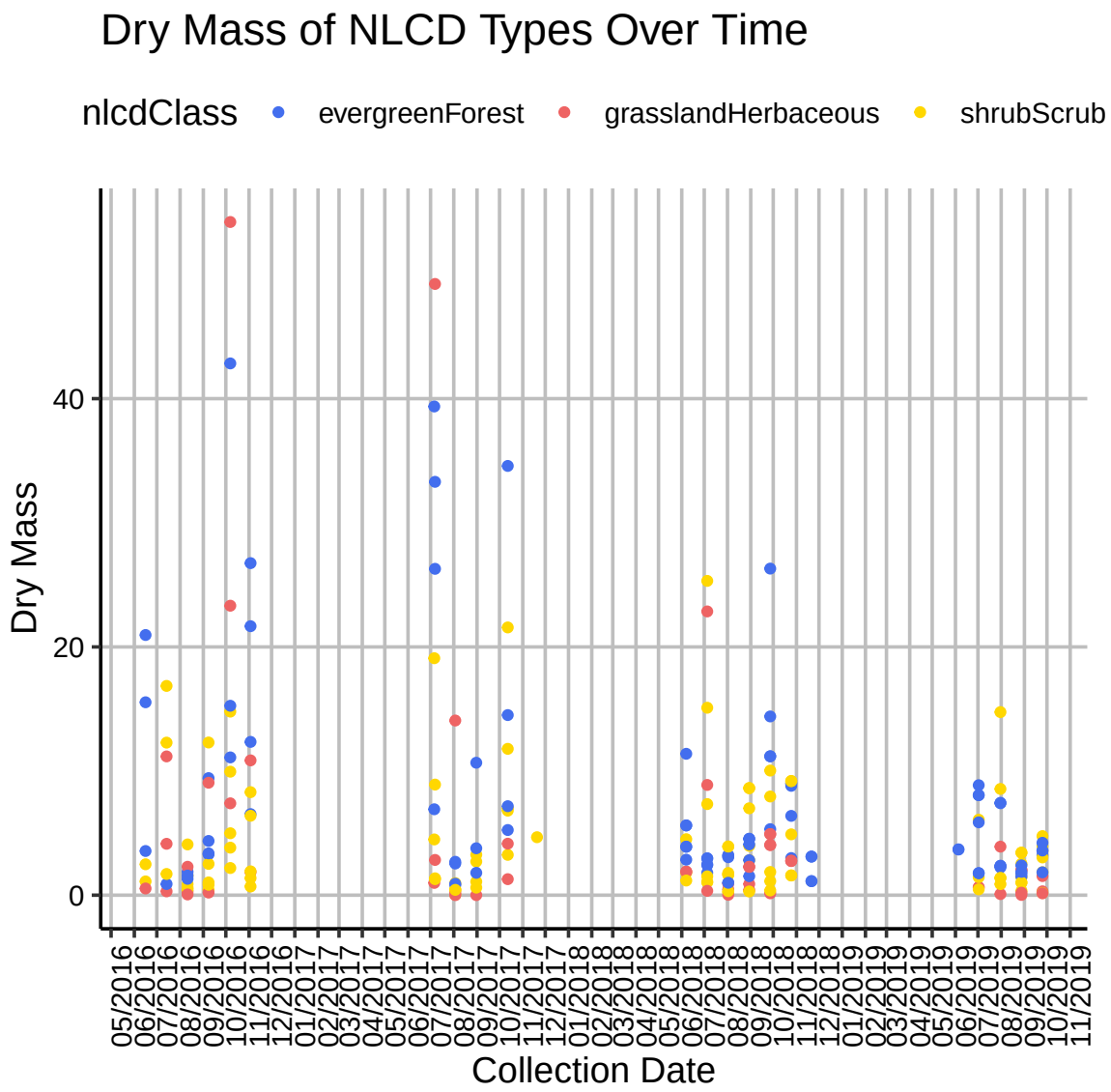


Question: What do you observe about the variables of interest over seasons and between lakes?

Answer: On the whole, the variables tend to be higher in the summer months and lower in the late spring & early fall. The TP and TN variables tend to be more elevated at Peter Lake than at Paul Lake, though there are less clear temperature trends between Peter and Paul Lakes.

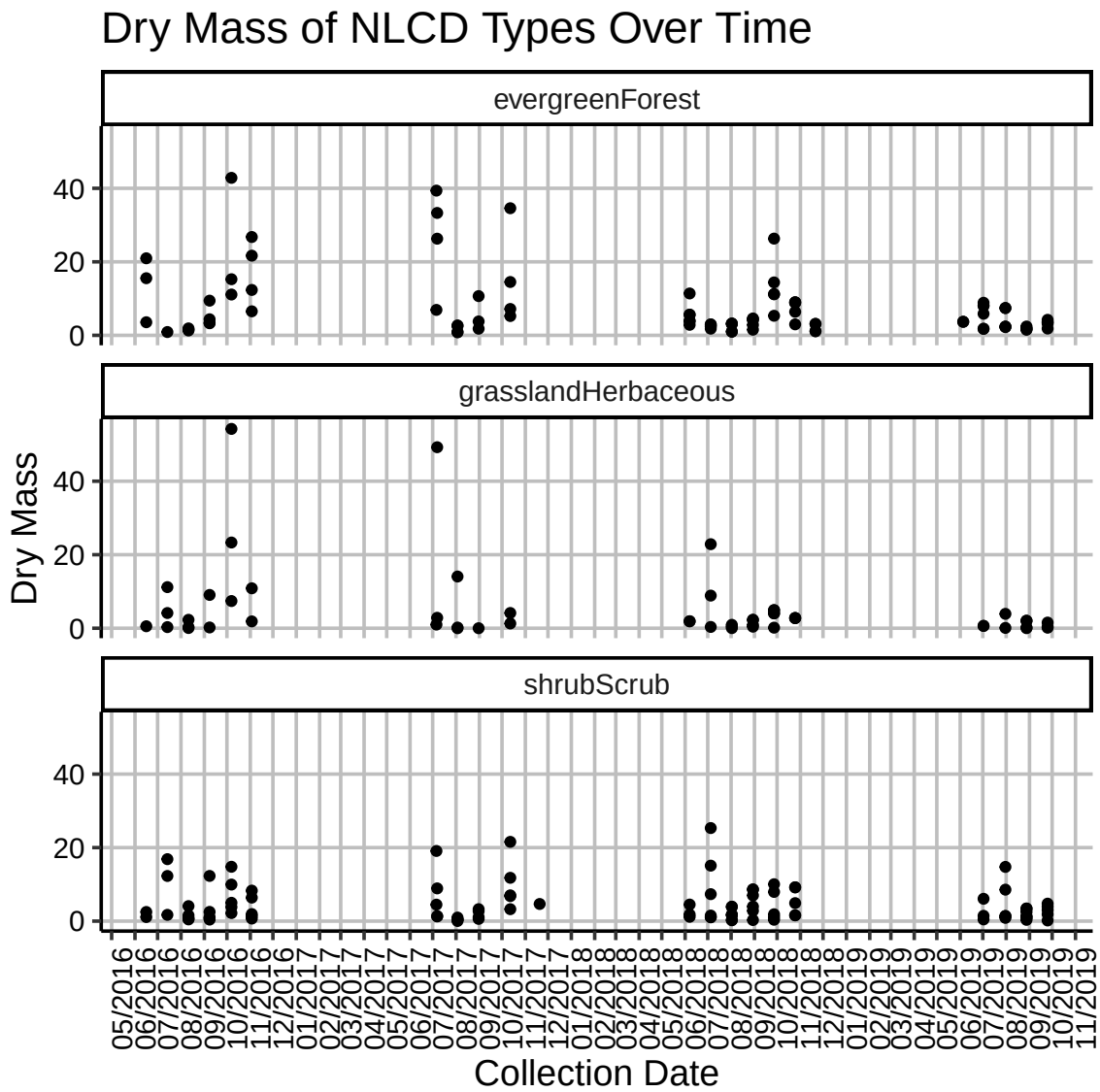
6. [Niwot Ridge] Plot a subset of the litter dataset by displaying only the "Needles" functional group. Plot the dry mass of needle litter by date and separate by NLCD class with a color aesthetic. (no need to adjust the name of each land use)

```
NiwotRidge_Litter_color <- ggplot(subset(NiwotRidge_Litter, functionalGroup %in% c("Needles")),
                                  aes(x = collectDate,
                                      y = dryMass,
                                      color = nlcdClass)) +
  geom_point() +
  scale_color_manual(values = c("royalblue2", "indianred2", "gold")) +
  labs(title = "Dry Mass of NLCD Types Over Time",
       y = "Dry Mass",
       x = "Collection Date") +
  scale_x_date(breaks = date_breaks("months"),
               labels = date_format("%m/%Y")) +
  theme(axis.text.x = element_text(angle = 90))
print(NiwotRidge_Litter_color)
```



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

```
NiwotRidge_Litter.faceted <- ggplot(subset(NiwotRidge_Litter,
                                           functionalGroup %in% c("Needles")),
                                    aes(x = collectDate,
                                        y = dryMass)) +
  geom_point() +
  labs(title = "Dry Mass of NLCD Types Over Time",
       y = "Dry Mass",
       x = "Collection Date") +
  scale_x_date(breaks = date_breaks("months"),
               labels = date_format("%m/%Y")) +
  theme(axis.text.x = element_text(angle = 90)) +
  facet_wrap(vars(nlcdClass), nrow = 3)
print(NiwotRidge_Litter.faceted)
```



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

Answer: I think facet wrap does a better job of conveying the information; it's easier to compare the dry masses, both across different NLCD Classes and also across time. For instance, it's easier to tell that the dry masses were lower for all three NLDC types in 2019. It's also easier to tell that there's a little more spread in Evergreen Forest needle dry masses than for Shrub Scrub needle dry masses. Alternatively, this information is a bit more masked/cluttered in the chart where NLCD Class is only rendered using the color aesthetic rather than through facet wrap. As a result, it's harder to discern some of these patterns in the color aesthetic chart (#6).