Assignment 5: Data Visualization

Kelly Davidson

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_PeterPau` version) and the processed data file for the Niwot Ridge litter dataset (use the [`NEON_NIWO_Litter_mass_trap_Processed` version).

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

```
# 1. Verifying working directory and loading tidyverse, lubridate, & cowplot
# packages Loading the two processed data files 'Lake_nutrients_processed' and
# 'Litter_processed'
getwd()
```

```
## [1] "/home/guest/EDA-Fall2022/EDA-Fall2022"
```

```
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   1.2.0      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
```

```
Lake_nutrients_processed <- read.csv("./Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Proce
    stringsAsFactors = TRUE)

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

# 2. Checking the class of the date columns and changing them to the
# appropriate date format for each dataset
class(Lake_nutrients_processed$sampledate)
```

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

```
class(Litter_processed$collectDate)
```

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

```
Lake_nutrients_processed$sampledate <- as.Date(Lake_nutrients_processed$sampledate,
    format = "%Y-%m-%d")
Litter_processed$collectDate <- as.Date(Litter_processed$collectDate, format = "%Y-%m-%d")
```

Define your theme

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

```
# 3. Building my theme and setting it as the default theme
A05_theme <- theme_classic(base_size = 14) + theme(axis.text = element_text(color = "dark gray"),
    legend.position = "top")

theme_set(A05_theme)
```

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()`).

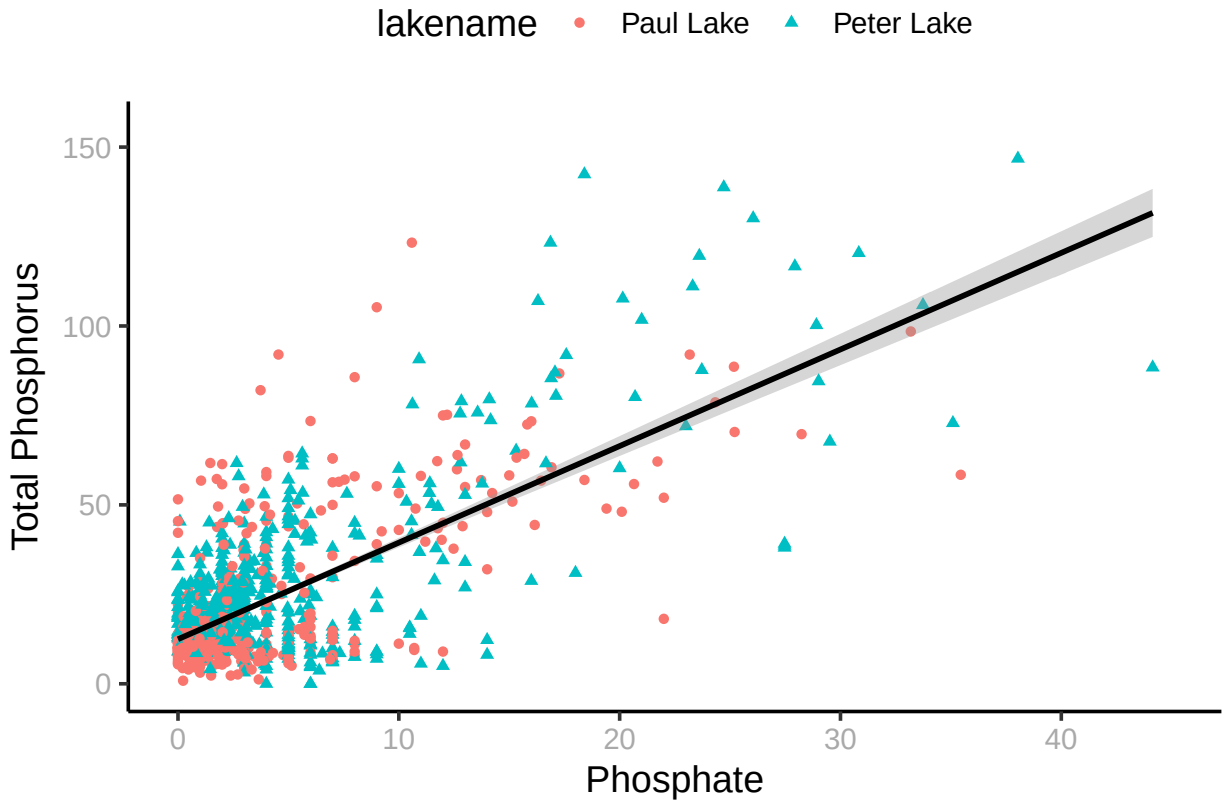
```
#4. Plotting total phosphorus by phosphate concentration
phosphate_plot <-
  ggplot(Lake_nutrients_processed, aes(x = po4, y = tp_ug)) +
  ylab(expression("Total Phosphorus")) +    #renaming y axis
  xlab(expression("Phosphate")) +    #renaming x axis
  xlim(0, 45) +    #adjusting x-axis to hid extreme values above 45
```

```
  ylim(0, 155) +    #adjusting y-axis to hide extreme values above 155
  geom_point(aes(shape = lakename, color = lakename)) +   #creating separate
                                  #aesthetics for Peter & Paul Lakes
  geom_smooth(method = lm, color = "black")    #adding line of best fit & coloring it black
print(phosphate_plot)
```

```
## `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).
```



5. [NTL-LTER] Make three separate boxplots of (a) temperature, (b) TP, and

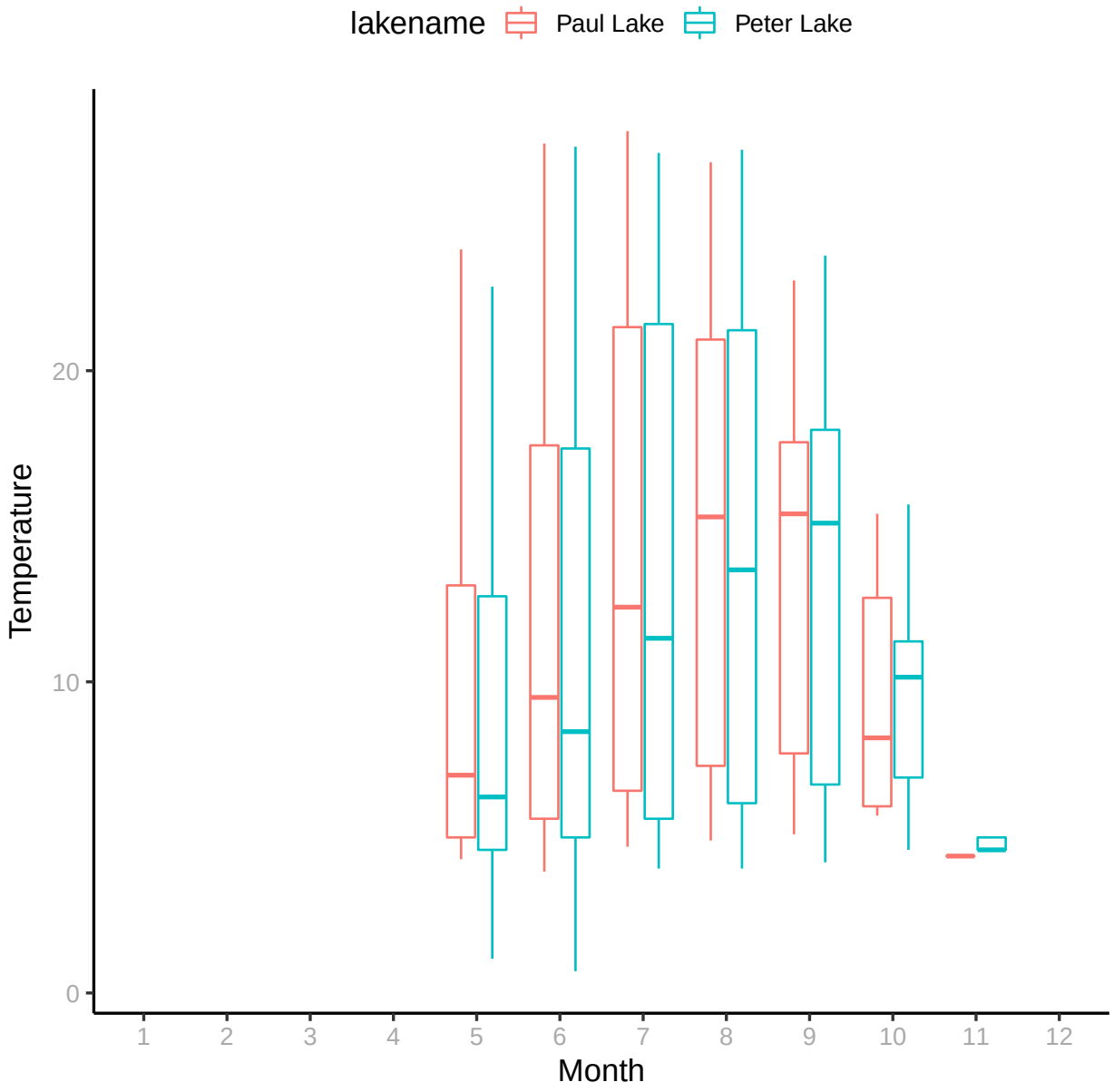
(c) TN, with month as the x axis and lake as a color aesthetic. Then, create a cowplot that combines the three graphs. Make sure that only one legend is present and that graph axes are aligned.

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

```
#5. Making 3 separate boxplots of temperature, TP, and TN with month as the x axis and
#lake as a color aesthetic
Temp_boxplot <-
  ggplot(Lake_nutrients_processed,
  aes(x = factor(month, levels = c(1:12)), y = temperature_C)) +
  xlab(expression("Month")) +    #renaming x axis
  ylab(expression("Temperature")) +    #renaming y axis
  geom_boxplot(aes(color = lakename)) +    #creating separate color aesthetics for Peter & Paul Lakes
```

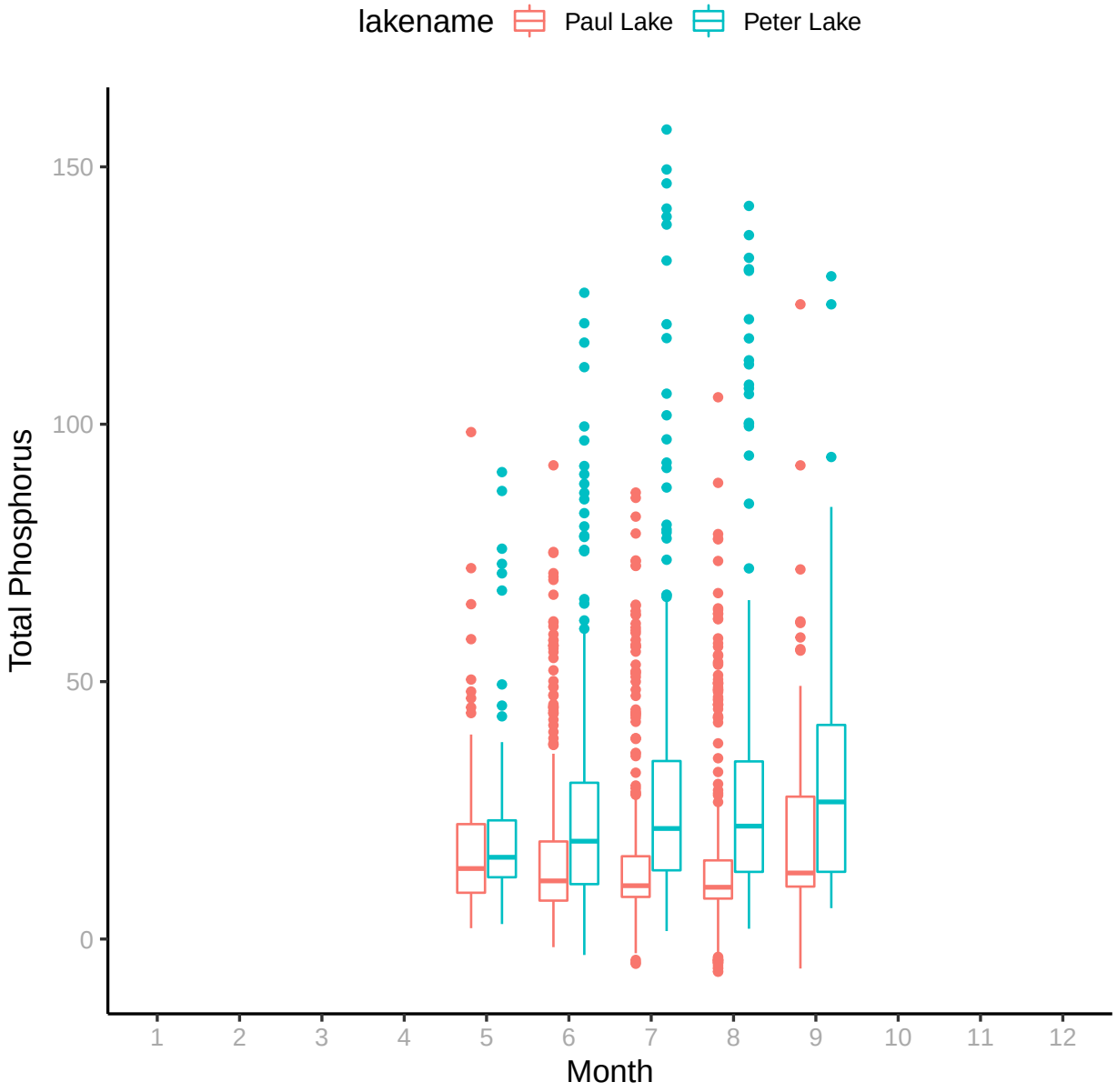
```
  scale_x_discrete(drop = FALSE)    #override default axis labels
print(Temp_boxplot)
```

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



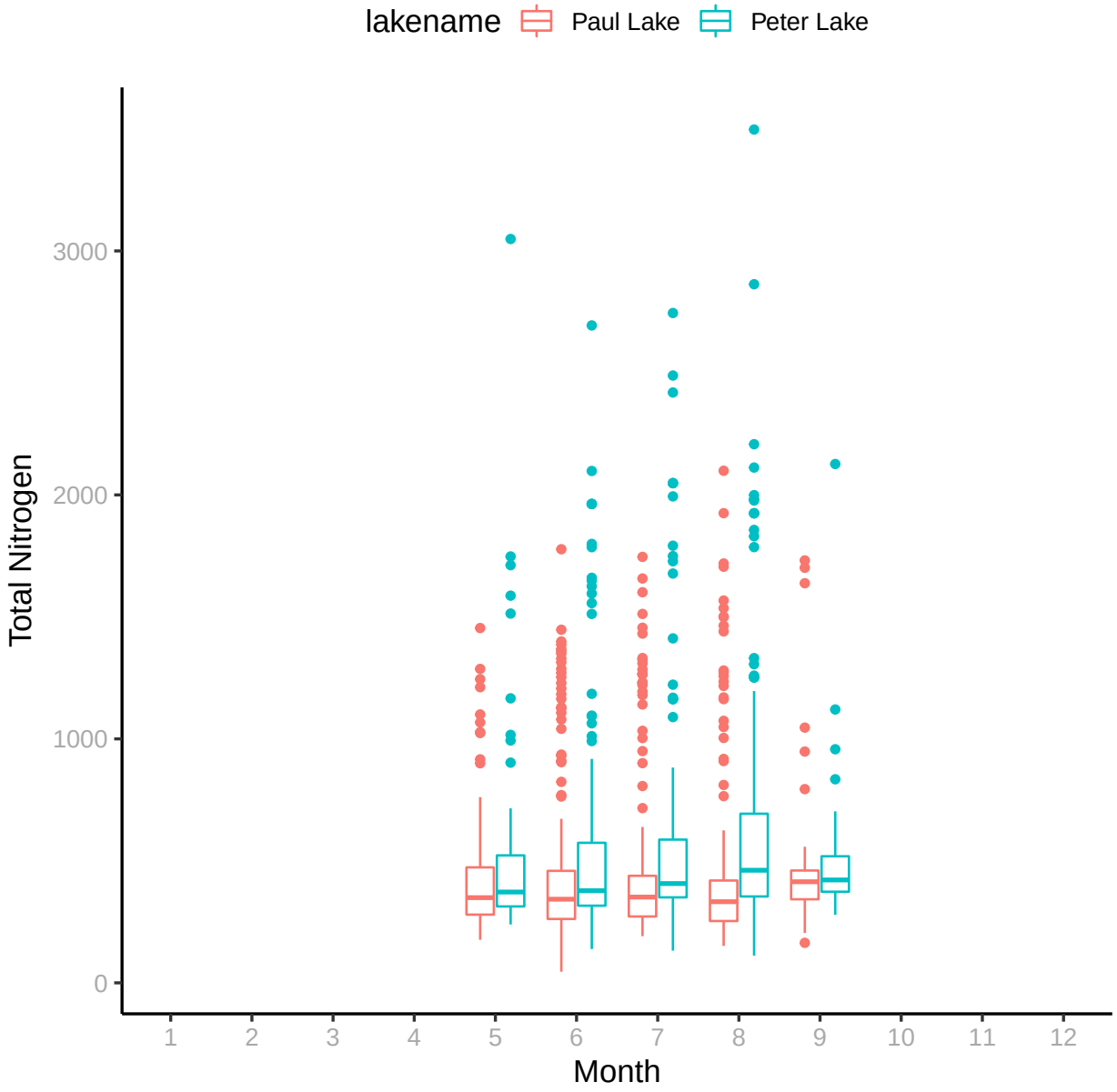

```
TP_boxplot <-
  ggplot(Lake_nutrients_processed,
  aes(x = factor(month, levels = c(1:12)), y = tp_ug)) +
  xlab(expression("Month")) +    #renaming x axis
  ylab(expression("Total Phosphorus")) +    #renaming y axis
  geom_boxplot(aes(color = lakename)) +    #creating separate color aesthetics for Peter & Paul Lakes
  scale_x_discrete(drop = FALSE)    #override default axis labels
print(TP_boxplot)
```

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




```
TN_boxplot <-
  ggplot(Lake_nutrients_processed,
  aes(x = factor(month, levels = c(1:12)), y = tn_ug)) +
  xlab(expression("Month")) +    #renaming x axis
  ylab(expression("Total Nitrogen")) +    #renaming y axis
  geom_boxplot(aes(color = lakename)) +    #creating separate color aesthetics for Peter & Paul Lakes
  scale_x_discrete(drop = FALSE)    #override default axis labels
print(TN_boxplot)
```

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



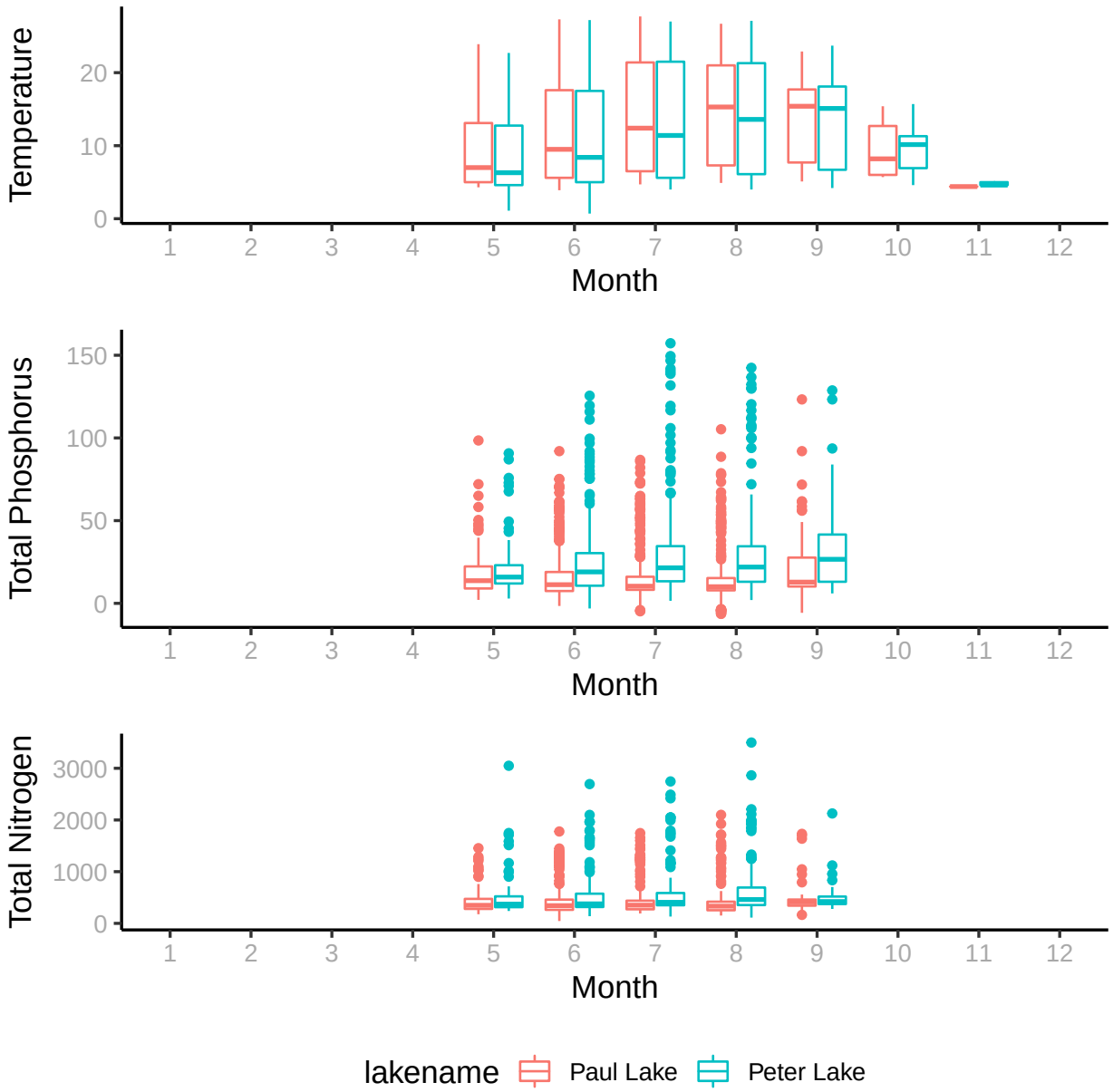

```
# Creating a cowplot to combine the 3 boxplots above (Temperature, TP, & TN)
Combined_boxplot <- plot_grid(Temp_boxplot + theme(legend.position = "none"),
                              TP_boxplot + theme(legend.position = "none"),
                              TN_boxplot + theme(legend.position = "bottom"),
                              #creating only 1 legend
                            nrow = 3, align = 'v', rel_heights = c(1, 1.25, 1.25))
```

```
## 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).
```

```
#organizing the 3 boxplots into 3 rows, aligning them vertically,
#and setting their relative heights
print(Combined_boxplot)
```



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

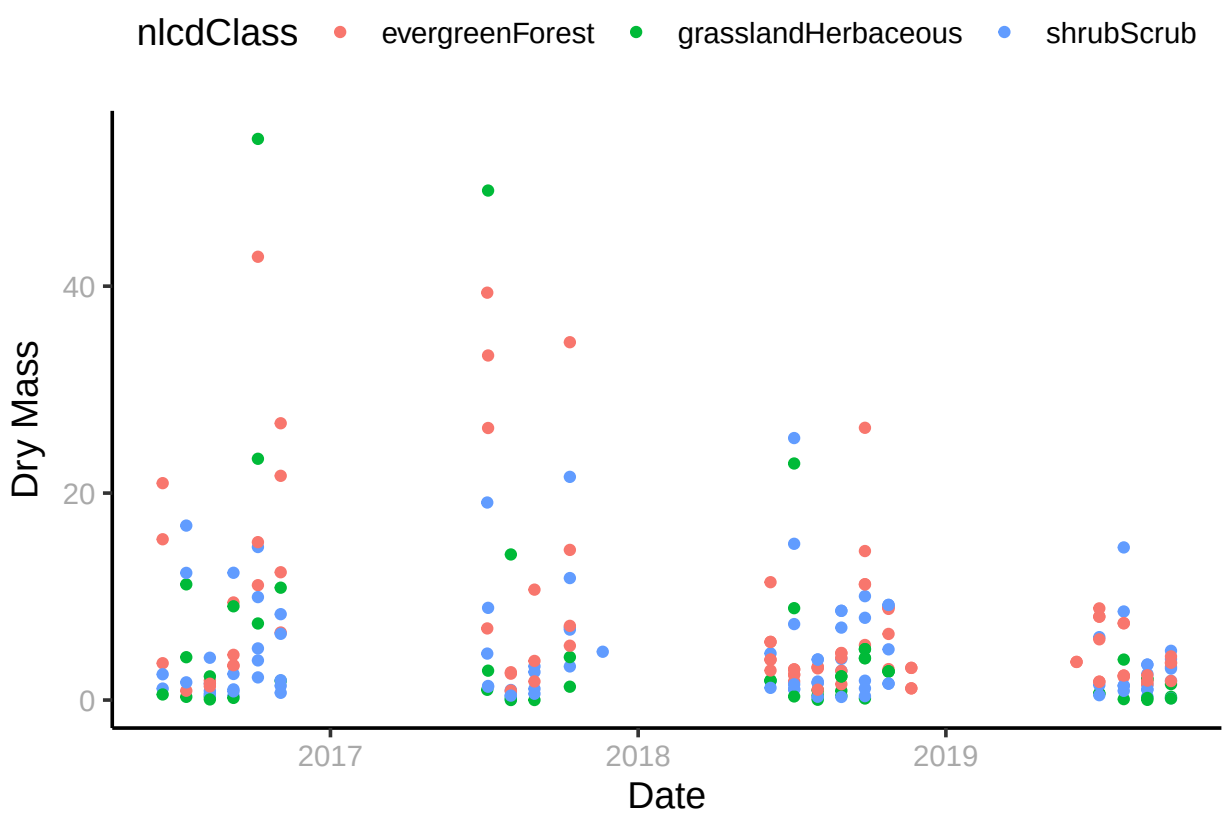
Answer: The temperature, total phosphorus, and total nitrogen values for both Peter and Paul Lakes vary month-by-month and between the two sites. However, there is a general trend observed in which all 3 variables peak during late-summer, around July-August. More specifically, temperatures of Peter and Paul Lake seem to be relatively similar with lower temperatures recorded in May, June, September, October, and November and higher temperatures recorded in July and August. Similarly, the total phosphorus and total nitrogren values are highest in July and August. Conversely, there is a slightly higher concentration of total phosphorus and total nitrogen in Peter Lake than Paul Lake in June, July, August, and Semptember.

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)

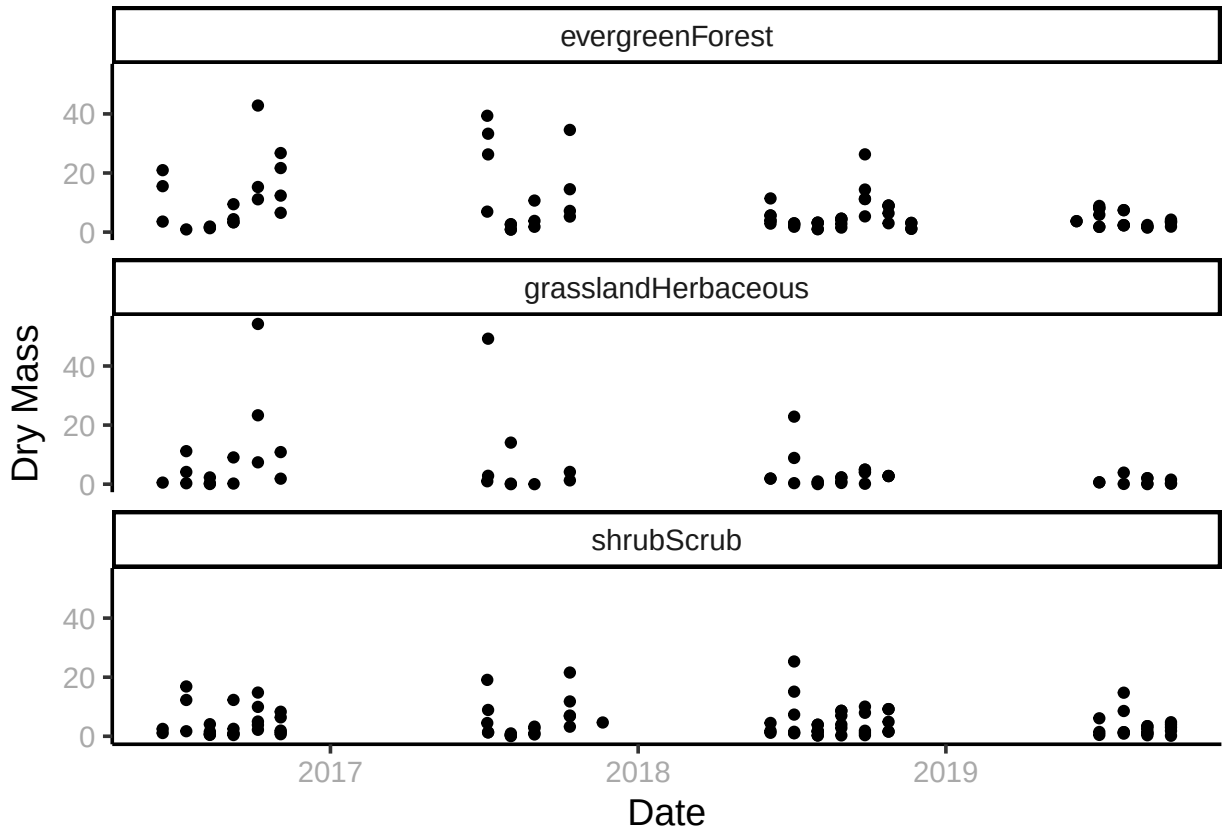
7. [Niwot Ridge] Now, plot the same plot but with NLCD classes separated into three facets rather than

separated by color.

```
#6. Plotting only the "Needles" functional group of the litter dataset using the subset function
# Plotting dry mass of needle litter by date
Needles_plot <-
  ggplot(subset(Litter_processed, functionalGroup == "Needles"),
  aes(x = collectDate, y = dryMass, color = nlcdClass)) +    #creating a separate color
                                                             #aesthetic for NLCD Classes
  xlab(expression("Date")) +    #renaming the x axis
  ylab(expression("Dry Mass")) +    #renaming the y axis
  geom_point()
print(Needles_plot)
```




```
#7. Plotting only the "Needles" functional group of the litter dataset using the subset function
# Plotting dry mass of needle litter by date AND separating NLCD classes into 3 facets
Needles_facets_plot <-
  ggplot(subset(Litter_processed, functionalGroup == "Needles"),
  aes(x = collectDate, y = dryMass)) +
  xlab(expression("Date")) +    #renaming the x axis
  ylab(expression("Dry Mass")) +    #renaming the y axis
  geom_point() +
  facet_wrap(vars(nlcdClass), nrow = 3)    #separating NLCD classes into 3 facets
print(Needles_facets_plot)
```



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

Answer: I think plot 7 is a more effective visualization of dry mass by NLCD class type because it allows for easier comparision between the 3 NLCD classes. When plotting dry mass by date in plot 6, although there are different colors used to represent the 3 different NLCD classes, it is difficult to get a sense of the true distribution of each class because some points within the graph are overlapped and clustered. In plot 7, you can more clearly see differences in the classes that comprise the total dry mass of needles.