Assignment 5: Data Visualization

Abhay V Rao

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/av241/Documents/Environmental_Data_Analytics_2022/Assignments"
```

```
library(tidyverse)
```

```
## -- 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

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

```
library(cowplot)

PeterPaul.A05 <-
  read.csv("../Data/Processed/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv",
```

```
           stringsAsFactors = TRUE)

Litter.A05 <-
  read.csv("../Data/Processed/NEON_NIWO_Litter_mass_trap_Processed.csv",
           stringsAsFactors = TRUE)

#2
PeterPaul.A05$sampledate <- as.Date(PeterPaul.A05$sampledate, format = "%Y-%m-%d")
class(PeterPaul.A05$sampledate)
```

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

Define your theme

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

```
#3

mytheme <- theme_gray(base_size = 10) +
  theme(axis.text = element_text(color = "darkgrey"),
        legend.position = "bottom")


#Setting the theme
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()`).

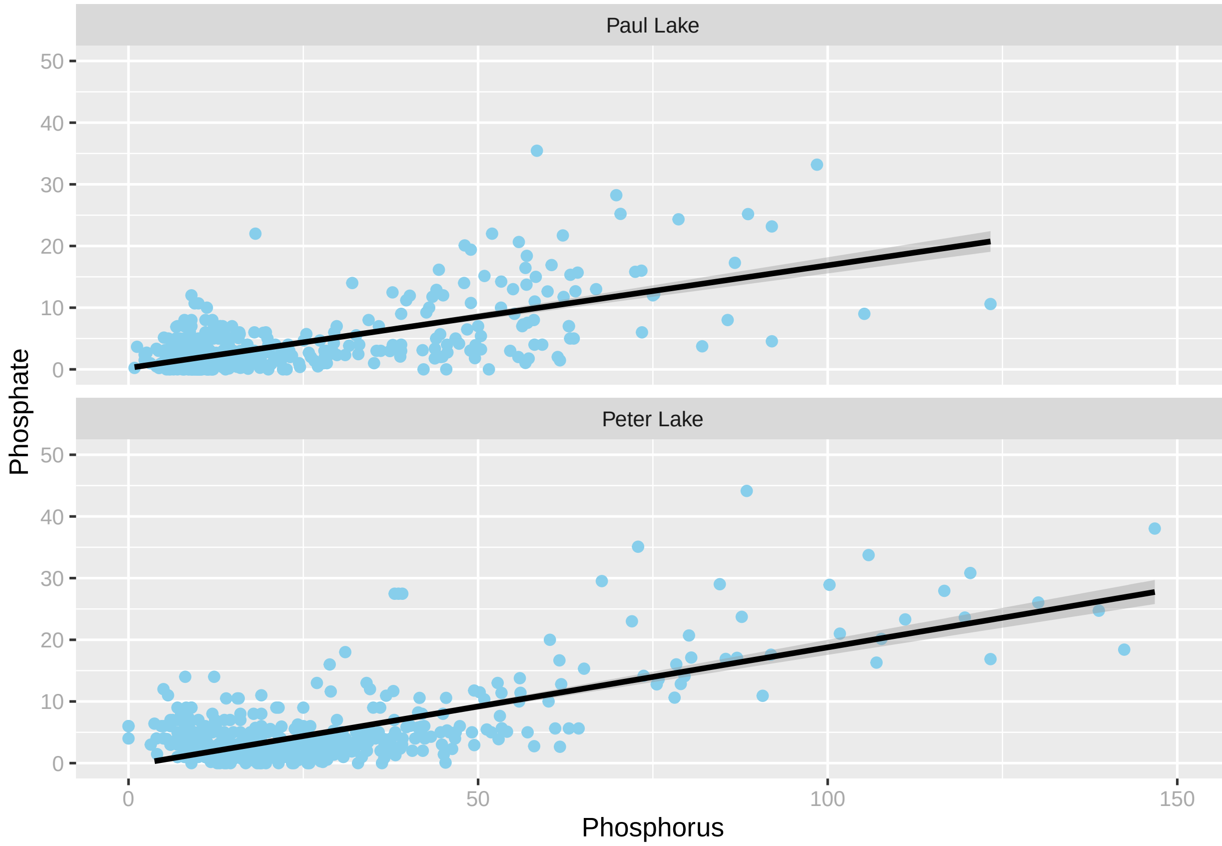
```
#4
PeterPaul.tpug.po4 <-
  ggplot(PeterPaul.A05, aes(x = tp_ug, y = po4)) +
  geom_point(color = "skyblue") +
  xlim(0, 150) +
  ylim(0, 50)+
  geom_smooth(method = lm, color = "black") +
  facet_wrap(vars(lakename), nrow = 2) +
  xlab("Phosphorus")+
  ylab("Phosphate")
print(PeterPaul.tpug.po4)
```

```
## `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 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.

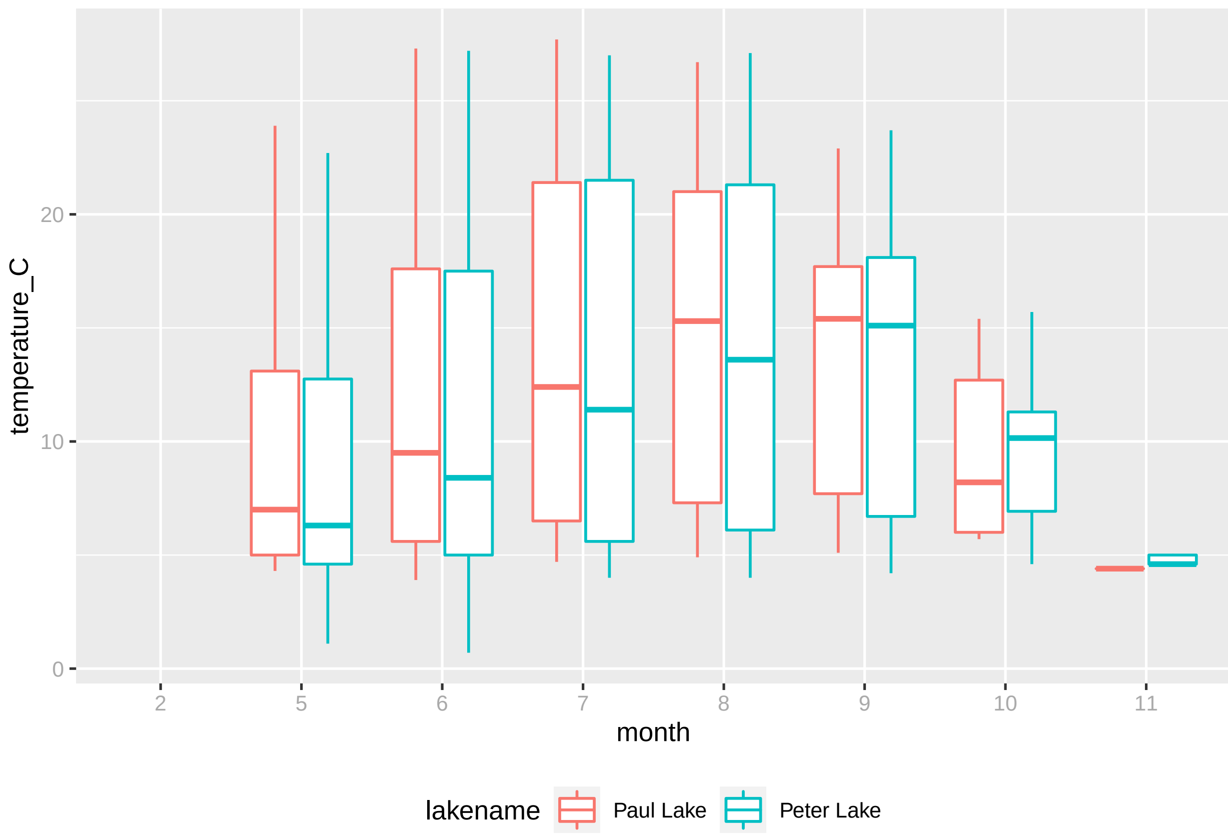
```
#5

#Changing month data to factors from integers
PeterPaul.A05$month <- as.factor(PeterPaul.A05$month)
class(PeterPaul.A05$month)
```

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

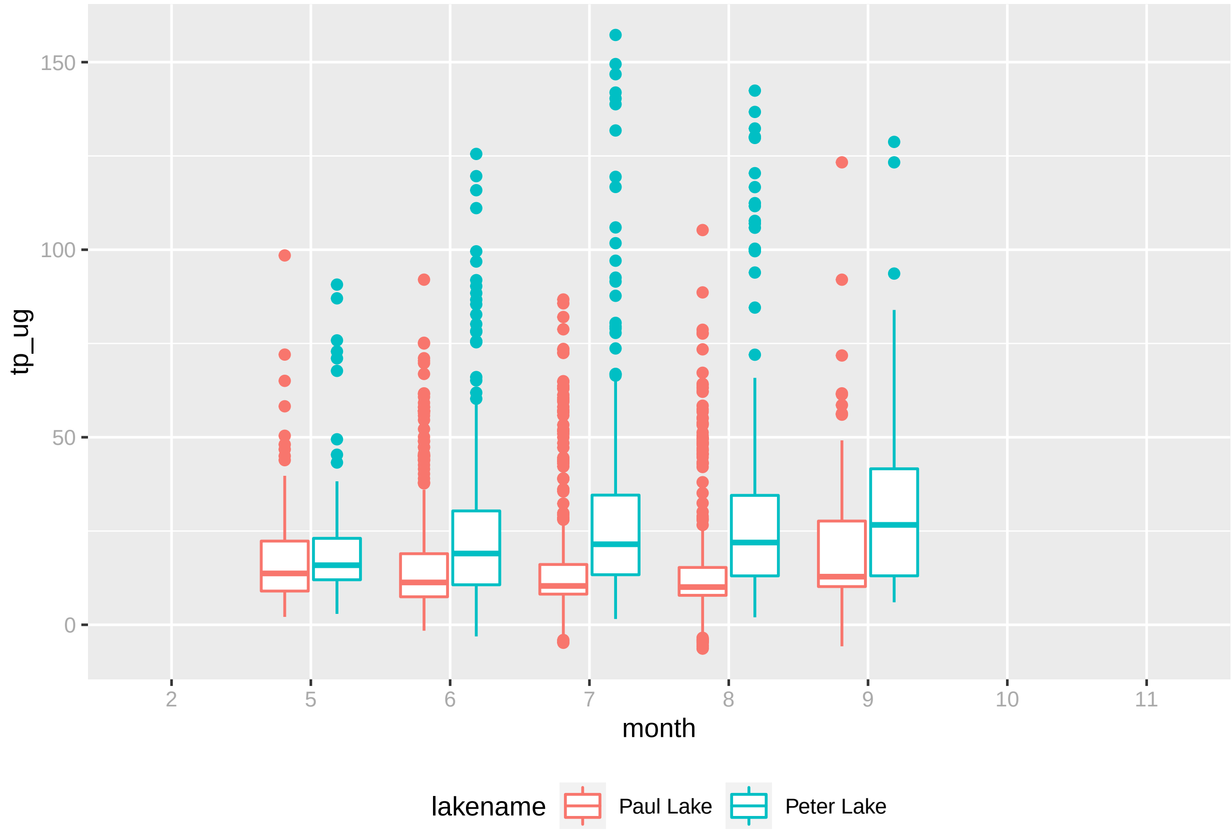
```
#Generating boxplots for temp, tp and tn.
PeterPaulTemp <-
  ggplot(PeterPaul.A05, aes(x = month, y = temperature_C, color = lakename)) +
  geom_boxplot()
PeterPaulTemp
```

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



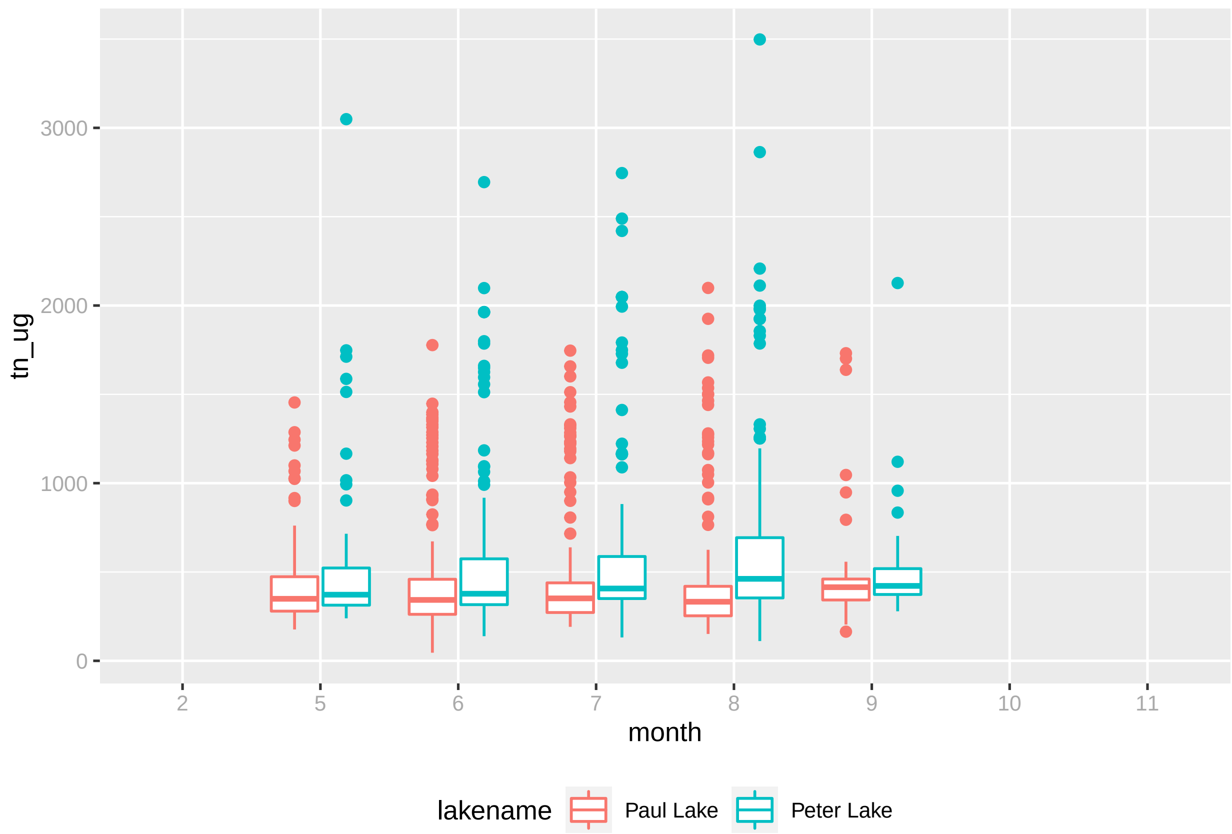

```
PeterPaulTP <-
  ggplot(PeterPaul.A05, aes(x = month, y = tp_ug, color = lakename)) +
  geom_boxplot()
PeterPaulTP
```

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




```
PeterPaulTN <-
  ggplot(PeterPaul.A05, aes(x = month, y = tn_ug, color = lakename)) +
  geom_boxplot()
PeterPaulTN
```

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



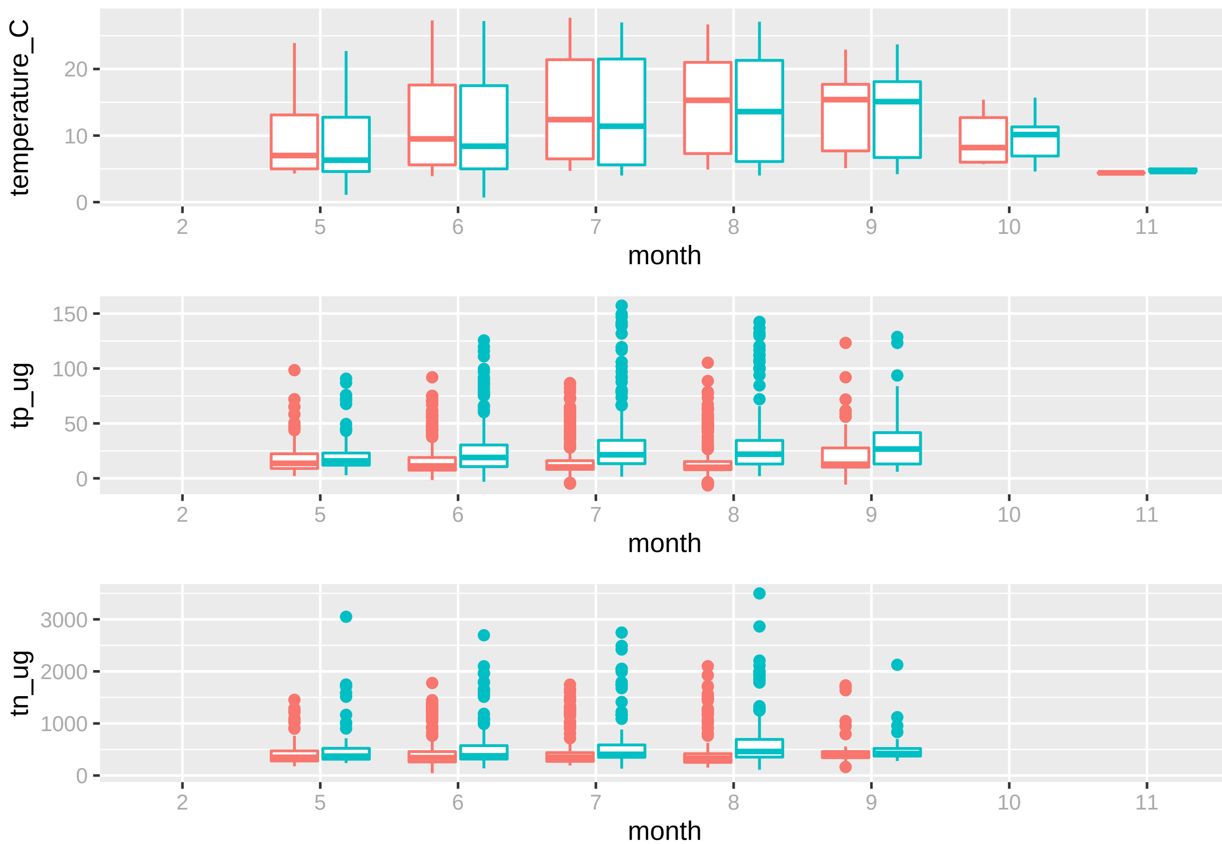

```
# Creating a cowplot with all three boxplots with all legends removed,
PeterPaulrow <- plot_grid(
  PeterPaulTemp + theme(legend.position="none"),
  PeterPaulTP + theme(legend.position="none"),
  PeterPaulTN + theme(legend.position="none"),
  align = 'vh',
  nrow = 3
)
```

```
## 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).
```

```
PeterPaulrow
```



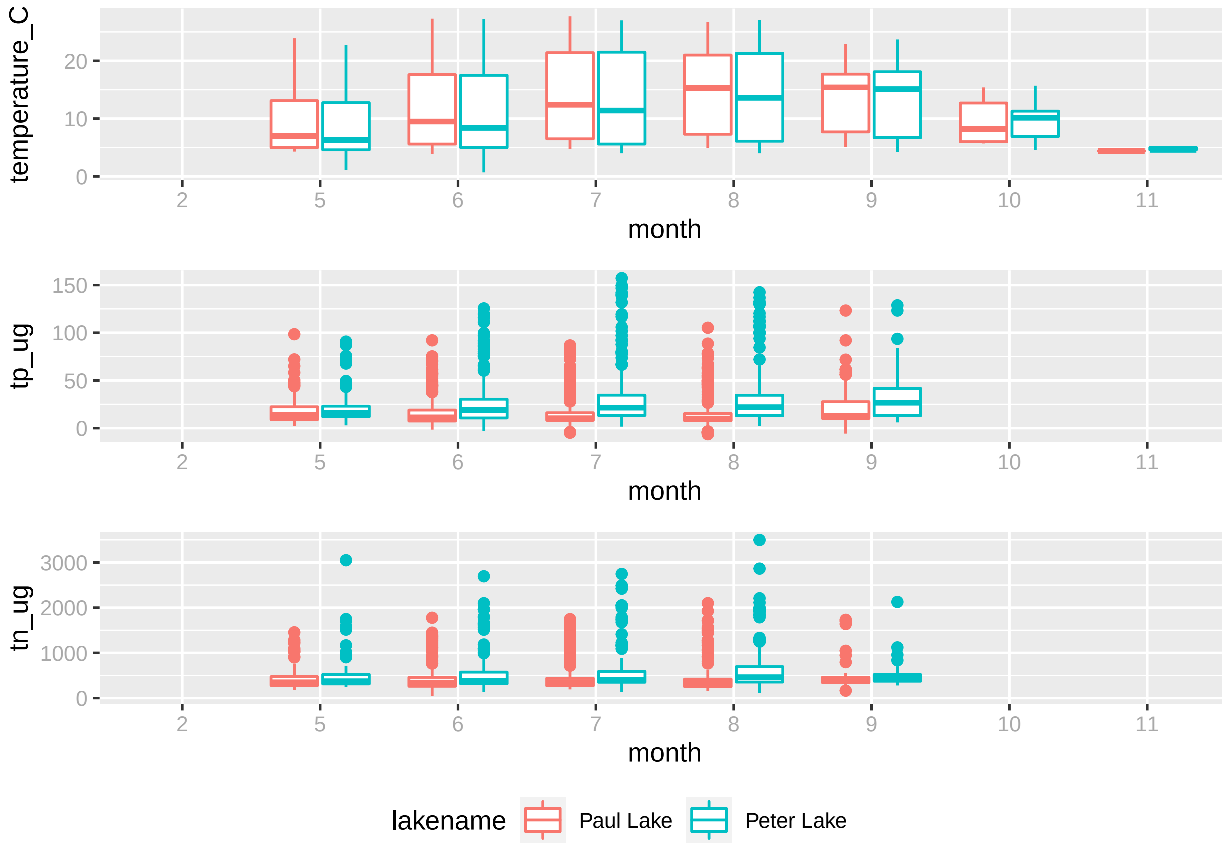

```r
#Extracting the legend for boxplot 1
legend <- get_legend(PeterPaulTemp)
```

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

```r
legend
```

```
## TableGrob (5 x 5) "guide-box": 2 grobs
##                                   z     cells                   name
## 99_262aad63bc8f1a837913c794dfc18661 1 (3-3,3-3)                 guides
##                                   0 (2-4,2-4) legend.box.background
##                                             grob
## 99_262aad63bc8f1a837913c794dfc18661 gtable[layout]
##                                   zeroGrob[NULL]
```

```r
#Creating a cowplot with the legend plotted
plot_grid(PeterPaulrow, legend, nrow = 2 ,rel_heights = c(10,1))
```



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

> Answer: Temperatures tended to be higher for most of the year in Paul lake, with the trend shifting with the onset of cold weather in the fall. Phosphorus and phosphate measurements, in terms of medians and the top end of outliers, were consistently higher in Peter lake: but there was no data on this for Octobers and Novembers, unlike temperature data. We therefore cannot establish if the trend shifted in these months.

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

#Filtering 'Needles' from the dataset

Needles <-
  Litter.A05 %>%
  filter(functionalGroup %in% c("Needles"))

class(Needles$collectDate)
```

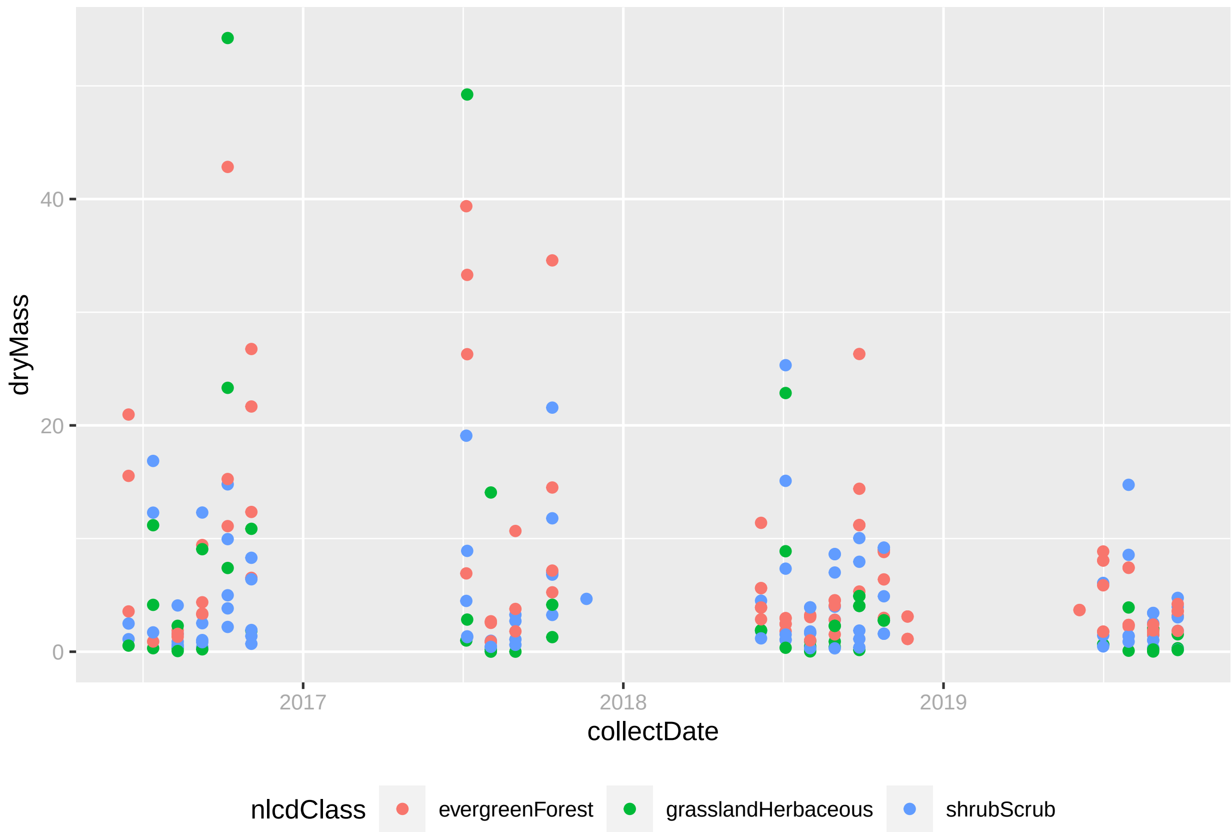
```
## [1] "factor"
```

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

```r
class(Needles$dryMass)
```

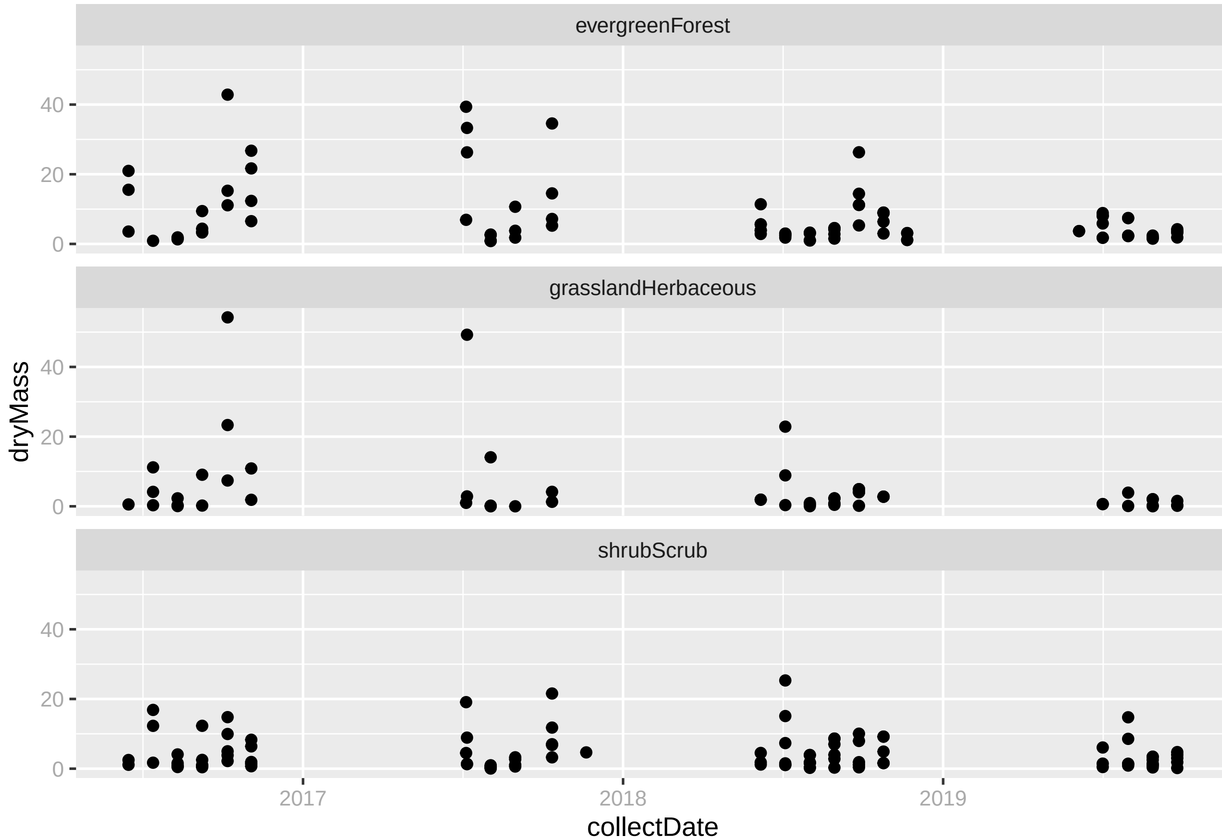
```
## [1] "numeric"
```

```r
#Plotting dry mass by date, separated by NLCD class
Needles_dryMass_Date_NLCD_Color <-
  ggplot(Needles)+
  geom_point(aes(x= collectDate, y= dryMass, color = nlcdClass))
print(Needles_dryMass_Date_NLCD_Color)
```




```r
#7
Needles_dryMass_Date_NLCD_Facet <-
  ggplot(Needles)+
  geom_point(aes(x= collectDate, y= dryMass))+
  facet_wrap(vars(nlcdClass), nrow = 3)

print(Needles_dryMass_Date_NLCD_Facet)
```



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

> Answer: The second plot provides an easier visualization of data across NLCD classes. The first one is comparatively cluttered, and it is difficult to observe trends relating to litter across different land-use areas at a glance.