

Assignment 5: Data Visualization

Taylor Coleman

Spring 2023

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Visualization

Directions

1. Rename this file `<FirstLast>_A05_DataVisualization.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 your code is tidy; use line breaks to ensure your code fits in the knitted output.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up your session

1. Set up your session. Load the tidyverse, lubridate, here, & cowplot packages, and verify your home directory. 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 Loading tidyverse, lubridate, here, and cowplot packages
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   1.0.0
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      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(here)
```

```
## here() starts at /home/guest/ENVR872-RepositoryClone
```

```
library(cowplot)
```

```
##  
## Attaching package: 'cowplot'  
##  
## The following object is masked from 'package:lubridate':  
##  
##    stamp
```

```
# Verifying home directory  
here()
```

```
## [1] "/home/guest/ENVR872-RepositoryClone"
```

```
# Reading in NTL-LTER processed data files and Niwot Ridge litter dataset  
PeterPaul.chem.nutrients <-  
  read.csv(here("Data/Processed_KEY/NTL-LTER_Lake_Chemistry_Nutrients_PeterPaul_Processed.csv"),  
           stringsAsFactors = T)  
Litter <-  
  read.csv(here("Data/Processed_KEY/NEON_NIWO_Litter_mass_trap_Processed.csv"),  
           stringsAsFactors = T)
```

```
#2 Checking to see if R reads dates as date format  
class(PeterPaul.chem.nutrients$sampdate)
```

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

```
class(Litter$collectDate)
```

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

```
# Fixing date objects to be recognized as dates and not factors  
PeterPaul.chem.nutrients$sampdate <- ymd(PeterPaul.chem.nutrients$sampdate)  
Litter$collectDate <- ymd(Litter$collectDate)
```

Define your theme

3. Build a theme and set it as your default theme. Customize the look of at least two of the following:

- Plot background
- Plot title
- Axis labels
- Axis ticks/gridlines
- Legend

```
#3 Constructing a personalized theme
mytheme <- theme_light(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "right",
        plot.title = element_text(face = "bold", hjust = 0.5))

# Setting theme as default
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/or `ylim()`).

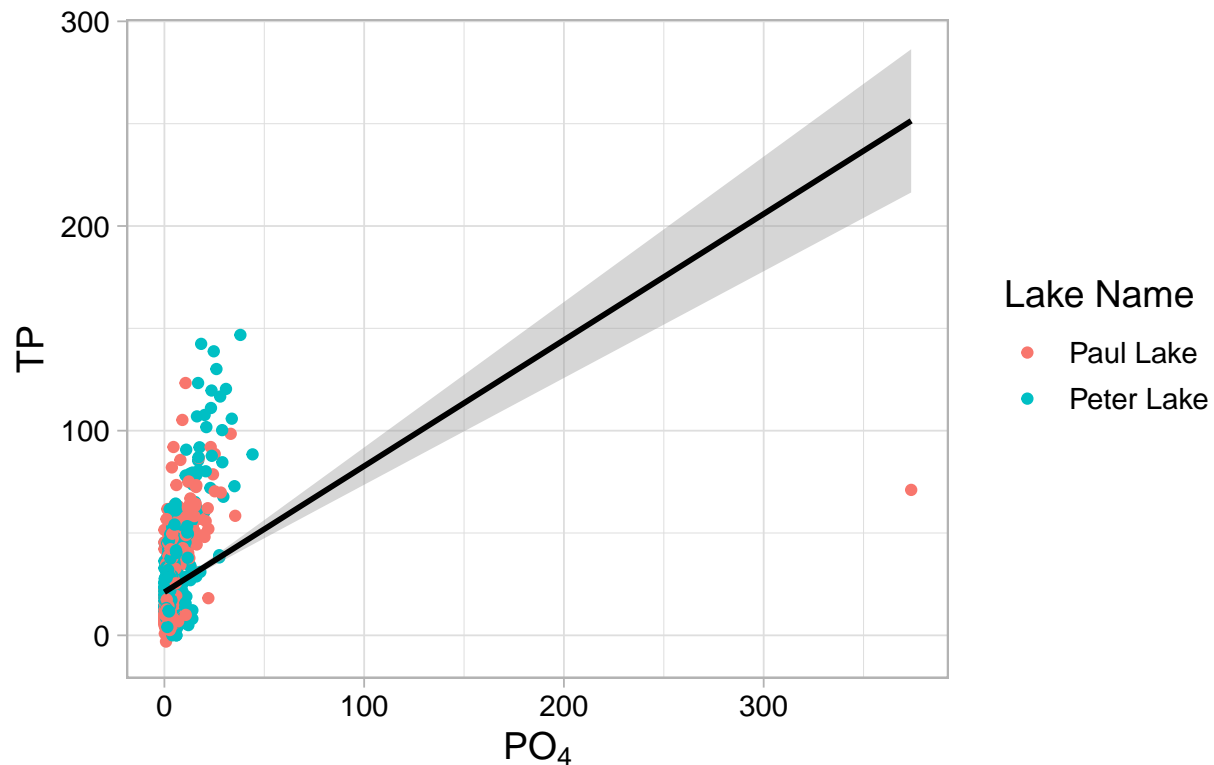
```
#4 Plotting TP x PO4
PeterPaul.TP.PO4.plot <- PeterPaul.chem.nutrients %>%
  ggplot(aes(x = po4, y = tp_ug, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, color = "black") +
  labs(title = expression('TP by PO' [4] * " Levels in Peter and Paul Lakes"),
        x = expression('PO' [4]),
        y = "TP") +
  guides(color = guide_legend(title = "Lake Name"))
print(PeterPaul.TP.PO4.plot)
```

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

```
## Warning: Removed 21946 rows containing non-finite values ('stat_smooth()').
```

```
## Warning: Removed 21946 rows containing missing values ('geom_point()').
```

TP by PO₄ Levels in Peter and Paul Lakes



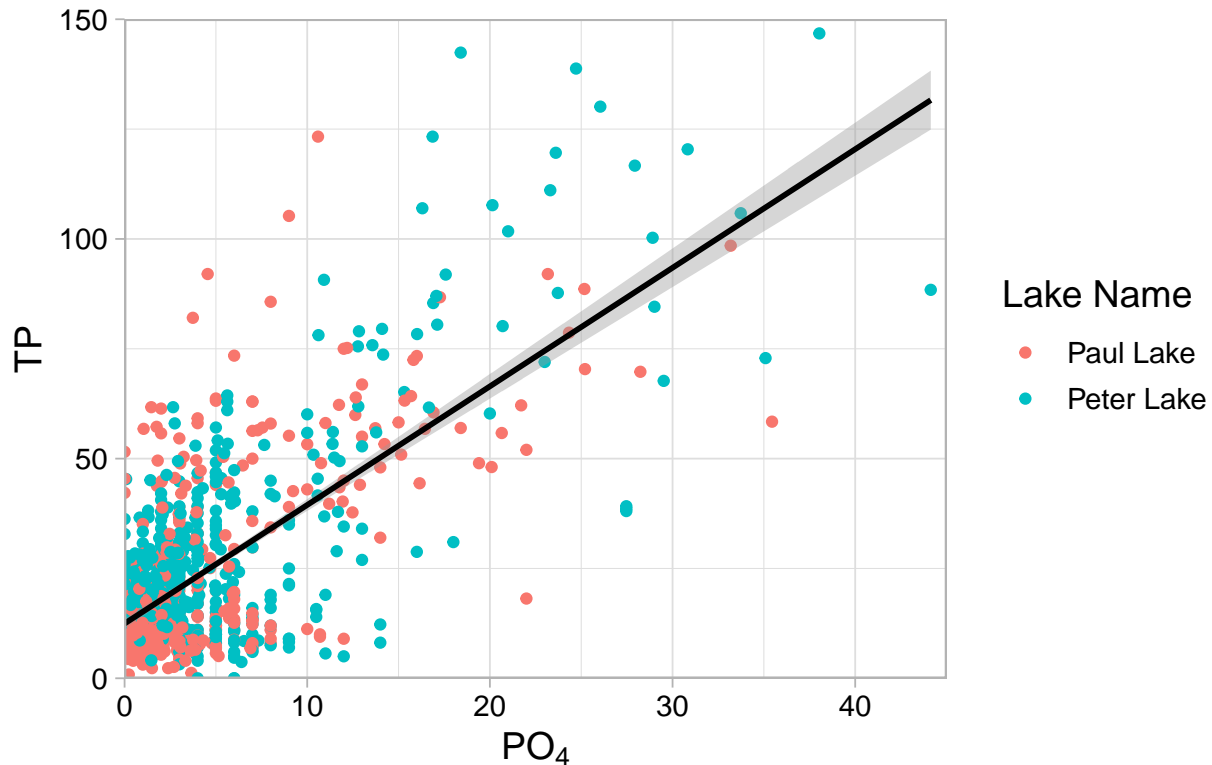
```
#Adjusting axes
PeterPaul.TP.PO4.plot <- PeterPaul.chem.nutrients %>%
  ggplot(aes(x = po4, y = tp_ug, color = lakename)) +
  geom_point() +
  geom_smooth(method = lm, color = "black") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlim(0, 45) +
  ylim(0, 150) +
  labs(title = expression('TP by PO' [4] * " Levels in Peter and Paul Lakes"),
       x = expression('PO' [4]),
       y = "TP") +
  coord_cartesian(expand = FALSE) + #Setting origin to (0,0)
  guides(color = guide_legend(title = "Lake Name"))
print(PeterPaul.TP.PO4.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()').
```

TP by PO₄ Levels in Peter and Paul Lakes



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 built-in variable called `month.abb` that returns a list of months; see <https://r-lang.com/month-abb-in-r-with-example>

```
#5 Converting month to a factor
factor(PeterPaul.chem.nutrients$month,
       levels = 1:12)
```

```
##      [1] 5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  5
##     [25] 5  5  5  5  5  5  5  5  5  5  5  5  5  5  5  6  6  6  6  6  6  6
##     [49] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##     [73] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##     [97] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##    [121] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##    [145] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##    [169] 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
##    [193] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
##    [217] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
##    [241] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
##    [265] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
##    [289] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
##    [313] 7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
```

```

## [337] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [361] 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [385] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [409] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [433] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [457] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [481] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [505] 8 8 8 8 8 8 8 8 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [529] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [553] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [577] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [601] 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [625] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [649] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [673] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [697] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [721] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [745] 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [769] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [793] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [817] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [841] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [865] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [889] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [913] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [937] 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8
## [961] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [985] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [1009] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [1033] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [1057] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [1081] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9
## [1105] 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
## [1129] 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
## [1153] 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
## [1177] 9 9 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1201] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6
## [1225] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1249] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1273] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1297] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1321] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1345] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1369] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [1393] 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1417] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1441] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1465] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1489] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1513] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1537] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [1561] 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8
## [1585] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [1609] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

###	[9409]	5	5	5	5	5	5	5	5	5	5	5	5	5	6	6	6	6	6	6	6	6
###	[9433]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9457]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9481]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9505]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9529]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9553]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9577]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
###	[9601]	6	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7	7	7	7	7
###	[9625]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9649]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9673]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9697]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9721]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9745]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9769]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
###	[9793]	7	7	7	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9817]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9841]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9865]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9889]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9913]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9937]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
###	[9961]	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
###	[9985]	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	5	5	5	5	5
##	[10009]	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
##	[10033]	5	5	5	5	5	5	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6
##	[10057]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[10081]	6	6																			

[illegible]

[illegible]

##	[13297]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
##	[13321]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
##	[13345]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5
##	[13369]	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
##	[13393]	5	5	5	5	5	5	5	5	5	5	5	5	5	5	6	6	6	6	6	6
##	[13417]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13441]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13465]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13489]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13513]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13537]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13561]	6	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7	7	7	7
##	[13585]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
##	[13609]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
##	[13633]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
##	[13657]	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
##	[13681]	7	7	7	7	7	7	7	7	7	7	7	8	8	8	8	8	8	8	8	8
##	[13705]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
##	[13729]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
##	[13753]	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5
##	[13777]	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
##	[13801]	5	5	5	5	5	5	5	5	5	5	5	5	5	5	6	6	6	6	6	6
##	[13825]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13849]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13873]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13897]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13921]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13945]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13969]	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
##	[13993]	6	6	6</																	

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

## [21073] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [21097] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [21121] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [21145] 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6
## [21169] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21193] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21217] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21241] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21265] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21289] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21313] 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21337] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21361] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21385] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21409] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21433] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21457] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21481] 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21505] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21529] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21553] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21577] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21601] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 5 5 5 5 5 5 5 5 5 6
## [21625] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21649] 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21673] 7 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21697] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 9 9 5 5 5
## [21721] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6
## [21745] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21769] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7
## [21793] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21817] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21841] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [21865] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9
## [21889] 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 5 5 5 5 5 5 5 5
## [21913] 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [21937] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7
## [21961] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [21985] 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22009] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9
## [22033] 9 9 9 9 9 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6
## [22057] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22081] 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22105] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 8
## [22129] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22153] 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 9 9 9 9 9 9 5
## [22177] 5 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22201] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22225] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22249] 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22273] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22297] 8 8 9 9 9 9 9 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6
## [22321] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7
## [22345] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7

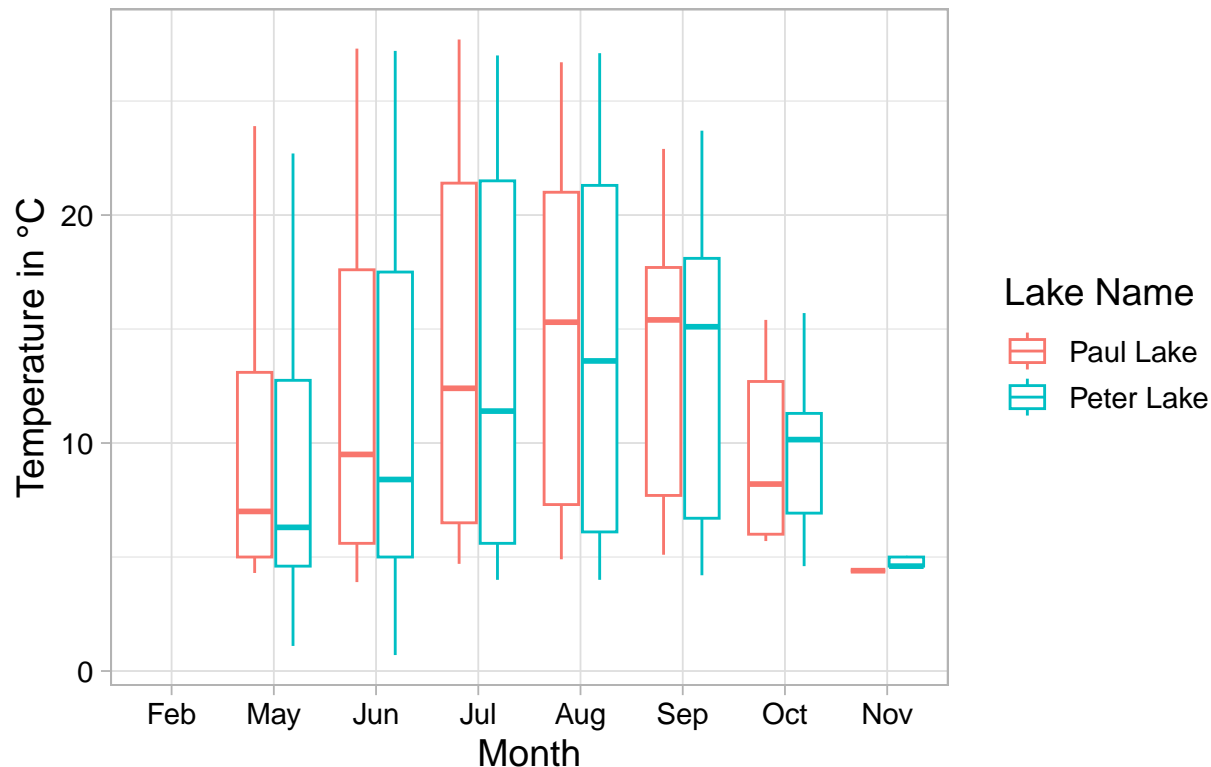
```

```
## [22369] 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22393] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 9 5
## [22417] 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22441] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7
## [22465] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22489] 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22513] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 5
## [22537] 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22561] 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7
## [22585] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8
## [22609] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22633] 8 8 8 8 8 8 8 8 8 5 5 5 5 5 5 5 6 6 6 6 6 6
## [22657] 6 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 5 5 5 5 5 5
## [22681] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22705] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22729] 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22753] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22777] 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22801] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22825] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 5 5 5 5 5
## [22849] 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22873] 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
## [22897] 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22921] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
## [22945] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8
## [22969] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## [22993] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## Levels: 1 2 3 4 5 6 7 8 9 10 11 12
```

```
# (a) Boxplot for temperature from samples obtained from 1984-2016
PeterPaul.temp.plot <- PeterPaul.chem.nutrients %>%
  ggplot(aes(x = factor(month, levels = 1:12, labels = month.abb),
              y = temperature_C, color = lakename)) +
  geom_boxplot() +
  labs(title = "Temperature of Peter and Paul Lakes by Month",
       x = "Month",
       y = ("Temperature in \u00B0C")) +
  guides(color = guide_legend(title = "Lake Name"))
print(PeterPaul.temp.plot)
```

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

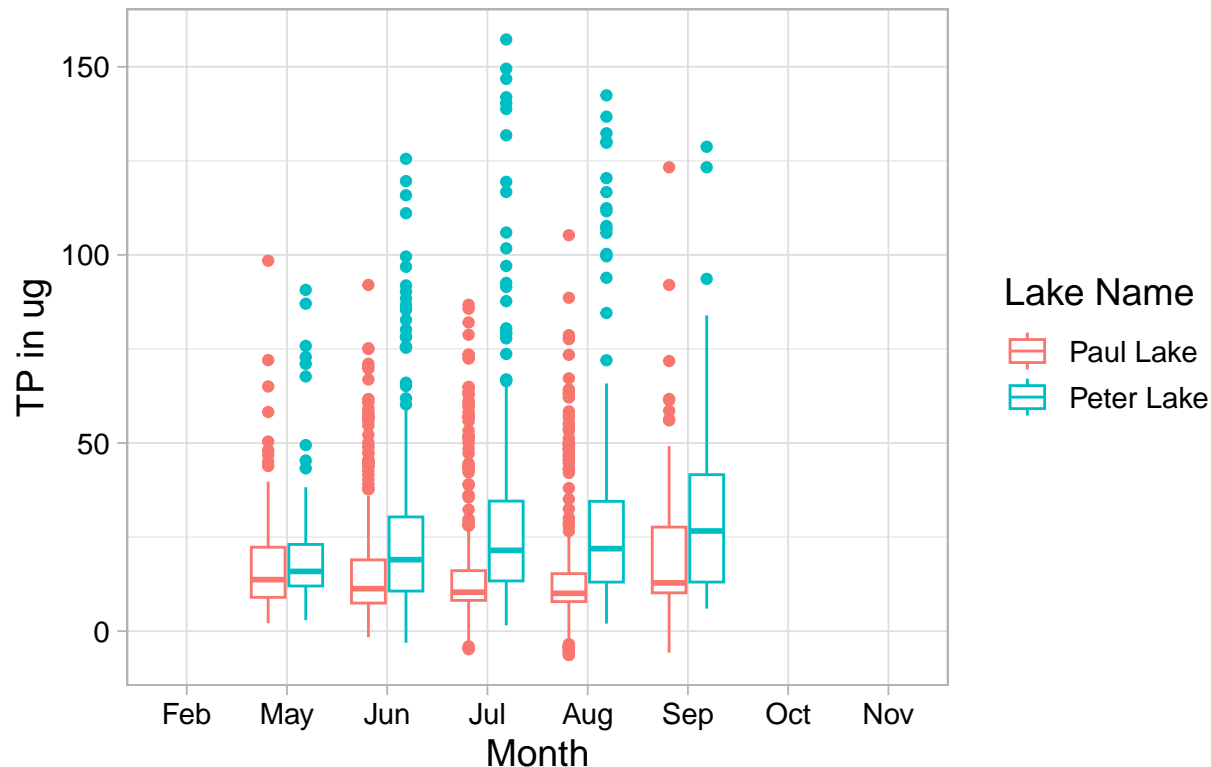
Temperature of Peter and Paul Lakes by Month



```
# (b) Boxplot for TP
PeterPaul.TP.plot <- PeterPaul.chem.nutrients %>%
  ggplot(aes(x = factor(month, levels = 1:12, labels = month.abb),
              y = tp_ug, color = lakename)) +
  geom_boxplot() +
  labs(title = "TP Levels in Peter and Paul Lakes by Month",
       x = "Month",
       y = "TP in ug") +
  guides(color = guide_legend(title = "Lake Name"))
print(PeterPaul.TP.plot)
```

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

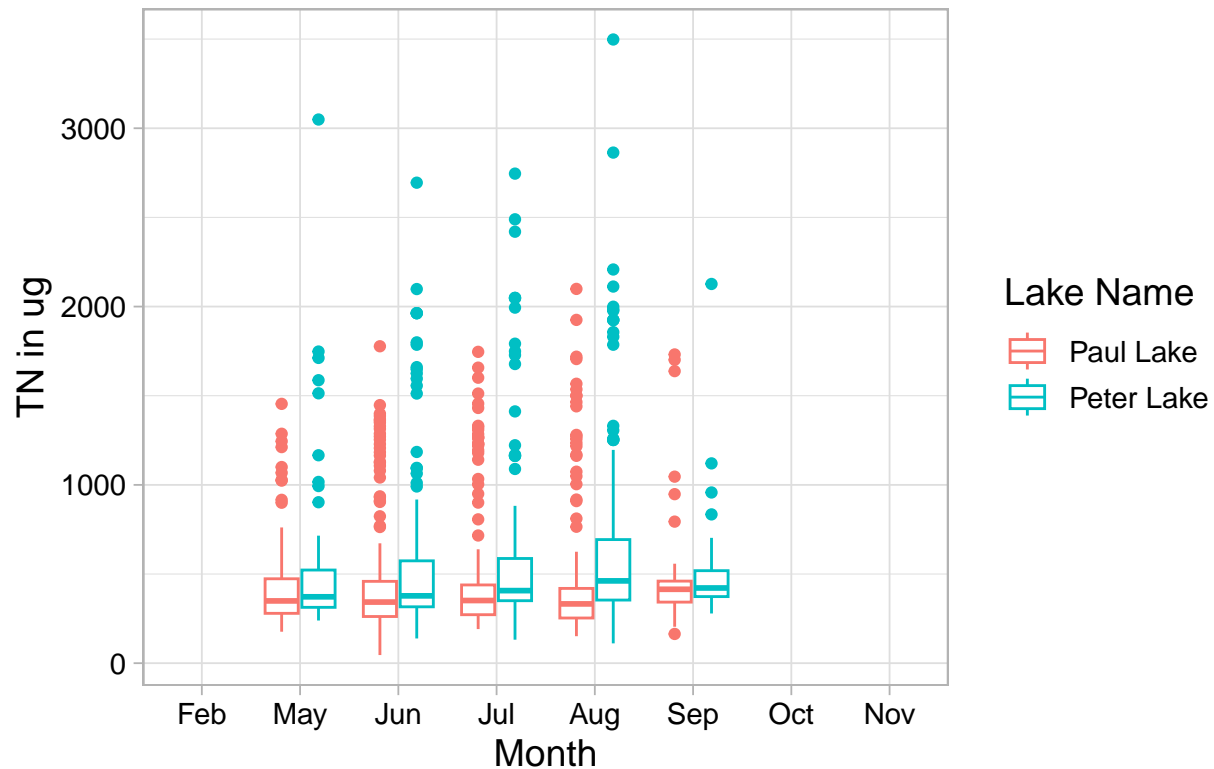

TP Levels in Peter and Paul Lakes by Month



```
# (c) Boxplot for TN
PeterPaul.TN.plot <- PeterPaul.chem.nutrients %>%
  ggplot(aes(x = factor(month, levels = 1:12, labels = month.abb),
               y = tn_ug, color = lakename,)) +
  geom_boxplot() +
  labs(title = "TN Levels in Peter and Paul Lakes by Month",
        x = "Month",
        y = "TN in ug") +
  guides(color = guide_legend(title = "Lake Name"))
print(PeterPaul.TN.plot)
```

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

TN Levels in Peter and Paul Lakes by Month



```
# Using cowplot to combine the 3 graphs
PeterPaul.combined.plot <- plot_grid(PeterPaul.TN.plot + theme(axis.text.x = element_blank(),
                                                                axis.title.x = element_blank(),
                                                                legend.position = "none",
                                                                plot.title = element_text(hjust = 0.5)),
                                     PeterPaul.TP.plot + theme(axis.text.x = element_blank(),
                                                                axis.title.x = element_blank(),
                                                                legend.position = "none",
                                                                plot.title = element_text(hjust = 0.5)),
                                     PeterPaul.temp.plot + theme(legend.position = "none",
                                                                plot.title = element_text(hjust = 0.5)),
                                     axis = "b", nrow = 3, rel_heights = c(1, 1, 1))
```

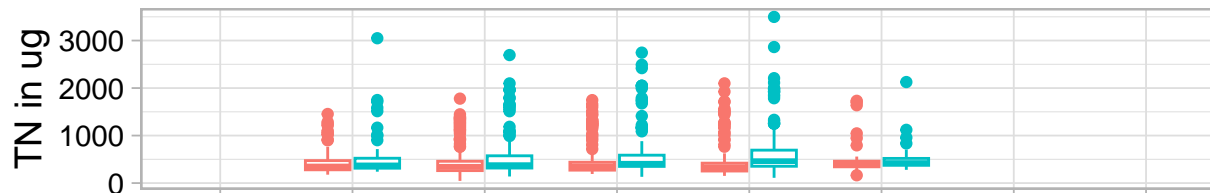
```
## Warning: Removed 21583 rows containing non-finite values ('stat_boxplot()').
```

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

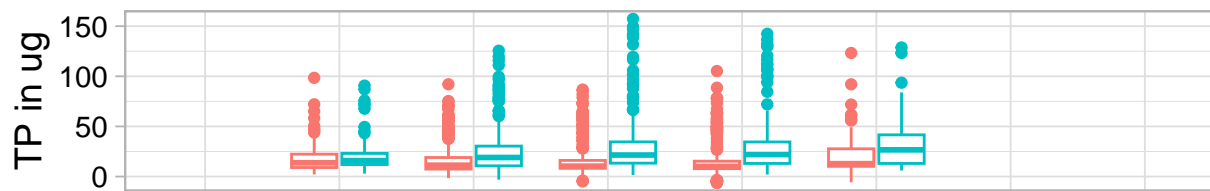
```
## Warning: Removed 3566 rows containing non-finite values ('stat_boxplot()').
```

```
print(PeterPaul.combined.plot)
```

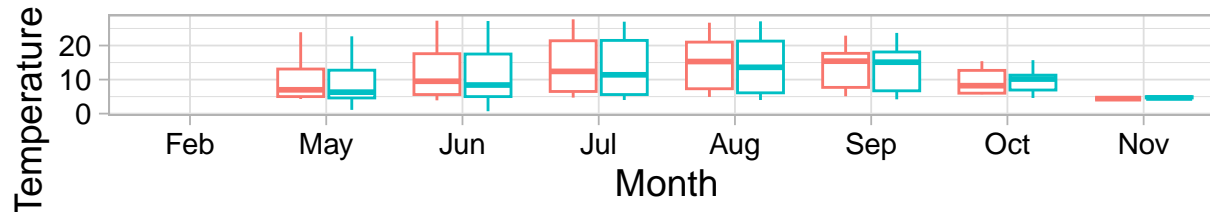
TN Levels in Peter and Paul Lakes by Month



TP Levels in Peter and Paul Lakes by Month



Temperature of Peter and Paul Lakes by Month

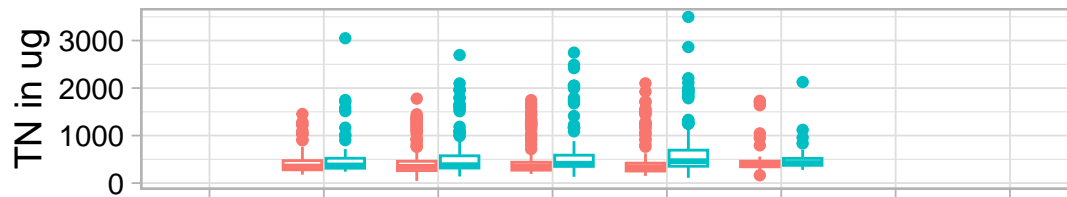


```
# Adding legend back manually
legend <- get_legend(PeterPaul.temp.plot)
```

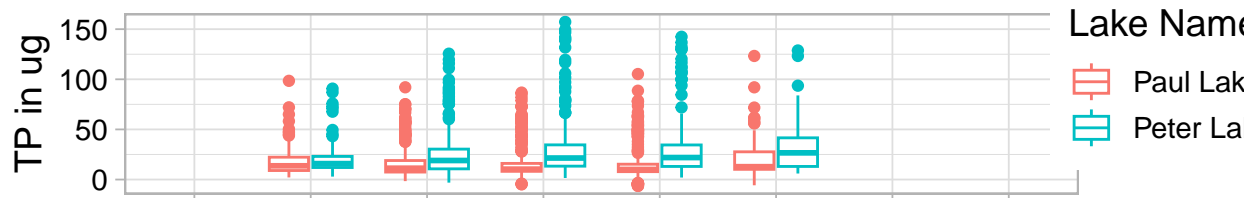
```
## Warning: Removed 3566 rows containing non-finite values ('stat_boxplot()').
```

```
PeterPaul.final.combined.plot <- plot_grid(PeterPaul.combined.plot,
                                           legend,
                                           rel_widths = c(3, .4))
print(PeterPaul.final.combined.plot)
```

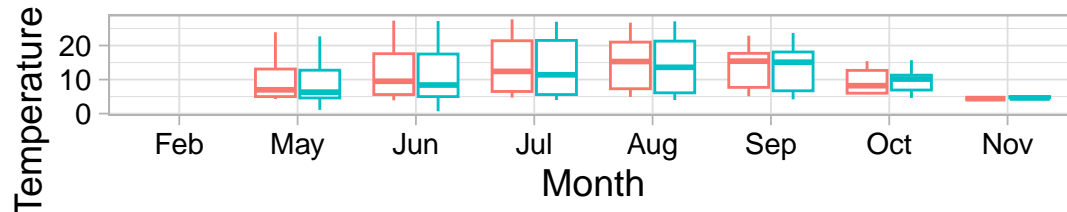
TN Levels in Peter and Paul Lakes by Month



TP Levels in Peter and Paul Lakes by Month



Temperature of Peter and Paul Lakes by Month



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

Answer: From the temperature plot, Paul Lake had higher median temperatures than Peter Lake until the final two sampled months of October and November. The median temperatures for Peter and Paul lakes were very similar in September, beyond which a change in the trend could be observed and the IQR or range of temperatures greatly diminished for both lakes (which makes sense with the onset of fall). The TP plot revealed that Peter Lake consistently had higher median levels of TP for all sample months whereas median TN concentrations were similar between the two lakes, with Peter Lake consistently exhibiting a greater presence of high “outlier” values. Peter Lake had higher outlier values than Paul Lake for its TP concentrations over time as well, with the exception of the month of May.

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.

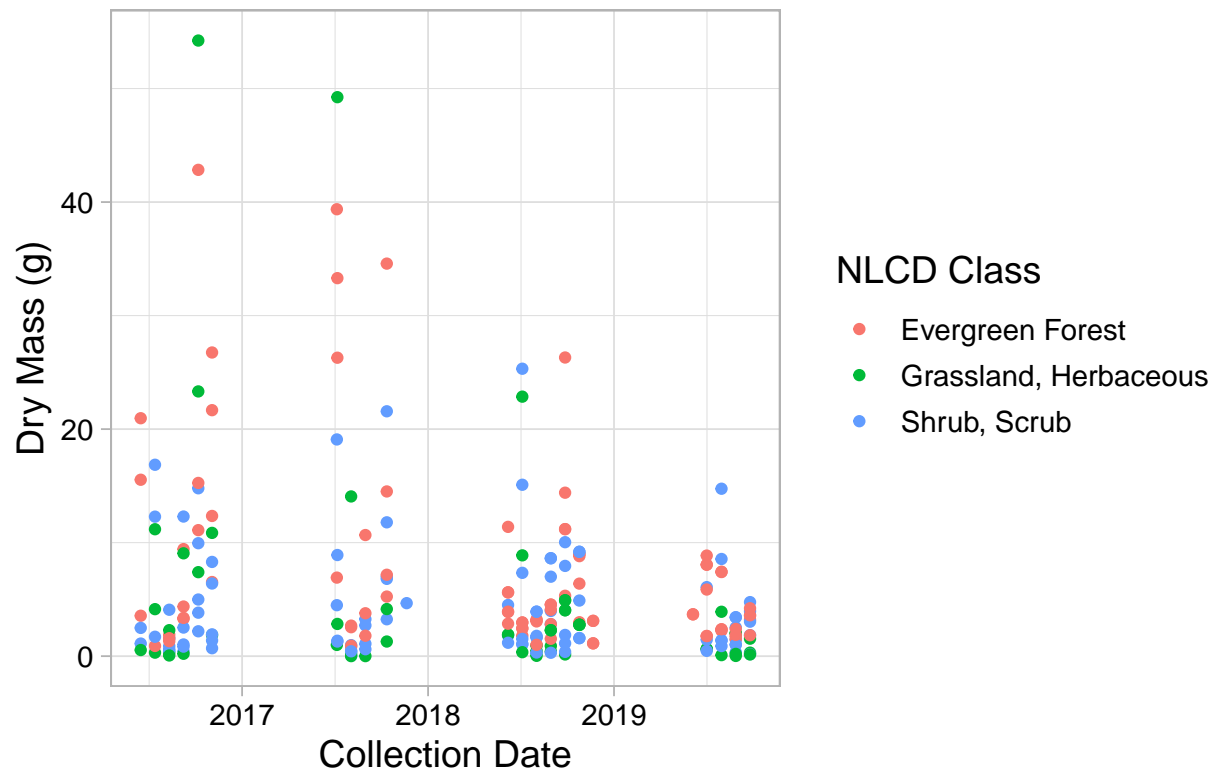
```
#6 Plotting a subset of Litter dataset for "Needles"
Litter.Needles.plot <- Litter %>%
  filter(functionalGroup == "Needles") %>%
  ggplot(aes(x = collectDate, y = dryMass, color = nlcdClass)) +
  geom_point() +
  labs(title = "Needle Litter Dry Mass by Date and NLCD Class",
       x = "Collection Date",
```

```

y = "Dry Mass (g)" +
theme(plot.title = element_text(hjust = 0.5, face = "bold")) +
scale_color_discrete(name = "NLCD Class",
                      label = c("Evergreen Forest",
                                "Grassland, Herbaceous",
                                "Shrub, Scrub")) # Unnecessary, but something I wanted to experiment w
print(Litter.Needles.plot)

```

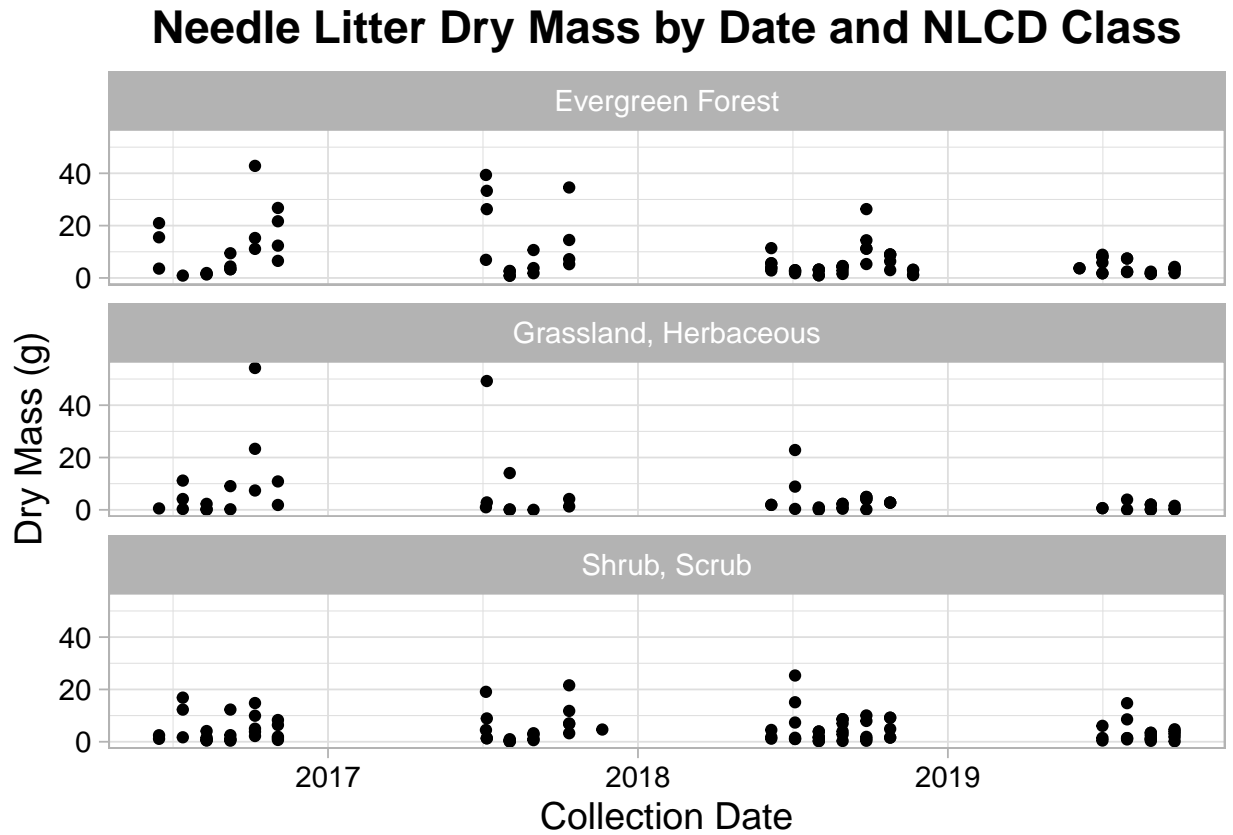
Needle Litter Dry Mass by Date and NLCD Class



```

#7 Separating each NLCD class into facets
Litter.Needles.facets.plot <- Litter %>%
  filter(functionalGroup == "Needles") %>%
  ggplot(aes(x = collectDate, y = dryMass)) +
  geom_point() +
  labs(title = "Needle Litter Dry Mass by Date and NLCD Class",
        x = "Collection Date",
        y = "Dry Mass (g)") +
  theme(plot.title = element_text(hjust = 0.5, face = "bold")) +
  facet_wrap(vars(nlcdClass), nrow = 3,
             labeller = labeller(nlcdClass = c("evergreenForest" = "Evergreen Forest",
                                                "grasslandHerbaceous" = "Grassland, Herbaceous",
                                                "shrubScrub" = "Shrub, Scrub"))))
print(Litter.Needles.facets.plot)

```



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

Answer: I think that plot 7 is more effective at displaying the dry mass values of needle litter because it separates each class type and thereby prevents the data points from each class type from overlapping and being unable to be distinguished from others, as can be the case when investigating the first plot.