Assignment 5: Water Quality in Lakes

Lydie Costes

OVERVIEW

This exercise accompanies the lessons in Water Data Analytics on water quality in lakes

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, check your PDF against the key and then submit your assignment completion survey at https://forms.gle/fSe18vMhgzcjUKM39

Having trouble? See the assignment's answer key if you need a hint. Please try to complete the assignment without the key as much as possible - this is where the learning happens!

Target due date: 2022-02-22

Setup

1. Verify your working directory is set to the R project file. Load the tidyverse, lubridate, and LAGOSNE packages. Set your ggplot theme (can be theme_classic or something else)
2. Load the LAGOSdata database and the trophic state index csv file we created in class.

```
 # 1. Verify directory, load packages, set theme
getwd()
```

```
## [1] "/Users/lydiecostes/Documents/Duke/WaterDataAnalytics/Water_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(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(LAGOSNE)

theme_set(theme_bw())

# 2. Load data
LAGOSdata <- lagosne_load()
```

```
## Warning in (function (version = NULL, fpath = NA) : LAGOSNE version unspecified,
## loading version: 1.087.3
```

```
LAGOStrophic <- read.csv("../Data/Processed/LAGOStrophic.csv")
```

Trophic State Index

3. Similar to the trophic.class column we created in class (determined from TSI.chl values), create two additional columns in the data frame that determine trophic class from TSI.secchi and TSI.tp (call these trophic.class.secchi and trophic.class.tp).

```
# 3. Add additional columns
LAGOStrophic <- LAGOStrophic %>%
  mutate(trophic.class.secchi = case_when(TSI.secchi < 40 ~ "Oligotrophic",
                                          TSI.secchi >= 40 & TSI.secchi < 50 ~ "Mesotrophic",
                                          TSI.secchi >= 50 & TSI.secchi < 70 ~ "Eutrophic",
                                          TSI.secchi >= 70 ~ "Hypereutrophic"),
         trophic.class.tp = case_when(TSI.tp < 40 ~ "Oligotrophic",
                                      TSI.tp >= 40 & TSI.tp < 50 ~ "Mesotrophic",
                                      TSI.tp >= 50 & TSI.tp < 70 ~ "Eutrophic",
                                      TSI.tp >= 70 ~ "Hypereutrophic"))

LAGOStrophic$trophic.class.secchi <- factor(LAGOStrophic$trophic.class.secchi,
                                     levels = c("Oligotrophic", "Mesotrophic", "Eutrophic", "Hypereutrop

LAGOStrophic$trophic.class.tp <- factor(LAGOStrophic$trophic.class.tp,
                                     levels = c("Oligotrophic", "Mesotrophic", "Eutrophic", "Hypereutrop
```

4. How many observations fall into the four trophic state categories for the three metrics (trophic.class, trophic.class.secchi, trophic.class.tp)? Hint: `n()` function.

```
# Chl
LAGOStrophic %>%
  group_by(trophic.class) %>%
  summarise(n())
```

```
## # A tibble: 4 x 2
##   trophic.class  ‘n()‘
##   <chr>          <int>
## 1 Eutrophic      37457
## 2 Hypereutrophic 13234
## 3 Mesotrophic    13964
## 4 Oligotrophic    2762
```

```
# Secchi
LAGOStrophic %>%
  group_by(trophic.class.secchi) %>%
  summarise(n())
```

```
## # A tibble: 4 x 2
##   trophic.class.secchi ‘n()‘
##   <fct>                <int>
## 1 Oligotrophic         14559
## 2 Mesotrophic          22344
## 3 Eutrophic            25793
## 4 Hypereutrophic        4721
```

```
# Tp
LAGOStrophic %>%
  group_by(trophic.class.tp) %>%
  summarise(n())
```

```
## # A tibble: 4 x 2
##   trophic.class.tp ‘n()‘
##   <fct>            <int>
## 1 Oligotrophic     17984
## 2 Mesotrophic      20607
## 3 Eutrophic        22419
## 4 Hypereutrophic    6407
```

5. What proportion of total observations are considered eutrophic or hypereutrophic according to the three different metrics (trophic.class, trophic.class.secchi, trophic.class.tp)?

```
# Chl proportions
LAGOStrophic %>%
  group_by(trophic.class) %>%
  summarise(count = n()) %>%
  mutate(proportion = count/sum(count))
```

```
## # A tibble: 4 x 3
##   trophic.class  count proportion
##   <chr>          <int>      <dbl>
## 1 Eutrophic      37457     0.556
## 2 Hypereutrophic 13234     0.196
## 3 Mesotrophic    13964     0.207
## 4 Oligotrophic    2762     0.0410
```

```
#Eutrophic = 55.6%
#Hypereutrophic = 19.6%

# Secchi
LAGOStrophic %>%
  group_by(trophic.class.secchi) %>%
  summarise(count = n()) %>%
  mutate(proportion = count/sum(count))
```

```
## # A tibble: 4 x 3
##   trophic.class.secchi count proportion
##   <fct>                <int>      <dbl>
## 1 Oligotrophic         14559     0.216
## 2 Mesotrophic          22344     0.331
## 3 Eutrophic            25793     0.383
## 4 Hypereutrophic        4721     0.0700
```

```
#Eutrophic = 21.6%
#Hypereutrophic = 33.1%

# Tp
LAGOStrophic %>%
  group_by(trophic.class.tp) %>%
  summarise(count = n()) %>%
  mutate(proportion = count/sum(count))
```

```
## # A tibble: 4 x 3
##   trophic.class.tp count proportion
##   <fct>            <int>      <dbl>
## 1 Oligotrophic     17984     0.267
## 2 Mesotrophic      20607     0.306
## 3 Eutrophic        22419     0.333
## 4 Hypereutrophic    6407     0.0950
```

```
#Eutrophic = 26.7%
#Hypereutrophic = 30.6%
```

Which of these metrics is most conservative in its designation of eutrophic conditions? Why might this be?

> The secchi metric is most conservative in the designation of the eutrophic category specific, whereas it has the highest estimate of hypereutrophic conditions. The most overall conservative is Chl, which designates just 19.6% as hypereutrophic. This could mean that overall productivity is not as high as the secchi method suggests, perhaps indicating that there is non-algal material in the lake that is amplifying the secchi estimate. Total phosphorus serves as a limiting factor.

Nutrient Concentrations

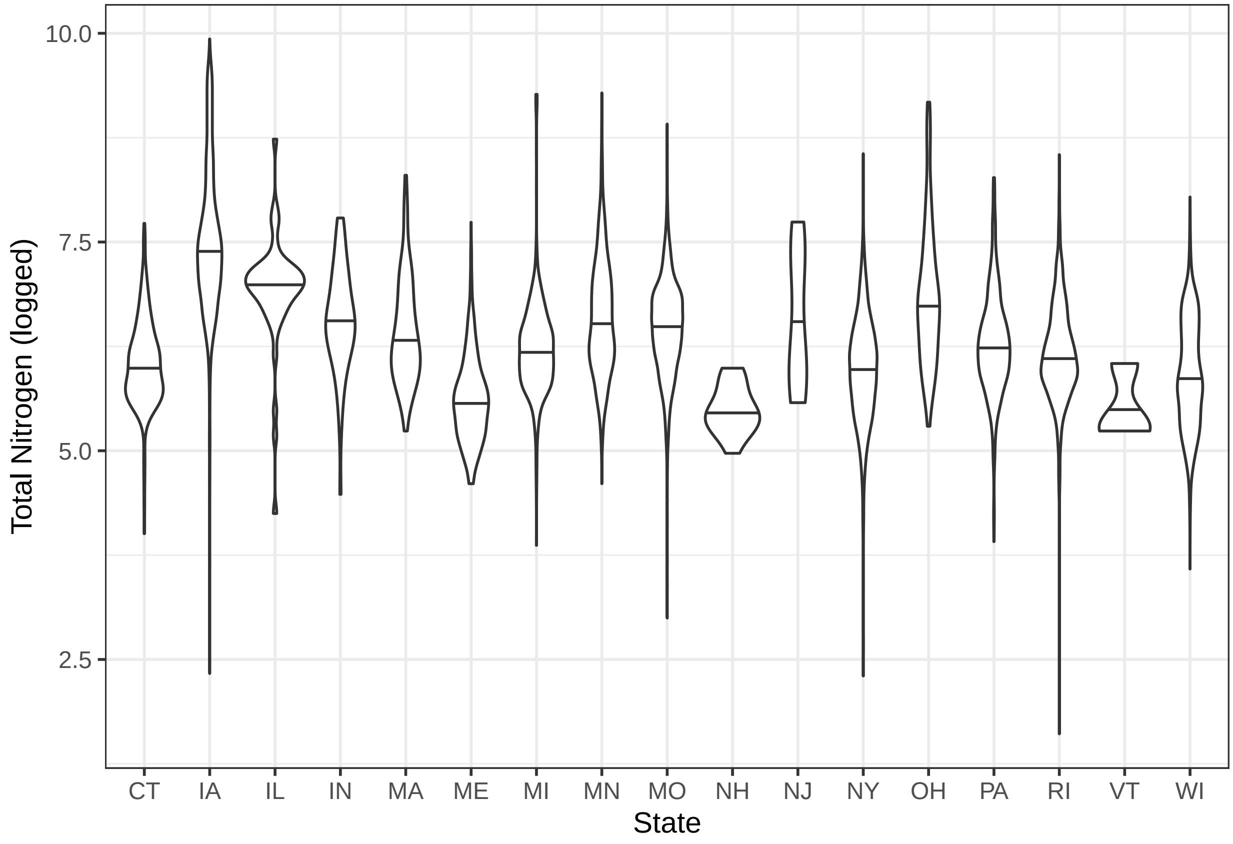
6. Create a data frame that includes the columns lagoslakeid, sampledate, tn, tp, state, and state_name. Mutate this data frame to include sampleyear and samplemonth columns as well. Filter the data frame for May-September. Call this data frame LAGOSNandP.

```
# Save needed dataframes
LAGOSnutrient <- LAGOSdata$epi_nutr
LAGOSlocus <- LAGOSdata$locus
LAGOSstate <- LAGOSdata$state
LAGOSlocations <- left_join(LAGOSlocus, LAGOSstate, by = "state_zoneid")

# Create dataframe with TN and TP
LAGOSNandP <- LAGOSnutrient %>%
  left_join(., LAGOSlocations, by = "lagoslakeid") %>%
  select(lagoslakeid, sampledate, tn, tp, state, state_name) %>%
  mutate(sampleyear = year(sampledate),
         samplemonth  = month(sampledate)) %>%
  filter(samplemonth %in% c(5:9)) %>%
  drop_na(tn, tp, state)
```

7. Create two violin plots comparing TN and TP concentrations across states. Include a 50th percentile line inside the violins. Create a logged y axis and relabel axes.

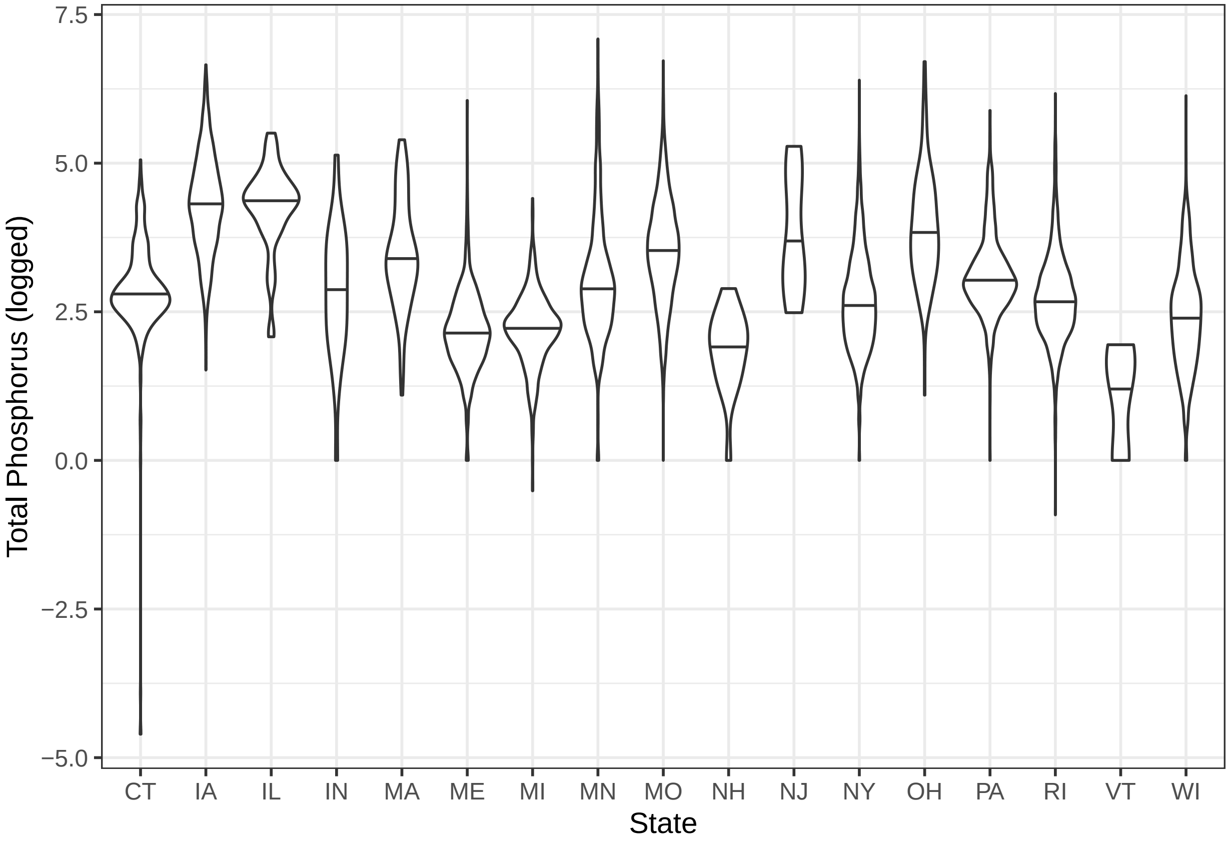
```
ggplot(LAGOSNandP) +
  geom_violin(aes(x=state, y=log(tn)), draw_quantiles = .5) +
  labs(x="State", y = "Total Nitrogen (logged)")
```

```
## Warning: Removed 7 rows containing non-finite values (stat_ydensity).

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values
```



```r
ggplot(LAGOSNandP) +
  geom_violin(aes(x=state, y=log(tp)), draw_quantiles = .5) +
  labs(x="State", y = "Total Phosphorus (logged)")
```

```
## Warning: Removed 65 rows containing non-finite values (stat_ydensity).

## Warning: collapsing to unique 'x' values
```



Which states have the highest and lowest median concentrations?

TN: Highest: Iowa. Lowest: New Hampshire.

TP: Highest: Illinois. Lowest: Vermont.

Which states have the largest and smallest concentration ranges?

TN: Largest: Iowa. Smallest: Vermont.

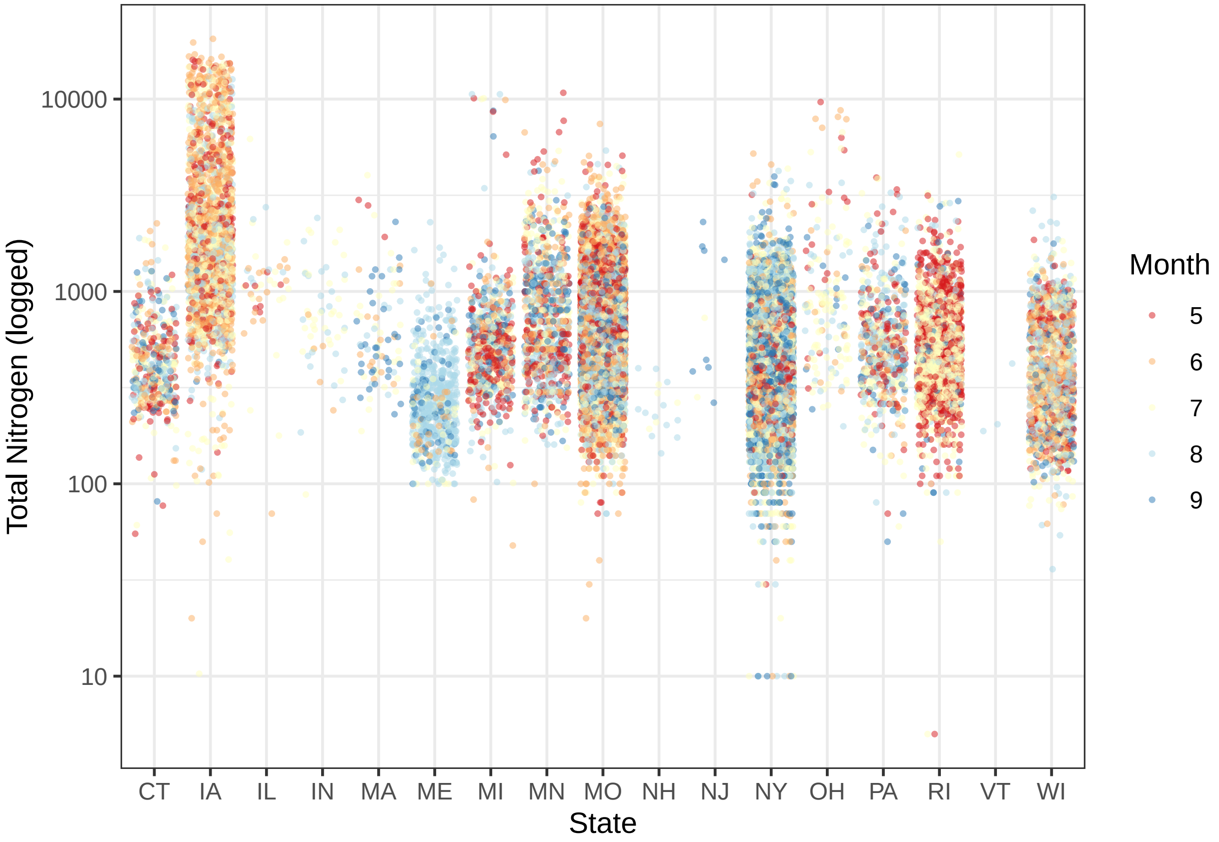
TP: Largest: Connecticut. Smallest: Vermont.

8. Create two jitter plots comparing TN and TP concentrations across states, with samplemonth as the color. Choose a color palette other than the ggplot default.

```
ggplot(LAGOSNandP, aes(x=state, y=tn, color=factor(samplemonth))) +
  geom_jitter(alpha = 0.5, size = 0.5) +
  scale_y_log10() +
  scale_color_brewer(palette="RdYlBu") +
  labs(x="State", y="Total Nitrogen (logged)", color="Month")
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

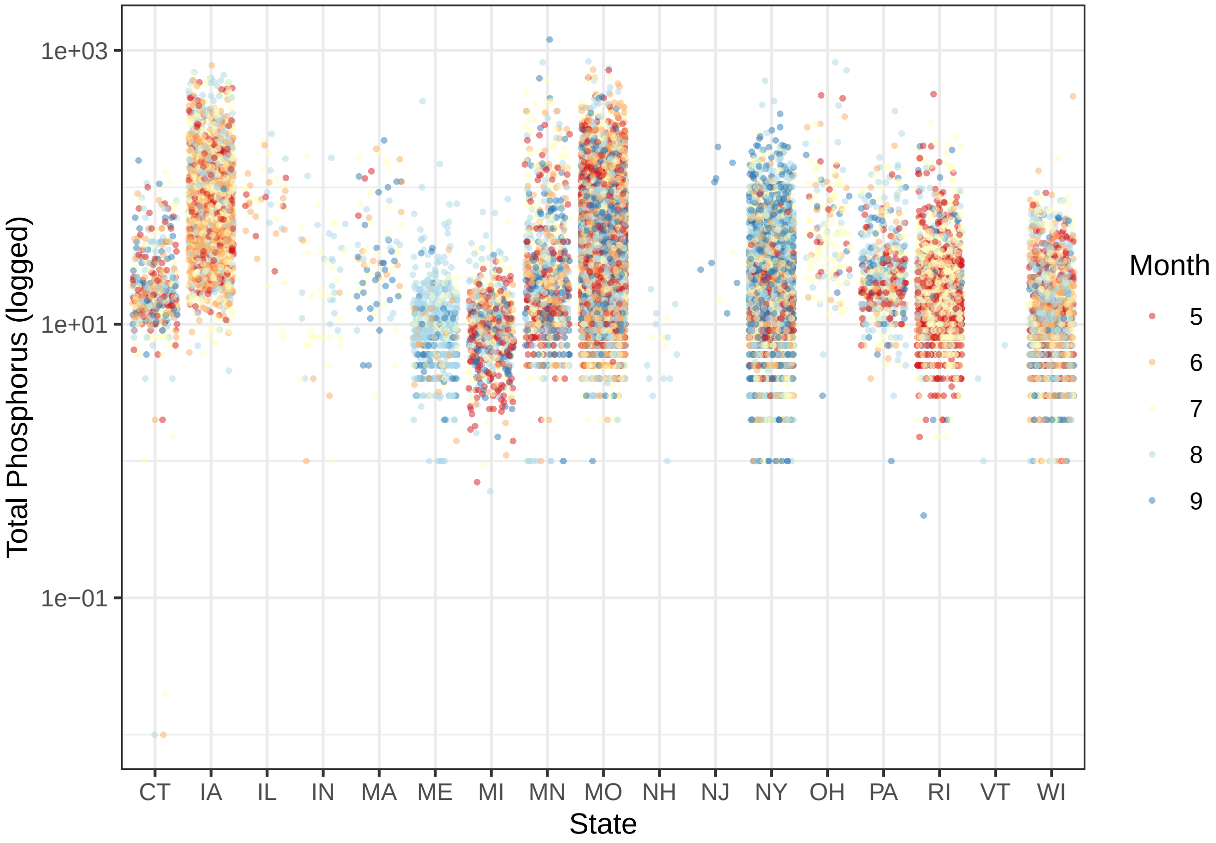
```
## Warning: Removed 7 rows containing missing values (geom_point).
```



```
ggplot(LAGOSNandP, aes(x=state, y=tp, color=factor(samplemonth))) +
  geom_jitter(alpha = 0.5, size = 0.5) +
  scale_y_log10() +
  scale_color_brewer(palette="RdYlBu") +
  labs(x="State", y="Total Phosphorus (logged)", color="Month")
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Removed 65 rows containing missing values (geom_point).
```



Which states have the most samples? How might this have impacted total ranges from #7?

TN: Iowa, New York, Montana

TP: Montana, New York, Iowa

More samples could definitely lead to a larger range of values because with a large number of samples, we could expect to see more variation.