Assignment 5: Water Quality in Lakes

Yutao Gong

## OVERVIEW

This exercise accompanies the lessons in Hydrologic Data Analysis 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 HTML file.
5. After Knitting, submit the completed exercise (HTML file) to the dropbox in Sakai. Add your last name into the file name (e.g., “A05\_Salk.html”) prior to submission.

The completed exercise is due on 2 October 2019 at 9:00 am.

## Setup

1. Verify your working directory is set to the R project file,
2. Load the tidyverse, lubridate, and LAGOSNE packages.
3. Set your ggplot theme (can be theme\_classic or something else)
4. Load the LAGOSdata database and the trophic state index csv file we created on 2019/09/27.

getwd()

## [1] "C:/Users/gongy/Documents/Hydrologic\_Data\_Analysis/Assignments"

library(tidyverse)

## -- Attaching packages ------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1 v purrr 0.3.2  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 0.8.3 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts ---------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

library(LAGOSNE)  
  
theme\_set(theme\_bw())  
LAGOStrophic = read.csv("../Data/LAGOStrophic.csv")

## Trophic State Index

1. 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).

LAGOStrophic <-   
 mutate(LAGOStrophic,   
 trophic.class.secchi =   
 ifelse(TSI.secchi < 40, "Oligotrophic",   
 ifelse(TSI.secchi < 50, "Mesotrophic",  
 ifelse(TSI.secchi < 70, "Eutrophic", "Hypereutrophic"))),  
 trophic.class.tp =   
 ifelse(TSI.tp < 40, "Oligotrophic",   
 ifelse(TSI.tp < 50, "Mesotrophic",  
 ifelse(TSI.tp < 70, "Eutrophic", "Hypereutrophic"))))  
  
LAGOStrophic$trophic.class.secchi <-   
 factor(LAGOStrophic$trophic.class.secchi,  
 levels = c("Oligotrophic", "Mesotrophic", "Eutrophic", "Hypereutrophic"))  
LAGOStrophic$trophic.class.tp <-   
 factor(LAGOStrophic$trophic.class.tp,  
 levels = c("Oligotrophic", "Mesotrophic", "Eutrophic", "Hypereutrophic"))

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

LAGOStrophic %>%  
 group\_by(trophic.class) %>%   
 summarise(Number\_of\_Observations = n()) %>%  
 mutate(Proportion\_of\_TotalObservations = Number\_of\_Observations / sum(Number\_of\_Observations))

## # A tibble: 4 x 3  
## trophic.class Number\_of\_Observations Proportion\_of\_TotalObservations  
## <fct> <int> <dbl>  
## 1 Eutrophic 41861 0.559   
## 2 Hypereutrophic 14379 0.192   
## 3 Mesotrophic 15413 0.206   
## 4 Oligotrophic 3298 0.0440

LAGOStrophic %>%  
 group\_by(trophic.class.secchi) %>%   
 summarise(Number\_of\_Observations = n()) %>%  
 mutate(Proportion\_of\_TotalObservations = Number\_of\_Observations / sum(Number\_of\_Observations))

## # A tibble: 4 x 3  
## trophic.class.secchi Number\_of\_Observations Proportion\_of\_TotalObservati~  
## <fct> <int> <dbl>  
## 1 Oligotrophic 16110 0.215   
## 2 Mesotrophic 25083 0.335   
## 3 Eutrophic 28659 0.382   
## 4 Hypereutrophic 5099 0.0680

LAGOStrophic %>%  
 group\_by(trophic.class.tp) %>%   
 summarise(Number\_of\_Observations = n()) %>%  
 mutate(Proportion\_of\_TotalObservations = Number\_of\_Observations / sum(Number\_of\_Observations))

## # A tibble: 4 x 3  
## trophic.class.tp Number\_of\_Observations Proportion\_of\_TotalObservations  
## <fct> <int> <dbl>  
## 1 Oligotrophic 19861 0.265   
## 2 Mesotrophic 23023 0.307   
## 3 Eutrophic 24839 0.331   
## 4 Hypereutrophic 7228 0.0964

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

# The proportion is shown as the third column "Proportion of Total Observations" in the last section

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

The tp metric is the most conservative in designation of eutrophic conditions. It might be because we assume phosphorus is the limiting nutrient for phytoplankton growth (especially in summer times), which therefore constraints the potential eutrophic level given a certain phosphorus level.

Note: To take this further, a researcher might determine which trophic classes are susceptible to being differently categorized by the different metrics and whether certain metrics are prone to categorizing trophic class as more or less eutrophic. This would entail more complex code.

## Nutrient Concentrations

1. 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. Call this data frame LAGOSNandP.

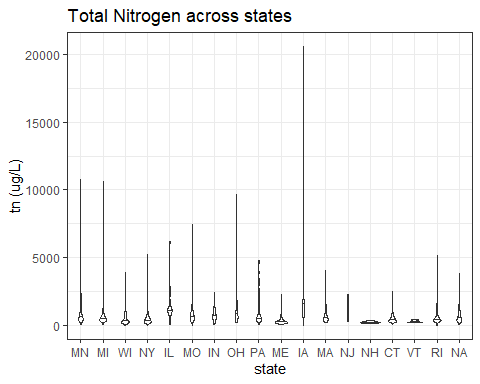
load(file = "../Data/Raw/LAGOSdata.rda")  
  
# Exploring the data types that are available  
LAGOSlocus <- LAGOSdata$locus # location  
LAGOSstate <- LAGOSdata$state  
LAGOSnutrient <- LAGOSdata$epi\_nutr  
LAGOSlocus$lagoslakeid <- as.factor(LAGOSlocus$lagoslakeid)  
LAGOSnutrient$lagoslakeid <- as.factor(LAGOSnutrient$lagoslakeid)  
  
LAGOSlocations <- left\_join(LAGOSlocus, LAGOSstate, by = "state\_zoneid")  
# by: column in common  
  
# Order by number of lakes  
LAGOSlocations <-   
 within(LAGOSlocations,   
 state <- factor(state, levels = names(sort(table(state), decreasing=TRUE))))  
LAGOSNandP <-   
 left\_join(LAGOSnutrient, LAGOSlocations, by = "lagoslakeid") %>%  
 select(lagoslakeid, sampledate, tn, tp, state, state\_name) %>%  
 mutate(sampleyear = year(sampledate),   
 samplemonth = month(sampledate)) %>%  
 drop\_na(tn:tp) # move an entire row when there's a NA

## Warning: Column `lagoslakeid` joining factors with different levels,  
## coercing to character vector

1. Create two violin plots comparing TN and TP concentrations across states. Include a 50th percentile line inside the violins.

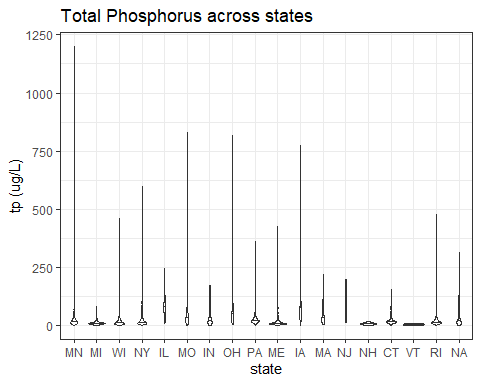
stateTNviolin <- ggplot(LAGOSNandP, aes(x = state, y = tn)) +  
 geom\_violin(draw\_quantiles = 0.50) +  
 labs(title = "Total Nitrogen across states", y = "tn (ug/L)")  
print(stateTNviolin)

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



stateTPviolin <- ggplot(LAGOSNandP, aes(x = state, y = tp)) +  
 geom\_violin(draw\_quantiles = 0.50) +  
 labs(title = "Total Phosphorus across states", y = "tp (ug/L)")  
print(stateTPviolin)

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



NitrogenMedian <- LAGOSNandP %>%   
 group\_by(state) %>%  
 summarize(median = median(tn),range = max(tn)-min(tn))

## Warning: Factor `state` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

PhosphorusMedian <- LAGOSNandP %>%   
 group\_by(state) %>%  
 summarize(median = median(tp), range = max(tp)-min(tp))

## Warning: Factor `state` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

Which states have the highest and lowest median concentrations?

TN: Highest median: Iowa, Lowest median: Vermont

TP: Highest median: Illinois, Lowest median: Vermont

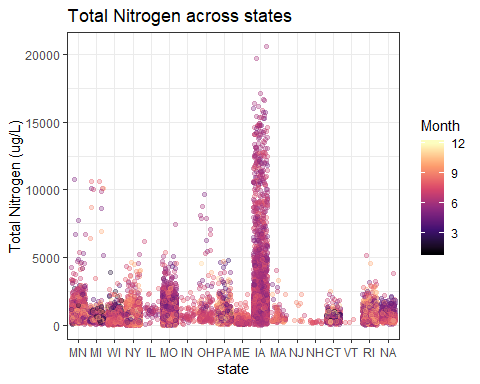
Which states have the highest and lowest concentration ranges?

TN: Highest range: Iowa, Lowest range: Vermont

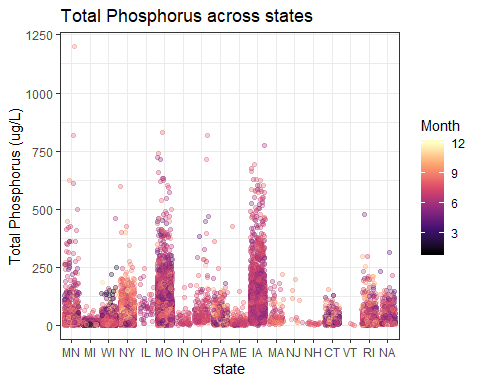
TP: Highest range: Minnesota, Lowest range: Vermont

1. 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.

stateTNjitter <- ggplot(LAGOSNandP, aes(x = state, y = tn, color = samplemonth)) +  
 geom\_jitter(alpha = 0.3) +  
 labs(title = "Total Nitrogen across states", y = "Total Nitrogen (ug/L)", color = "Month") +  
 theme(legend.position = "right") +  
 scale\_color\_viridis\_c(option = "magma")  
print(stateTNjitter)



stateTPjitter <- ggplot(LAGOSNandP, aes(x = state, y = tp, color = samplemonth)) +  
 geom\_jitter(alpha = 0.3) +  
 labs(title = "Total Phosphorus across states", y = "Total Phosphorus (ug/L)", color = "Month") +  
 theme(legend.position = "right") +  
 scale\_color\_viridis\_c(option = "magma")  
print(stateTPjitter)



LAGOSNandP %>%  
 group\_by(state) %>%   
 summarise(Number\_of\_Observations = n())

## Warning: Factor `state` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

## # A tibble: 18 x 2  
## state Number\_of\_Observations  
## <fct> <int>  
## 1 MN 1341  
## 2 MI 877  
## 3 WI 2336  
## 4 NY 7715  
## 5 IL 46  
## 6 MO 11412  
## 7 IN 57  
## 8 OH 166  
## 9 PA 983  
## 10 ME 633  
## 11 IA 2638  
## 12 MA 93  
## 13 NJ 10  
## 14 NH 17  
## 15 CT 636  
## 16 VT 3  
## 17 RI 2753  
## 18 <NA> 428

LAGOSNandP %>%  
 group\_by(samplemonth, state) %>%   
 summarise(Number\_of\_Observations = n())

## Warning: Factor `state` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

## # A tibble: 132 x 3  
## # Groups: samplemonth [12]  
## samplemonth state Number\_of\_Observations  
## <dbl> <fct> <int>  
## 1 1 WI 146  
## 2 1 MO 2  
## 3 1 PA 10  
## 4 1 CT 5  
## 5 1 <NA> 1  
## 6 2 MN 10  
## 7 2 MI 11  
## 8 2 WI 147  
## 9 2 PA 13  
## 10 2 CT 1  
## # ... with 122 more rows

Which states have the most samples? How might this have impacted total ranges from #9?

TN: If we only look at the graph we may think that Iowa has the most samples, but if we count observations by state (since all rows with NA in EITHER tn OR tp have been dropped when wrangling with data, number of observations of tn should equal that of tp), we will find it’s Missouri.

TP: Missouri

Impact: The state with the fewest observations (Vermont) has the lowest median and range, possibly just because not enough data are collected.

Which months are sampled most extensively? Does this differ among states?

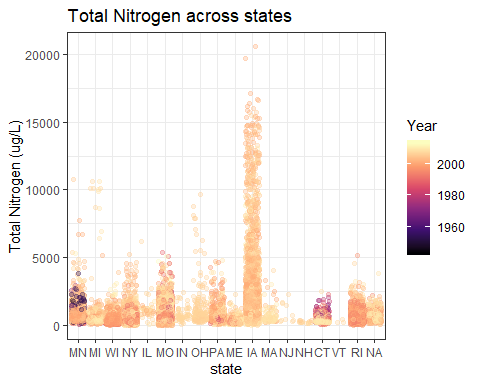
TN: June, July, August

TP: June, July, August

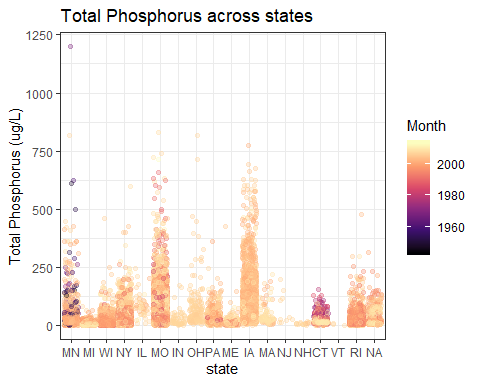
It differs among states - for example, Wisconsin only samples in October, November and December.

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

stateTNjitter.year <- ggplot(LAGOSNandP, aes(x = state, y = tn, color = sampleyear)) +  
 geom\_jitter(alpha = 0.3) +  
 labs(title = "Total Nitrogen across states", y = "Total Nitrogen (ug/L)", color = "Year") +  
 theme(legend.position = "right") +  
 scale\_color\_viridis\_c(option = "magma")  
print(stateTNjitter.year)



stateTPjitter.year <- ggplot(LAGOSNandP, aes(x = state, y = tp, color = sampleyear)) +  
 geom\_jitter(alpha = 0.3) +  
 labs(title = "Total Phosphorus across states", y = "Total Phosphorus (ug/L)", color = "Month") +  
 theme(legend.position = "right") +  
 scale\_color\_viridis\_c(option = "magma")  
print(stateTPjitter.year)



LAGOSNandP %>%  
 group\_by(sampleyear, state) %>%   
 summarise(Number\_of\_Observations = n())

## Warning: Factor `state` contains implicit NA, consider using  
## `forcats::fct\_explicit\_na`

## # A tibble: 306 x 3  
## # Groups: sampleyear [62]  
## sampleyear state Number\_of\_Observations  
## <dbl> <fct> <int>  
## 1 1944 MN 1  
## 2 1945 MN 2  
## 3 1946 MN 3  
## 4 1947 MN 2  
## 5 1948 MN 2  
## 6 1949 MN 5  
## 7 1950 MN 1  
## 8 1953 MN 8  
## 9 1954 MN 6  
## 10 1955 MN 14  
## # ... with 296 more rows

Which years are sampled most extensively? Does this differ among states?

TN: 2007

TP: 2007

Yes. For example, for New York it is 2008 that is sampled most extensively, and for Missouri it is 2006.

## Reflection

1. What are 2-3 conclusions or summary points about lake water quality you learned through your analysis?
2. Using different metrics we would make different evaluations of a lake’s eutriphication state.
3. Summer time is really the peak time of eutriphication.
4. What data, visualizations, and/or models supported your conclusions from 12?

The data on different tn, tp levels across months and eutriphication classes using different metrics.

1. Did hands-on data analysis impact your learning about water quality relative to a theory-based lesson? If so, how?

Yes, it makes the process of exploration also part of the learning.

1. How did the real-world data compare with your expectations from theory?

It is more messy - so many NAs!