10: Water Quality in Lakes

Hydrologic Data Analysis | Kateri Salk

Fall 2019

Lesson Objectives

1. Navigate and explore the LAGOSNE database and R package
2. Predict nitrogen and phosphorus concentrations in lakes using landscape-scale factors
3. Analyze spatial and temporal patterns of water quality across the northeast U.S.

Opening Discussion

Nutrient loading is one of the most widespread water quality issues in lakes. Eutrophication leads to further issues such as harmful algal blooms, food web alterations, and hypoxia. Nitrogen (N) and phosphorus (P) are the main focus of nutrient loading studies and management, as they are the primary limiting nutrients for phytoplankton (algae and other single-celled primary producers) growth. Today, we will explore the LAGOS-NE dataset to find predictor variables for total N (TN) and total P (TP) concentrations in lakes.

Session Set Up

```
getwd()
```

```
## [1] "/Users/katerisalk/Box Sync/Courses/Hydrologic Data Analysis/Lessons"
```

```
library(tidyverse)
library(lubridate)
#install.packages("LAGOSNE")
library(LAGOSNE)
#install.packages("corrplot")
library(corrplot)
#install.packages("car")
library(car)

theme_set(theme_classic())
options(scipen = 100)

# Load LAGOSNE data into R session
LAGOSdata <- lagosne_load()
```

```
## Warning in `_f`(version = version, fpath = fpath): LAGOSNE version
## unspecified, loading version: 1.087.3
```

```
# If the lagosne_get function has not worked, use this code:
# load(file = "./Data/Raw/LAGOSdata.rda")

# What types of data are available in the database?
names(LAGOSdata)
```

```
##  [1] "county"               "county.chag"          "county.conn"
##  [4] "county.lulc"          "edu"                  "edu.chag"
```

```
##  [7] "edu.conn"             "edu.lulc"             "hu4"
## [10] "hu4.chag"             "hu4.conn"             "hu4.lulc"
## [13] "hu8"                  "hu8.chag"             "hu8.conn"
## [16] "hu8.lulc"             "hu12"                 "hu12.chag"
## [19] "hu12.conn"            "hu12.lulc"            "iws"
## [22] "iws.conn"             "iws.lulc"             "state"
## [25] "state.chag"           "state.conn"           "state.lulc"
## [28] "buffer100m"           "buffer100m.lulc"      "buffer500m"
## [31] "buffer500m.conn"      "buffer500m.lulc"      "lakes.geo"
## [34] "epi_nutr"             "lakes_limno"          "lagos_source_program"
## [37] "locus"
```

Predicting TN and TP concentrations

In your group, look through the LAGOS-NE database and decide on 5 variables available in the database that may predict N and P concentrations in lakes. Remembering that hypotheses differ from predictions in that they propose a mechanism, make hypotheses about your choices in predictor variables and how they might affect N and P concentrations.

Variable 1:

Variable 2:

Variable 3:

Variable 4

Variable 5:

Wrangle LAGOSdata so that you have a new data frame that includes TN, TP, your five variables, and any other columns that would be useful for you (e.g., lake id numbers). Helpful functions may include `join`, `select`, `filter`, and `mutate`. Consider whether you want to use `drop_na` to retain complete cases of individual variables or entire rows.

```
# Wrangle data here.
```

Examine potential correlations among your predictor variables. If your predictor variables are highly correlated with each other, your model will suffer from *multicollinearity,* essentially that two or more of your predictor variables provide redundant information. Model fit and accuracy of model coefficients suffer when multicollinearity is present.

1. Create a correlation plot for your 7 variables of interest (TN, TP, and the five predictor variables). What patterns do you see? (hint: function `corrplot` in the corrplot package. I like to use `upper = "ellipse"` inside the function).

What patterns do you see?

2. Create two linear regression models (hint: function `lm`) to predict TN and TP using your five predictor variables.

How much variance in TN and TP is accounted for by your full model?

3. Calculate variance inflation factors (VIF) for each of the variables in both models (hint: `vif` function in the car package). VIF values exceeding 5-10 indicates an issue with multicollinearity for that variable. Note: you should evaluate VIF values separately for each model.

Which variable(s) might you choose to remove from each model based on VIF values?

4. With your revised list of predictor variables (you may have chosen to remove some variables from consideration), create a model that optimizes explanatory power and simplicity. Remember, it is possible to over-parameterize a linear model, when fewer variables might be more appropriate. To help with this tradeoff, we can use the **Akaike's Information Criterion (AIC)** to compute a stepwise regression that either adds explanatory variables from the bottom up or removes explanatory variables from a full set of suggested options. The smaller the AIC value, the better.

To create a stepwise regression, first create a model with the revised list of predictor variables, and then use the function `step(modelname)` to calculate AIC values and choose the most parsimonous model. Do this for both models (TN and TP).

5. Based on the AIC analysis, create two final models predicting TN and TP concentrations with the combination of predictor variables recommended by the AIC analysis.

Interpret the effect of each variable on TN or TP (e.g., "TN increases by one unit with every z unit increase/decrease in x predictor variable")

How much variance in TN and TP do your models account for?