Assignment 10: Data Scraping

Iman Byndloss

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on data scraping.

Directions

1. Rename this file `<FirstLast>_A10_DataScraping.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

1. Set up your session:

- Load the packages `tidyverse`, `rvest`, and any others you end up using.
- Check your working directory

```r
#1 library() calls relevant packages and here() checks working directory
library(tidyverse)
library(rvest)
library(here)
library(ggplot2)

here()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2023 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- Scroll down and select the LWSP link next to Durham Municipality.
- Note the web address: https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2023

Indicate this website as the as the URL to be scraped. (In other words, read the contents into an `rvest` webpage object.)

```r
#2 Read the resources from the web address into a webpage object
webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2023')
```

3. The data we want to collect are listed below:

- From the "1. System Information" section:
- Water system name
- PWSID
- Ownership
- From the "3. Water Supply Sources" section:
- Maximum Day Use (MGD) - for each month

In the code chunk below scrape these values, assigning them to four separate variables.

> HINT: The first value should be "Durham", the second "03-32-010", the third "Municipality", and the last should be a vector of 12 numeric values (represented as strings)“.

```r
#3 Requires 1) using SelectorGadget Google Chrome extension,
# 2) selecting relevant data to retrieve a unique identifier,
# and 3) assigning the data to a variable in RStudio
# From the "1. System Information" section:
WaterSystem <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()

PWSID <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()

Ownership <- webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()


# From the "3. Water Supply Sources" section:
# Also, scrape the months to simplify the next step
MGD <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()

Month <- webpage %>%
  html_nodes(".fancy-table:nth-child(31) tr+ tr th") %>%
  html_text()
```

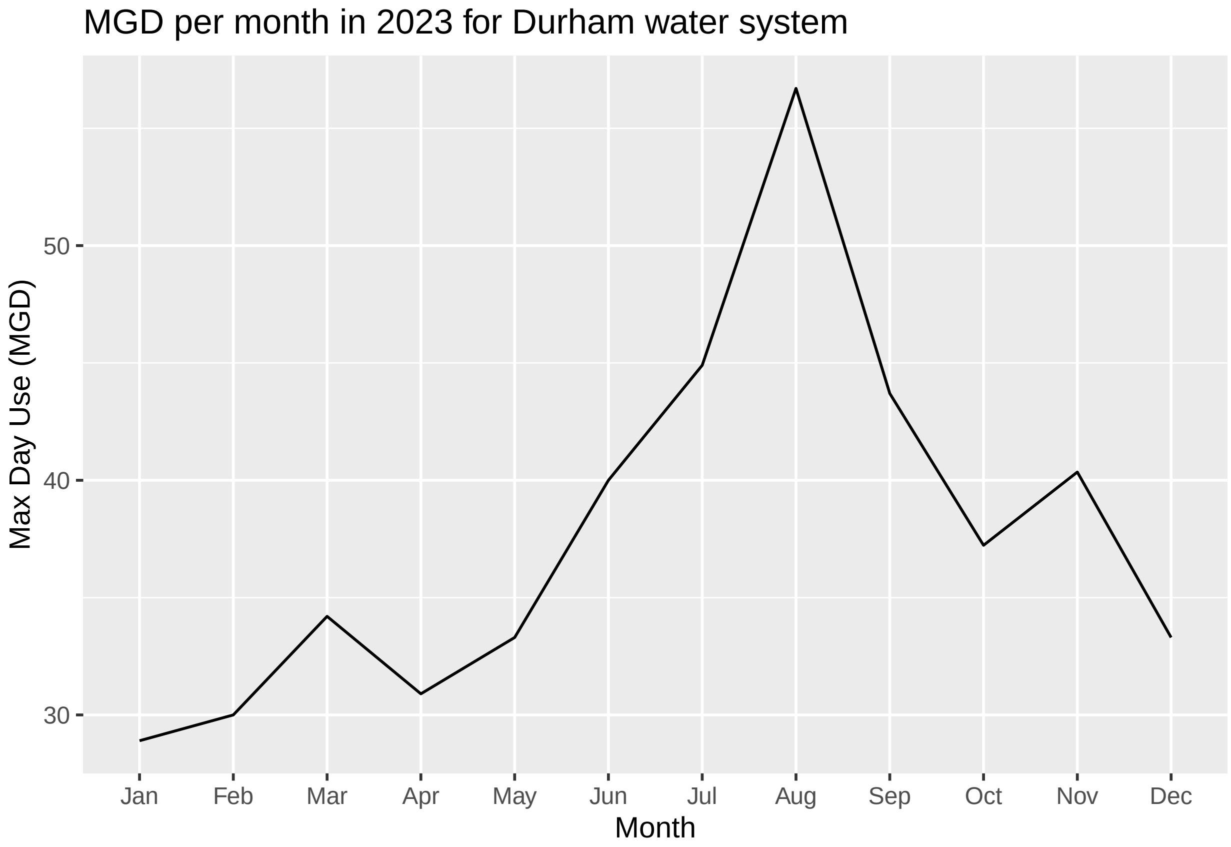
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

TIP: Use `rep()` to repeat a value when creating a dataframe.

NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc… Or, you could scrape month values from the web page…

5. Create a line plot of the maximum daily withdrawals across the months for 2023, making sure, the months are presented in proper sequence.

```r
#4 Use data.frame() to create a new data frame with the relevant variables
LWSP_Durham2023 <- data.frame("WaterSystem" = rep(WaterSystem, 12),
                              "PWSID" = rep(PWSID, 12),
                              "Ownership" = rep(Ownership, 12),
                              Month,
                              "Date"=paste(Month, 2023, sep = "-"),
                              "MGD" = as.numeric(MGD))

#5 Use ggplot() to create a line graph of MGD per Month in 2023 for Durham
# Also, order the month column with factor() before graphing
LWSP_Durham2023$Month <- factor(LWSP_Durham2023$Month, levels = c("Jan", "Feb",
                                                                   "Mar", "Apr",
                                                                   "May", "Jun",
                                                                   "Jul", "Aug",
                                                                   "Sep", "Oct",
                                                                   "Nov", "Dec"))

ggplot(LWSP_Durham2023, aes(x = Month, y = MGD, group = 1)) +
  # group=1 ensures all months connected by a single line
  geom_line() +
  labs(title = "MGD per month in 2023 for Durham water system",
       x = "Month",
       y = "Max Day Use (MGD)")
```

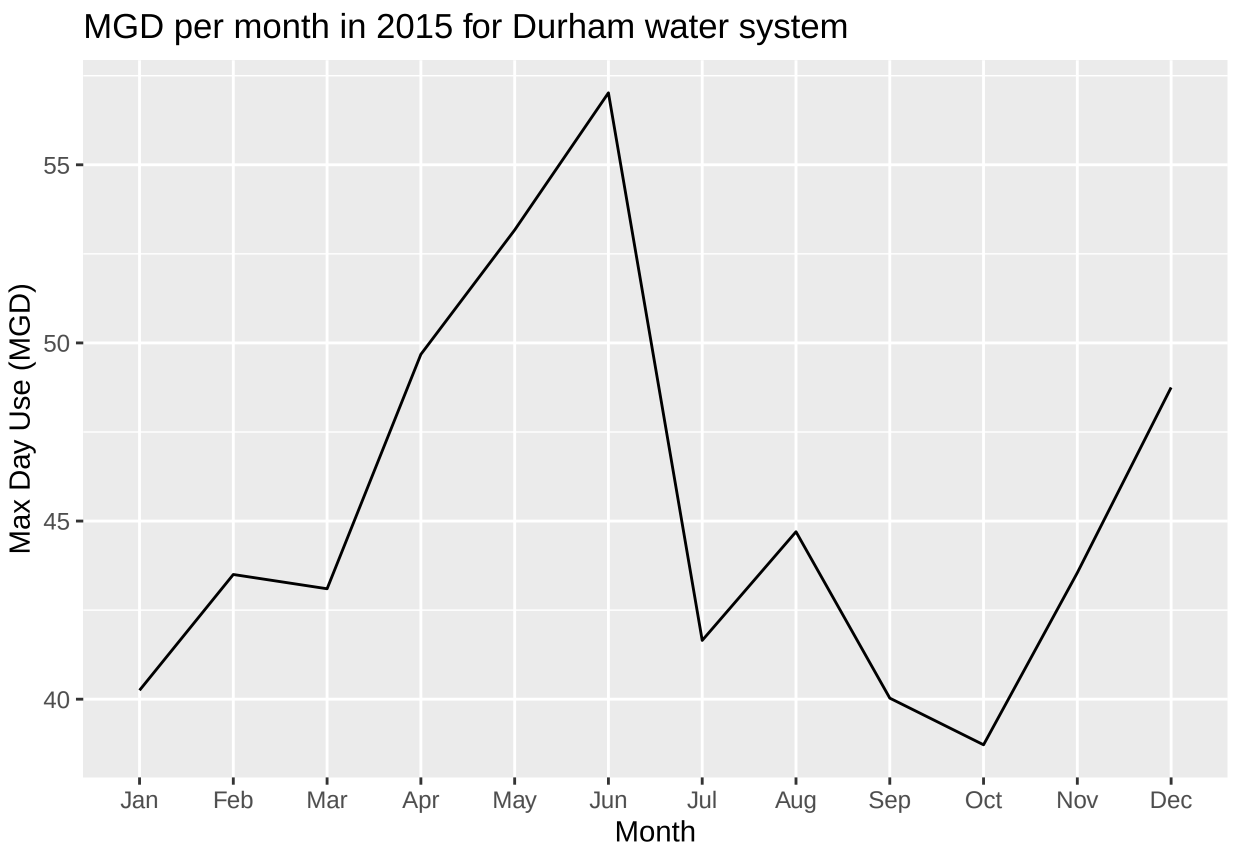


6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data, returning a dataframe. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```r
#6 Create our scraping function
scrape.it <- function(PWSID, Year){

  # Retrieve the website contents
  webpage <- read_html(
    paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=', PWSID,
           '&year=', Year))

  # Set the element address variables (determined in the previous step)
  WaterSystem_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
  PWSID_tag <- 'td tr:nth-child(1) td:nth-child(5)'
  Ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
  MGD_tag <- 'th~ td+ td'

  # Scrape the data items
  WaterSystem <- webpage %>% html_nodes(WaterSystem_tag) %>% html_text()
  PWSID <- webpage %>%   html_nodes(PWSID_tag) %>%  html_text()
  Ownership <- webpage %>% html_nodes(Ownership_tag) %>% html_text()
  MGD <- webpage %>% html_nodes(MGD_tag) %>% html_text()
  # Convert to a data frame
  dfLWSP <- data.frame("WaterSystem" = rep(WaterSystem, 12),
                       "PWSID" = rep(PWSID, 12),
                       "Ownership" = rep(Ownership, 12),
                       "Month" = c("Jan", "May", "Sep", "Feb", "Jun", "Oct",
                                   "Mar", "Jul", "Nov", "Apr", "Aug", "Dec"),
                       "Year" = rep(Year, 12),
                       "Date"=as.Date(paste("01", Month, Year, sep = "-"), format = "%d-%b-%Y"),
                       "MGD" = as.numeric(MGD))

  dfLWSP$Month <- factor(dfLWSP$Month, levels = c("Jan", "Feb", "Mar", "Apr",
                                                  "May", "Jun", "Jul", "Aug",
                                                  "Sep", "Oct", "Nov", "Dec"))

  # Return the data frame
  return(dfLWSP)
}
```

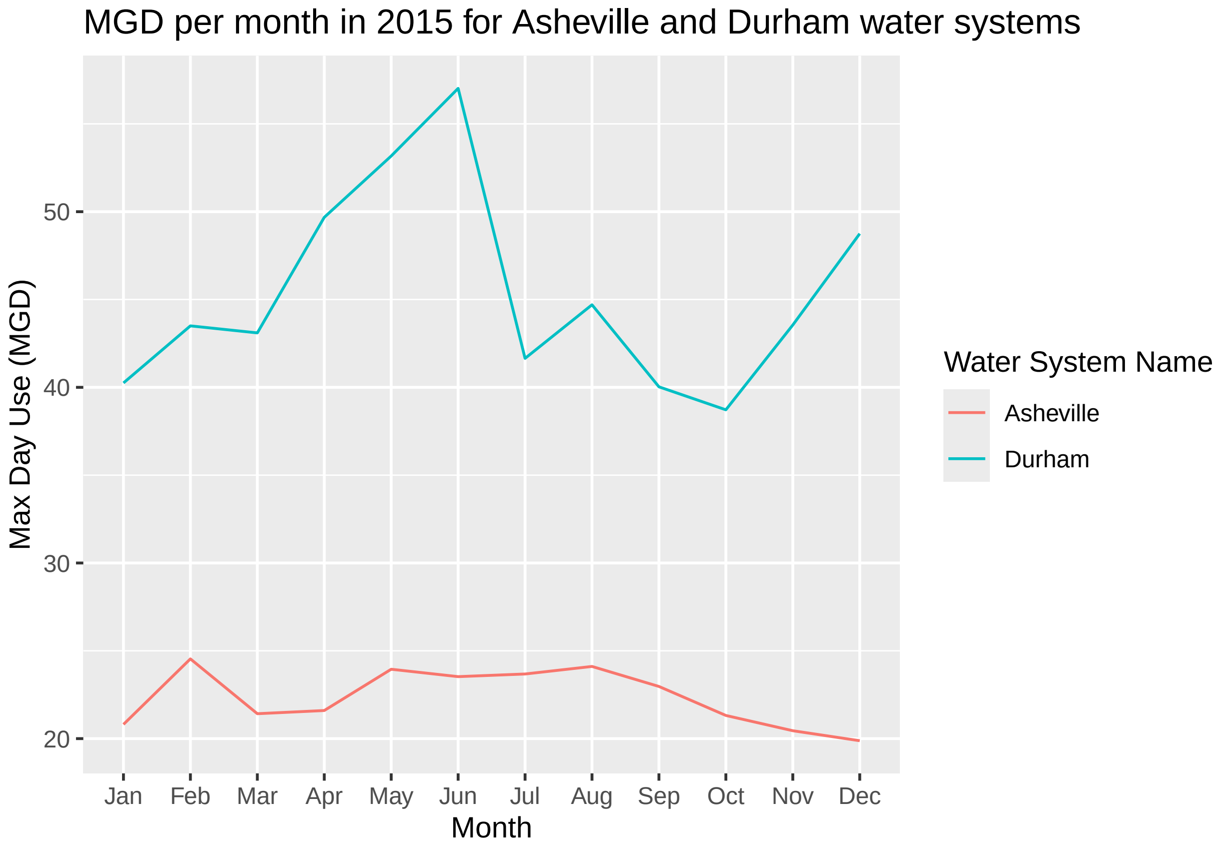
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID='03-32-010') for each month in 2015

```
#7 Run the function for Durham in 2015
LWSP_Durham2015 <- scrape.it('03-32-010', 2015)
view(LWSP_Durham2015)

#5 Use ggplot() to create a line graph of MGD per Month in 2015 for Durham
ggplot(LWSP_Durham2015, aes(x = Month, y = MGD, group = 1)) +
  # group=1 ensures all months connected by a single line
  geom_line() +
  labs(title = "MGD per month in 2015 for Durham water system",
       x = "Month",
       y = "Max Day Use (MGD)")
```



8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8 Run the function for Asheville in 2015
LWSP_Asheville2015 <- scrape.it('01-11-010', 2015)
view(LWSP_Asheville2015)

# Combine the Durham and Ashville data frames with rbind()
DurhamAshville2015 <- rbind(LWSP_Durham2015, LWSP_Asheville2015)

# Use ggplot() to create a line graph that compares Durham and Asheville
ggplot(DurhamAshville2015, aes(x = Month, y = MGD, color = WaterSystem, group = WaterSystem)) +
  geom_line() +
  labs(title = "MGD per month in 2015 for Asheville and Durham water systems",
       x = "Month",
       y = "Max Day Use (MGD)",
       color = "Water System Name")
```

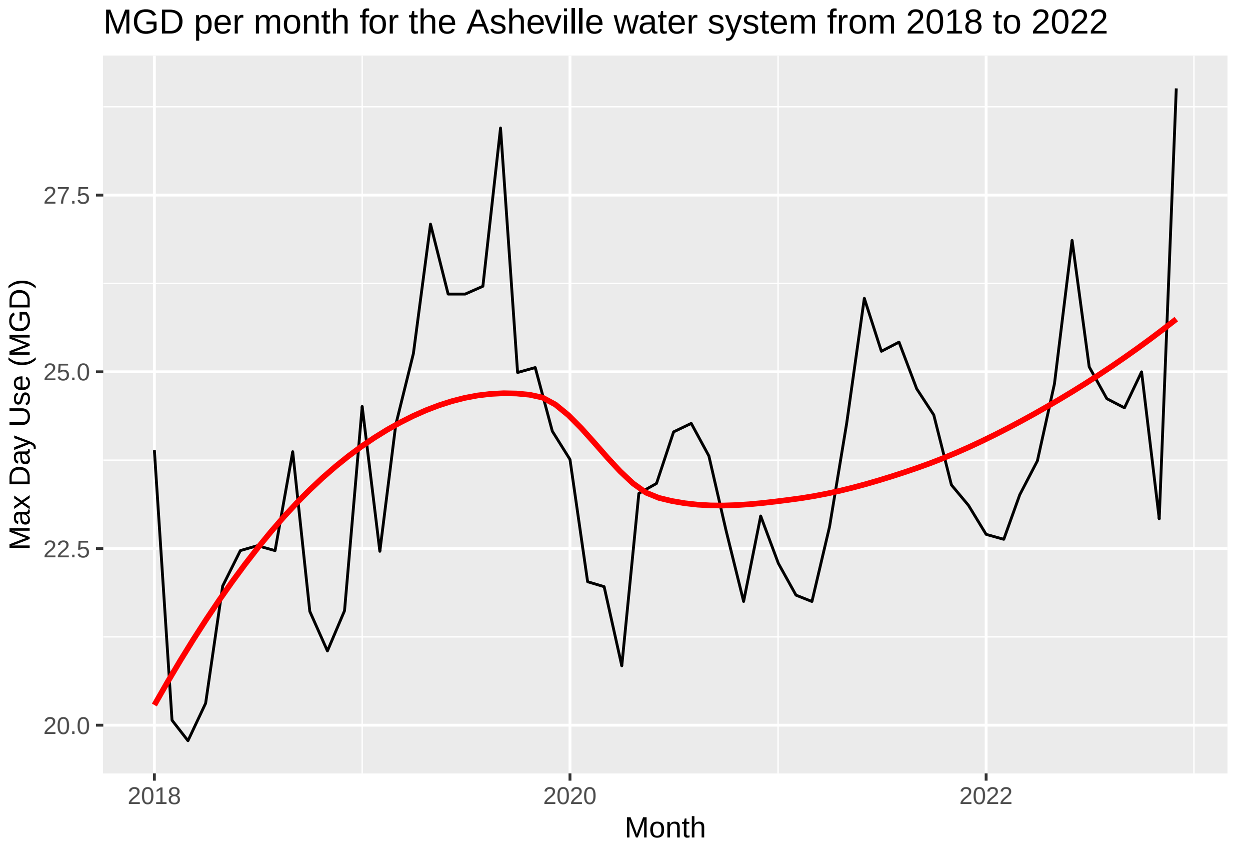


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2022. Add a smoothed line to the plot (method = 'loess').

   TIP: See Section 3.2 in the "10_Data_Scraping.Rmd" where we apply "map2()" to iteratively run a function over two inputs. Pipe the output of the map2() function to `bindrows()` to combine the dataframes into a single one.

```
#9 Map the function to scrape data from 2018 to 2022 for Asheville
Asheville2018to2022 <- seq(2018,2022) %>% map(~ scrape.it("01-11-010", .)) %>% bind_rows()

# Use ggplot() to create a line graph that compares Durham and Asheville
ggplot(Asheville2018to2022, aes(x = Date, y = MGD)) +
  geom_line(color = "black") +
  geom_smooth(method = "loess", se = FALSE, color = "red") +
  labs(title = "MGD per month for the Asheville water system from 2018 to 2022",
       x = "Month",
       y = "Max Day Use (MGD)",
       color = "Year")
```

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



Question: Just by looking at the plot (i.e. not running statistics), does Asheville have a trend in water usage over time? > Answer: > Almost all records of water usage in Asheville from 2018 to 2022 are in the 20s MGD. Generally speaking, according to the trend line, water usage in Asheville is increasing over time. To be more specific, water usage increased by around 5 MGD from 2018 to 2022, but it was not a continuous rise as a slight decrease occurred from 2020 to 2021.