Assignment 10: Data Scraping

Abby Liu

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

```
#1
#Install packages
library(tidyverse)
library(rvest)
library(lubridate)
library(dplyr)

#Check working directory
getwd()
```

```
## [1] "/home/guest/EDA-Spring2023"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2022 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=2022

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

```
#2
theURL <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
theURL
```

```
## {html_document}
## <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
## [1] <head>\n<title>DWR :: Local Water Supply Planning</title>\n<meta http-equ ...
## [2] <body id="plan">\r\n<!--<div id="division-header">\r\n<a name="top" href= ...
```

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), with the first value being "36.1000".

```
#3
water.system.name <- theURL %>%
html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>% html_text()

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

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

max.withdrawals.mgd <- theURL %>% html_nodes("th~ td+ td") %>% html_text()
```

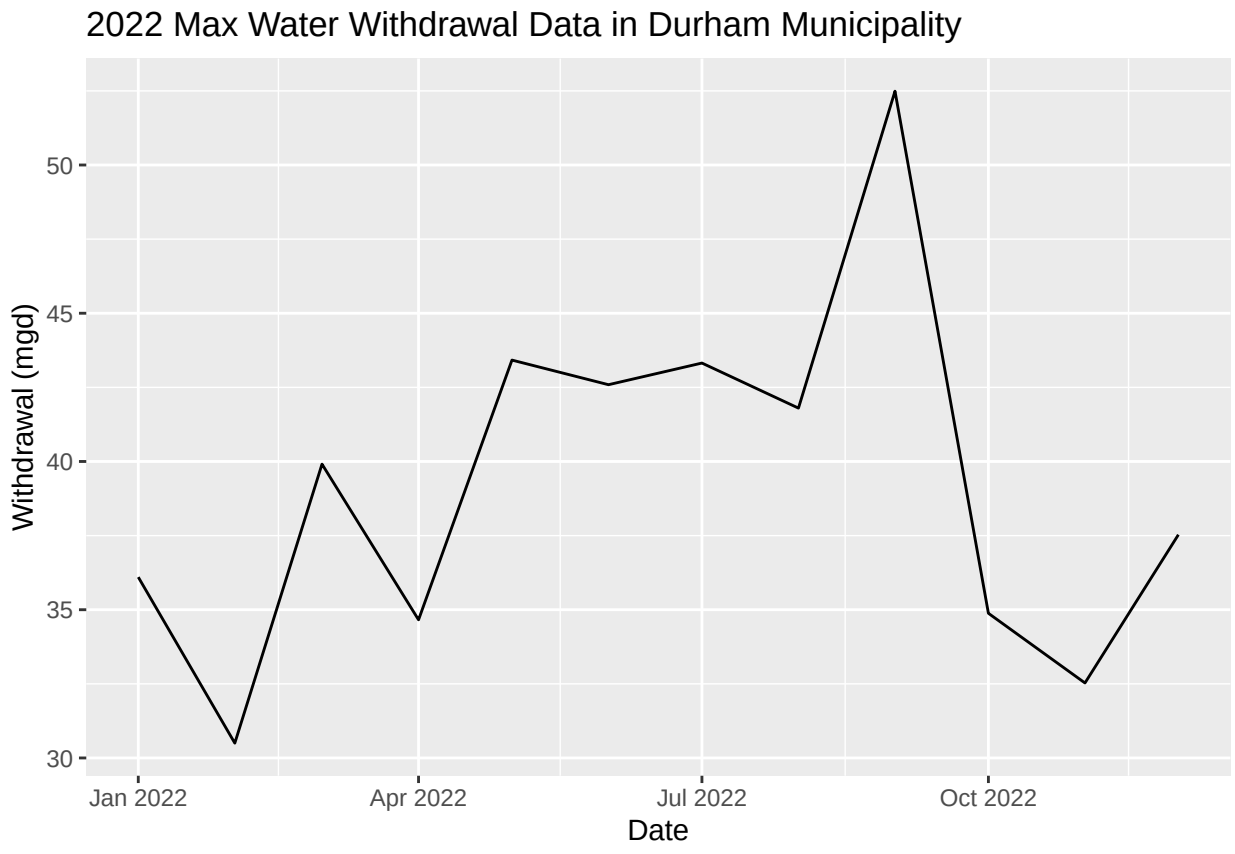
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   > 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 average daily withdrawals across the months for 2022

```
#4
month <- theURL %>% html_nodes(".fancy-table:nth-child(31) tr+ tr th") %>% html_text()

df_withdrawals <- data.frame("Year" = rep(2022,12),
                             "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd))
df_withdrawals <- df_withdrawals %>%
  mutate(Water_System_Name = !!water.system.name,
         PSWID = !!PWSID,
         Ownership = !!ownership,
         Month = !!month,
         Date = my(paste(Month,"-",Year))) %>%
  arrange(ymd(Date))

#5
ggplot(df_withdrawals,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  labs(title = paste("2022 Max Water Withdrawal Data in Durham Municipality"),
       y="Withdrawal (mgd)",
       x="Date")
```

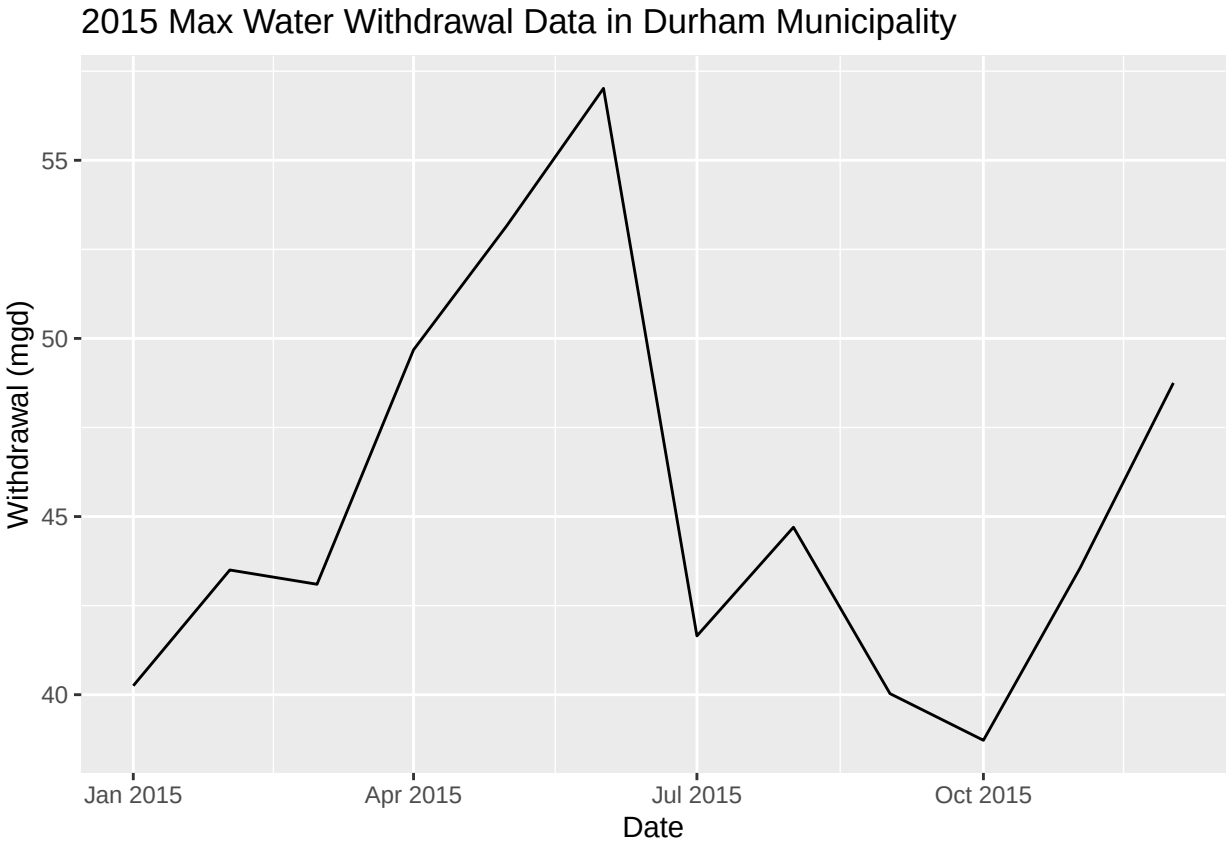


6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6.
# Assign initial values to variables
the_main_url <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid='
my_PWSID <- '03-63-020'
my_Year <- '2020'
my_month <- c(1,5,9,2,6,10,3,7,11,4,8,12)

my_scrape <- function(my_PWSID, my_Year){
# Data scraping
  my_url <- read_html(paste0(the_main_url, my_PWSID, '&year=', my_Year))
  my_water.system.name <- my_url %>% html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>% html_
  my_PWSID <- my_url %>% html_nodes("td tr:nth-child(1) td:nth-child(5)") %>% html_text()
  my_ownership <- my_url %>% html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>% html_text()
  my_max.withdrawals.mgd <- my_url %>% html_nodes("th~ td+ td") %>% html_text()

# Construct the dataframe
  df_water <- data.frame("Month" = my_month,
                         "Year" = rep(my_Year, 12),
                         "Max_Withdrawals_mgd" = as.numeric(my_max.withdrawals.mgd)) %>%
  mutate(Water_System_Name = !!my_water.system.name,
         PWSID = !!my_PWSID,
         Ownership = !!my_ownership,
         Date = my(paste(Month,"-",Year))) %>%
  arrange(ymd(Date))
  return(df_water) }
```

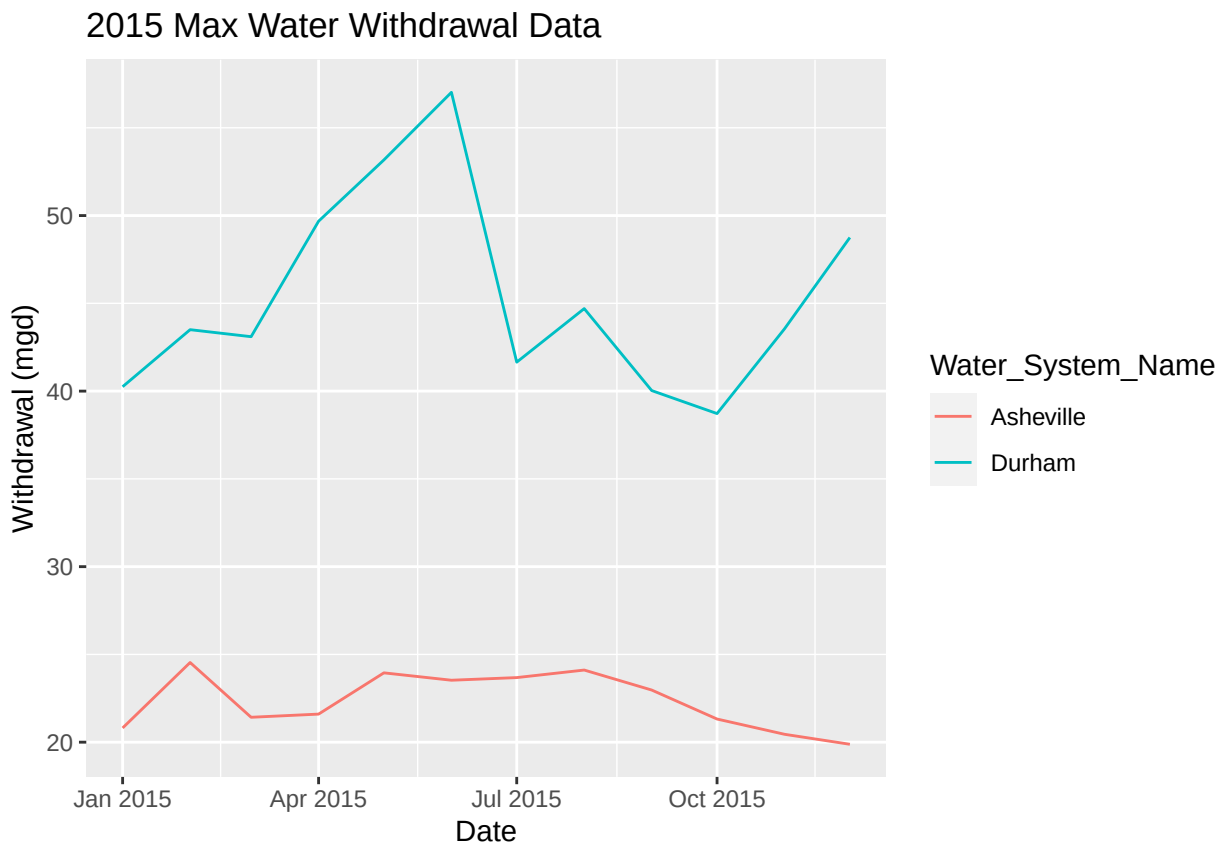
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

```
#7
my_PWSID <- '03-32-010'
my_Year <- '2015'
df_durham_2015 <- my_scrape(my_PWSID, my_Year)

ggplot(df_durham_2015,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  labs(title = paste("2015 Max Water Withdrawal Data in Durham Municipality"),
       y="Withdrawal (mgd)",
       x="Date")
```



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
my_PWSID <- '01-11-010'
my_Year <- '2015'
df_ash_2015 <- my_scrape(my_PWSID, my_Year)

df_combined <- rbind(df_durham_2015, df_ash_2015)

ggplot(df_combined,aes(x=Date,y=Max_Withdrawals_mgd, color=Water_System_Name)) +
  geom_line() +
  labs(title = paste("2015 Max Water Withdrawal Data"),
       y="Withdrawal (mgd)",
       x="Date")
```

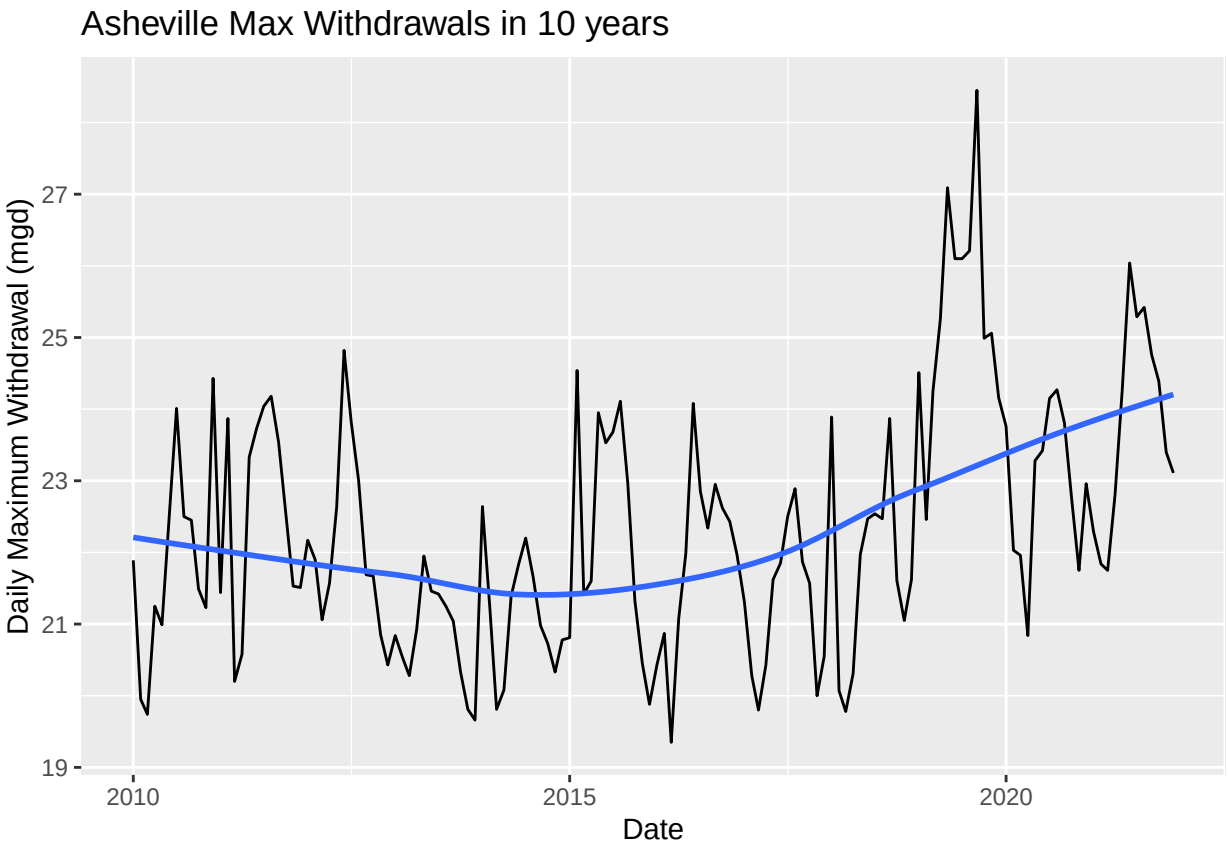


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   TIP: See Section 3.2 in the "09_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
the_years = c(2010:2021)

ash_dfs <- map(the_years, my_scrape, my_PWSID='01-11-010')
asheville <- bind_rows(ash_dfs)

ggplot(asheville,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("Asheville Max Withdrawals in 10 years"),
       y="Daily Maximum Withdrawal (mgd)",
       x="Date")
```

```
## `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? From the year 2010 to 2015, there's a very slight declining trend in water usage in Asheville. After the year 2015, there's a very apparent increasing trend.