Assignment 10: Data Scraping

Yue Zhang

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
library(tidyverse)
library(rvest)
library(lubridate)
library(ggthemes)
getwd()
```

```
## [1] "E:/MCRP/Spring2023/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.)

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

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 "27.6400".

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

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

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

max.withdrawals.mgd = the_website %>%
    html_nodes("th~ td+ td") %>%
    html_text()
```

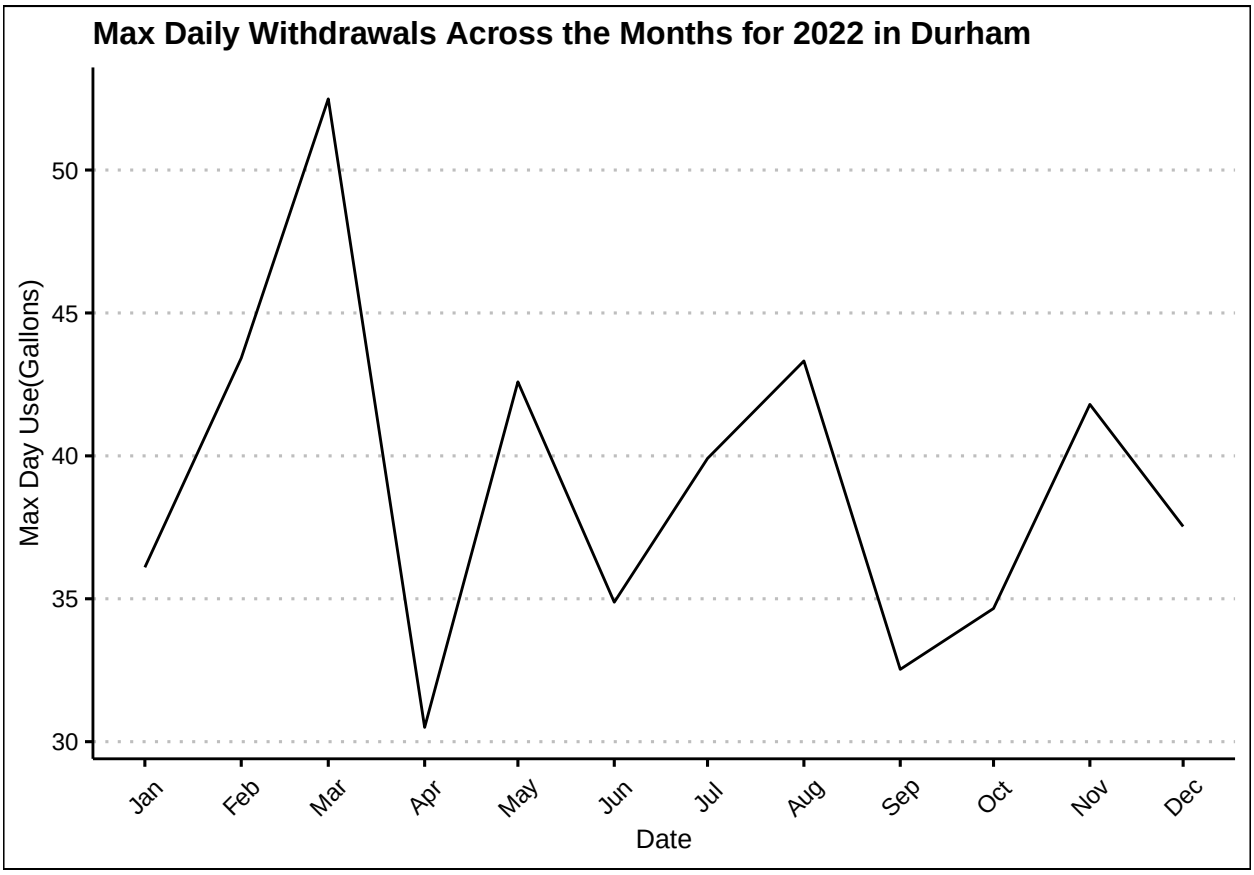
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   > 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 max daily withdrawals across the months for 2022

```r
# 4
withdrawal_df = data.frame(Month = rep(1:12),
    `Water System Name` = water.system.name,
    Ownership = ownership, PWSID = PWSID,
    MGD = as.numeric(max.withdrawals.mgd))
withdrawal_df = withdrawal_df %>%
    mutate(Date = ym(paste0("2022", "-",
        Month)))

# 5
mytheme = theme_clean(base_size = 12) + theme(axis.text = element_text(color = "black"),
    legend.position = "right", axis.text.x = element_text(angle = 45,
        vjust = 0.5, hjust = 0.5), plot.title = element_text(size = 12))
theme_set(mytheme)
MGD_plot = ggplot(withdrawal_df) + geom_line(aes(x = Date,
    y = MGD)) + ylab("Max Day Use(Gallons)") +
    xlab("Date") + labs(title = "Max Daily Withdrawals Across the Months for 2022 in Durham") +
    scale_x_date(date_breaks = "1 month",
        date_labels = "%b")
print(MGD_plot)
```

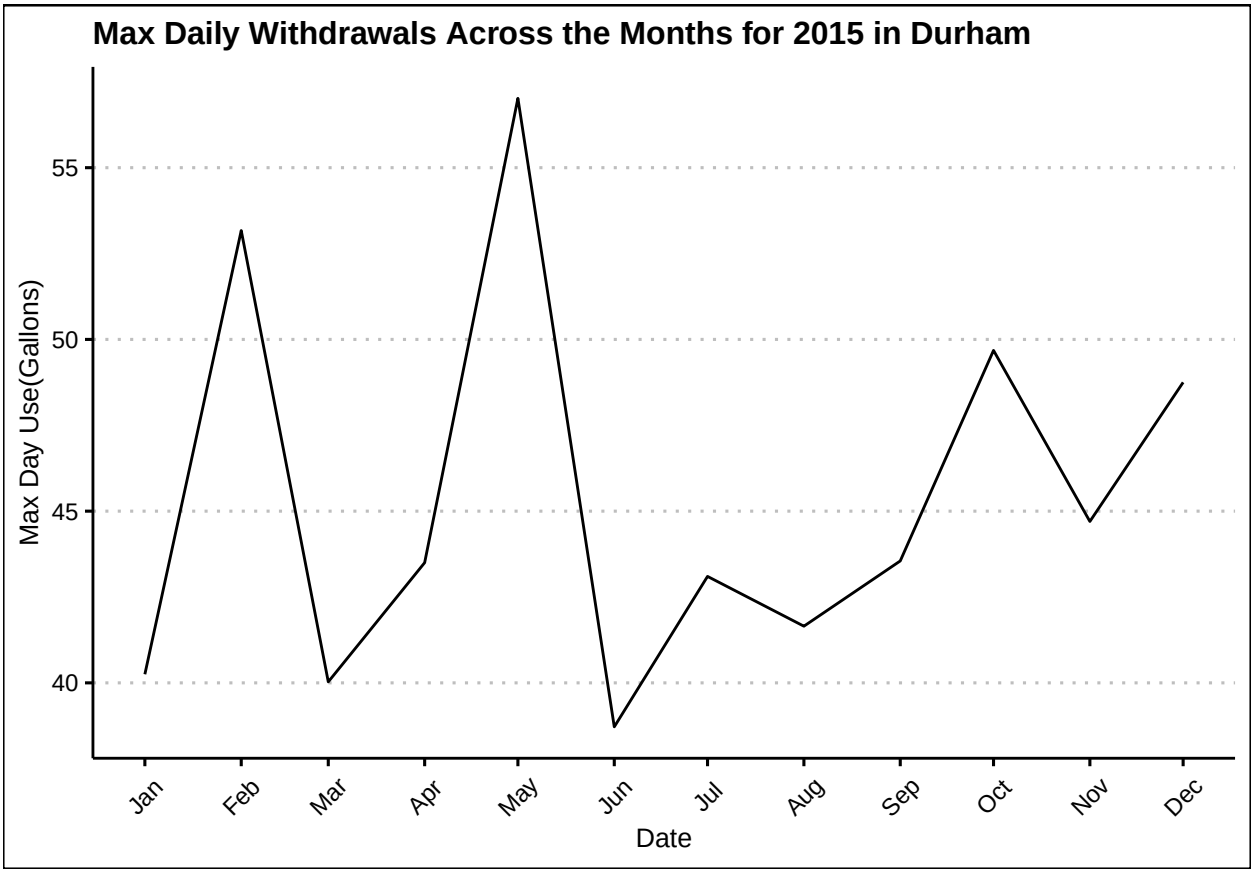


6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
# 6.
scrape.it = function(pwsid, the_year) {
    the_url = ifelse(pwsid == "03-32-010" &
        the_year == 2022, "https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022",
        paste0("https://www.ncwater.org/WUDC/app/LWSP/report.php",
            "?", "pwsid=", pwsid, "&", "year=",
            the_year))
    website = read_html(the_url)
    water.system.name2 = website %>%
        html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
        html_text()
    PWSID2 = website %>%
        html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
        html_text()
    ownership2 = website %>%
        html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
        html_text()
    max.withdrawals.mgd2 = website %>%
        html_nodes("th~ td+ td") %>%
        html_text()
    withdrawal_df2 = data.frame(Month = rep(1:12),
        `Water System Name` = water.system.name2,
        Ownership = ownership2, PWSID = PWSID2,
        MGD = as.numeric(max.withdrawals.mgd2))
    withdrawal_df2 = withdrawal_df2 %>%
        mutate(Date = ym(paste0(the_year,
            "-", Month)))
    return(withdrawal_df2)
}
```

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

```
# 7
Durham_2015 = scrape.it("03-32-010", 2015)
Durham_2015_plot = ggplot(Durham_2015) +
    geom_line(aes(x = Date, y = MGD)) + ylab("Max Day Use(Gallons)") +
    xlab("Date") + labs(title = "Max Daily Withdrawals Across the Months for 2015 in Durham") +
    scale_x_date(date_breaks = "1 month",
        date_labels = "%b")
print(Durham_2015_plot)
```



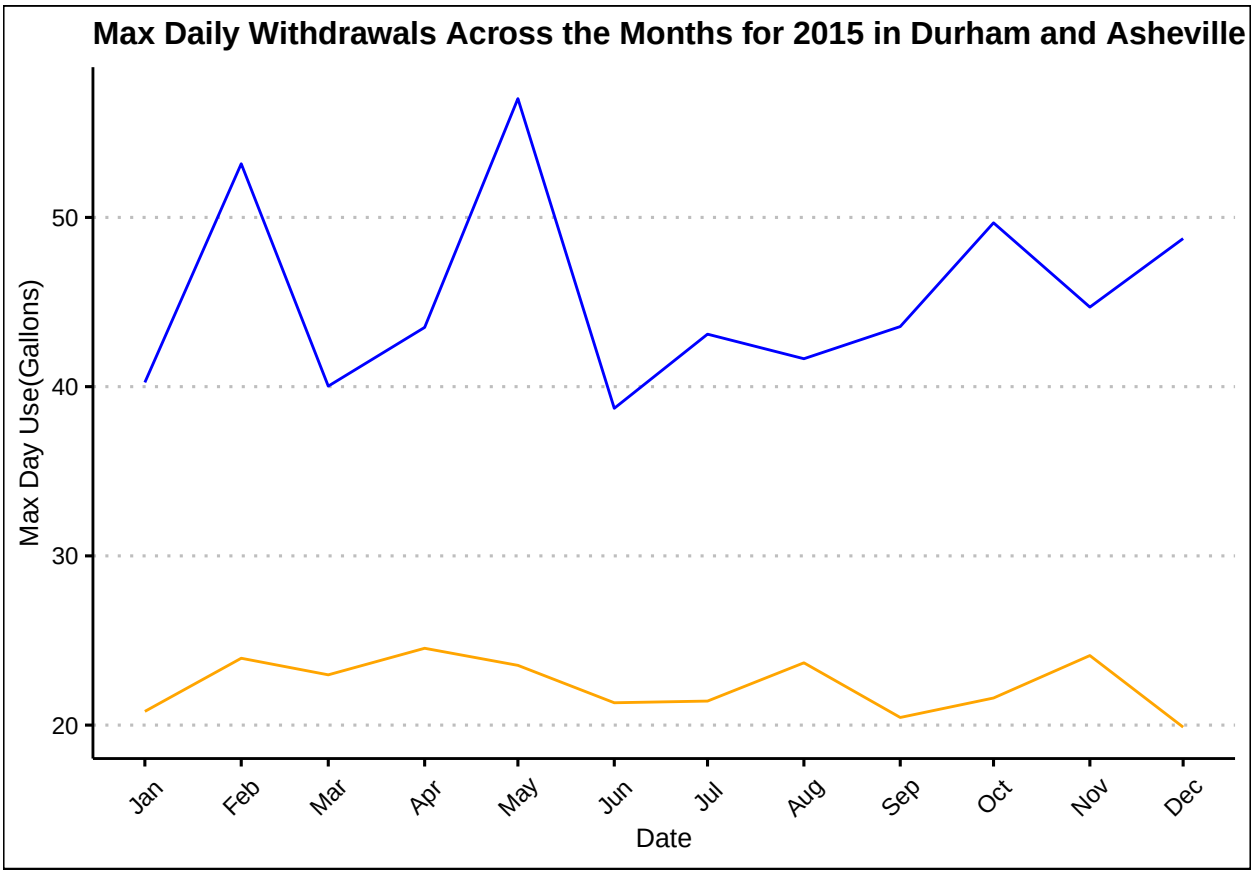
8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
# 8
Asheville_2015 = scrape.it("01-11-010", 2015)
Asheville_2015
```

```
##    Month Water.System.Name    Ownership     PWSID   MGD       Date
## 1      1         Asheville Municipality 01-11-010 20.81 2015-01-01
## 2      2         Asheville Municipality 01-11-010 23.95 2015-02-01
## 3      3         Asheville Municipality 01-11-010 22.97 2015-03-01
## 4      4         Asheville Municipality 01-11-010 24.54 2015-04-01
## 5      5         Asheville Municipality 01-11-010 23.53 2015-05-01
## 6      6         Asheville Municipality 01-11-010 21.32 2015-06-01
## 7      7         Asheville Municipality 01-11-010 21.42 2015-07-01
## 8      8         Asheville Municipality 01-11-010 23.68 2015-08-01
## 9      9         Asheville Municipality 01-11-010 20.45 2015-09-01
## 10    10         Asheville Municipality 01-11-010 21.60 2015-10-01
## 11    11         Asheville Municipality 01-11-010 24.11 2015-11-01
## 12    12         Asheville Municipality 01-11-010 19.88 2015-12-01
```

```
Compare_plot = ggplot(Durham_2015, aes(x = Date,
    y = MGD)) + geom_line(color = "blue") +
    ylab("Max Day Use(Gallons)") + xlab("Date") +
```

```
    geom_line(data = Asheville_2015, color = "orange") +
    labs(title = "Max Daily Withdrawals Across the Months for 2015 in Durham and Asheville") +
    scale_x_date(date_breaks = "1 month",
        date_labels = "%b")
print(Compare_plot)
```

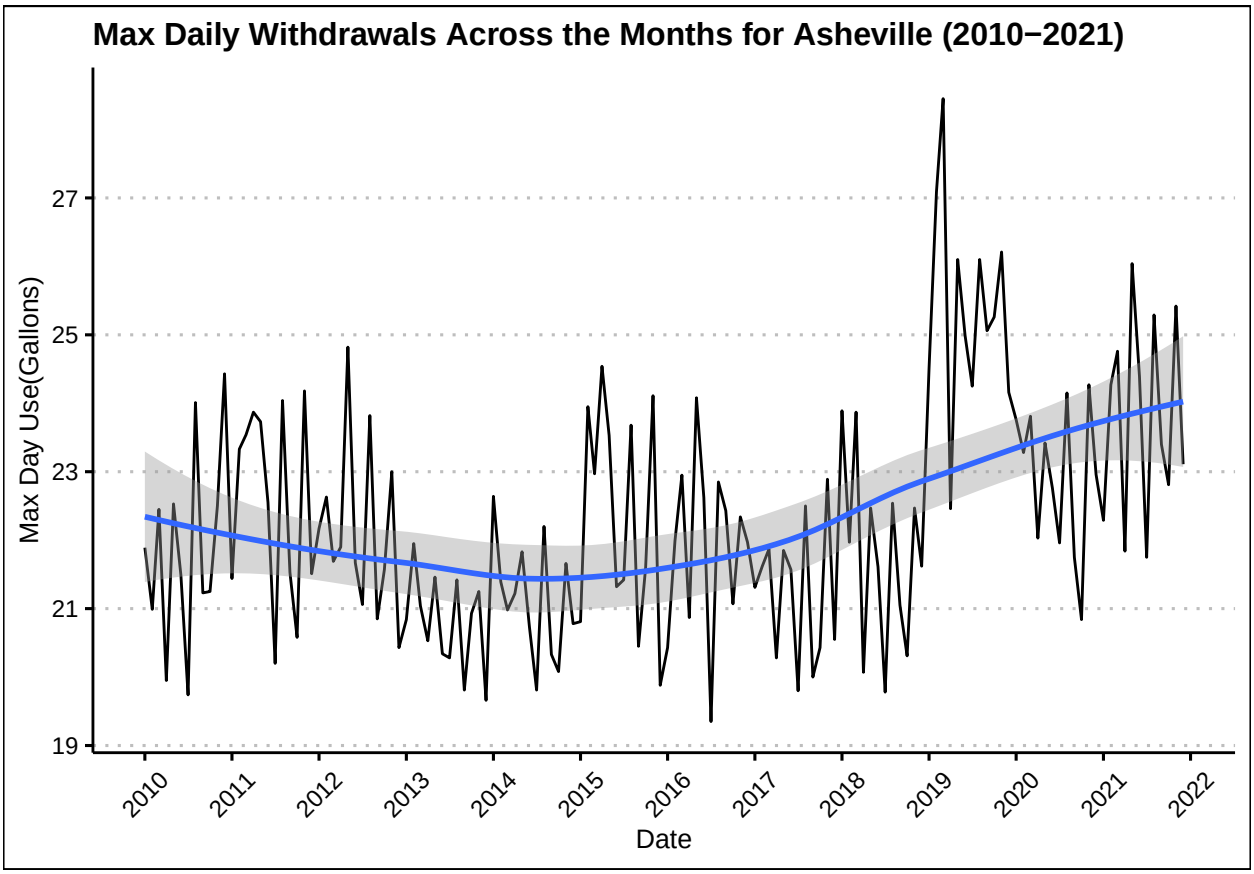


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   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
Asheville_years = map2("01-11-010", c(2010:2021),
    scrape.it)
Asheville_years_df = bind_rows(Asheville_years)
Asheville_years_plot = ggplot(Asheville_years_df,
    aes(x = Date, y = MGD)) + geom_line() +
    geom_smooth(method = "loess") + ylab("Max Day Use(Gallons)") +
    xlab("Date") + labs(title = "Max Daily Withdrawals Across the Months for Asheville (2010-2021)") +
    scale_x_date(date_breaks = "1 year",
        date_labels = "%Y")
print(Asheville_years_plot)
```

```
## `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: There's no trend in water usage over time. From 2010 to 2015, the water usage went down and it increased again from 2016 to 2021.