Assignment 09: Data Scraping

Megan McClaugherty

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A09_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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up

1. Set up your session:

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

```
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=40), tidy=TRUE)
#1 Checking working directory, loading packages, setting theme.

getwd()
```

```
## [1] "/home/guest/EDA-Fall2022/Assignments"
```

```
library(tidyverse)
library(rvest)
library(dplyr)
library(lubridate)



A9Theme <- theme_light(base_size = 12) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "bottom")
theme_set(A9Theme)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2021 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=2021

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

```
# 2 reading in the LWSP website for
# Durham in 2021

DurhamLWSPwebpage <- read_html("https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2
DurhamLWSPwebpage
```

```
## {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
- PSWID
- Ownership
- From the "3. Water Supply Sources" section:
- Maximum Daily 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".

```
# 3 Scraping the data for the 4
# variables of interest

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

```
## [1] "Durham"
```

```
pswid <- DurhamLWSPwebpage %>%
    html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
    html_text()
pswid
```

```
## [1] "03-32-010"
```

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

```
## [1] "Municipality"
```

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

```
##  [1] "27.6400" "41.7900" "36.7200" "27.9700" "37.9500" "42.2400" "30.5400"
##  [8] "43.6200" "31.2800" "33.7600" "46.0800" "29.7800"
```

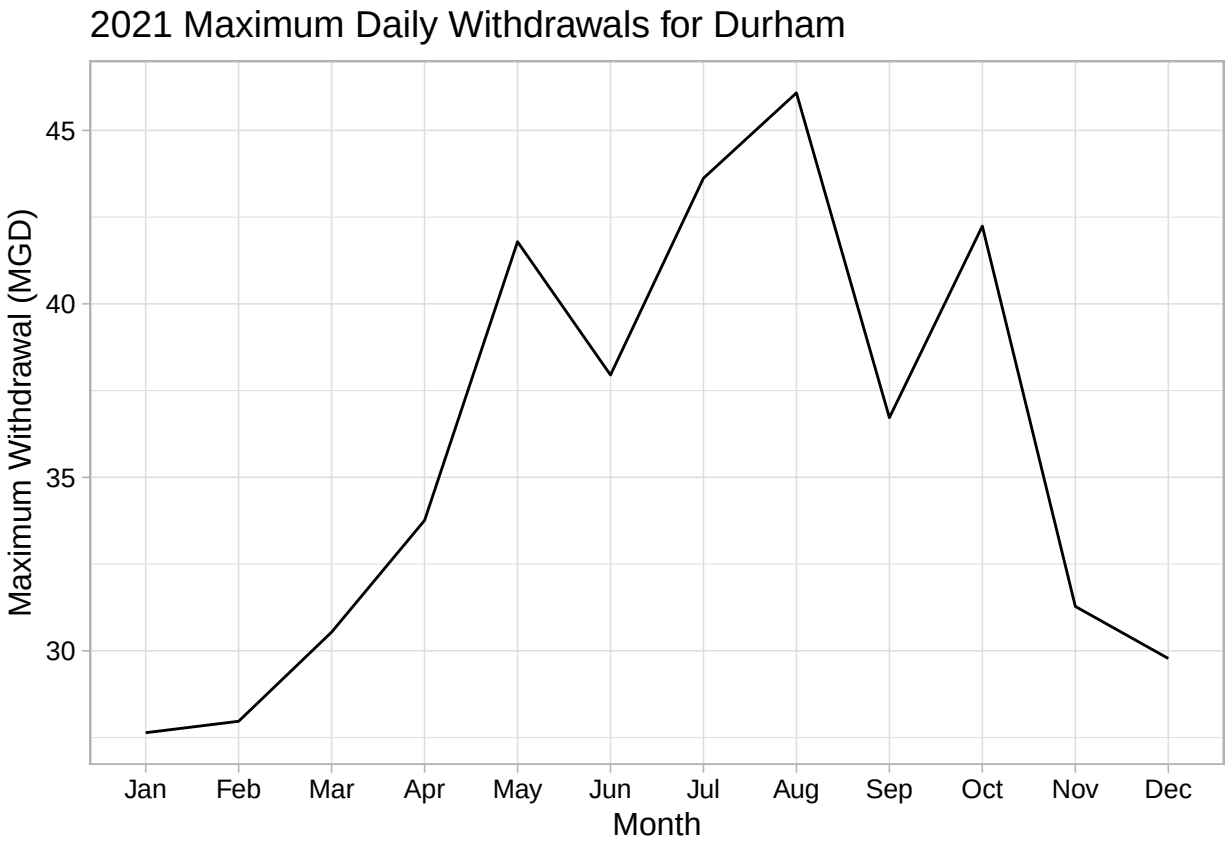
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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...

5. Create a line plot of the maximum daily withdrawals across the months for 2021

```
# 4 Creating a dataframe with the 4
# variables
DurhamLWSPdf <- data.frame(Month = c("Jan",
    "May", "Sep", "Feb", "Jun", "Oct", "Mar",
    "Jul", "Nov", "Apr", "Aug", "Dec"), Water_System_Name = (water.system.name),
    Ownership = (ownership), PSWID = (pswid),
    Max_Withdrawals_MGD = as.numeric(max.withdrawals.mgd)) %>%
    mutate(Date = my(paste0(Month, "-", 2021)))

# arranging in chronological order
DurhamLWSPdf <- arrange(DurhamLWSPdf, Date)


# 5 Line plot of max daily withdrawals
# for each month in 2021.
MaxWithdrawalbyMonth <- ggplot(DurhamLWSPdf,
    aes(x = Month, y = Max_Withdrawals_MGD,
        group = 1)) + geom_line() + scale_x_discrete(limits = month.abb) +
    labs(title = "2021 Maximum Daily Withdrawals for Durham",
        y = "Maximum Withdrawal (MGD)")
print(MaxWithdrawalbyMonth)
```



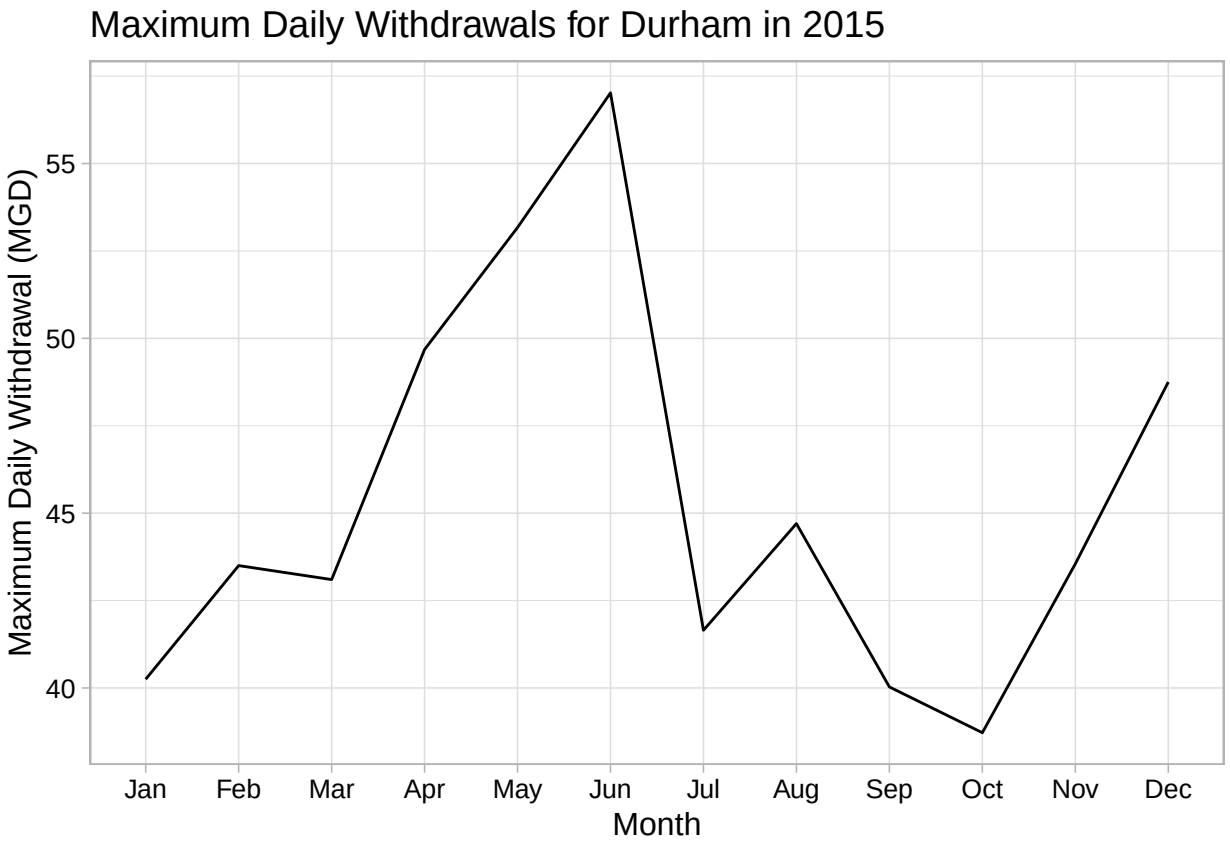
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
# 6. Creating a scrape function to
# scrape data for any PWSID and year

baseURL <- "https://www.ncwater.org/WUDC/app/LWSP/report.php?"


scrape.it <- function(PWSID, Year) {
    website <- read_html(paste0("https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=",
        PWSID, "&year=", Year))
    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)"
    max.withdrawals.mgd_tag <- "th~ td+ td"

    # reading in the data for each of
    # the 4 variables
    watersystem <- website %>%
        html_nodes(watersystem_tag) %>%
        html_text()
    pwsid <- website %>%
        html_nodes(pwsid_tag) %>%
        html_text()
    ownership <- website %>%
```

```
        html_nodes(ownership_tag) %>%
        html_text()
    maxwithdrmgd <- website %>%
        html_nodes(max.withdrawals.mgd_tag) %>%
        html_text()

    withdrawalsdf <- data.frame(Month = c("Jan",
        "May", "Sep", "Feb", "Jun", "Oct",
        "Mar", "Jul", "Nov", "Apr", "Aug",
        "Dec"), Year = rep(Year, 12), Max.Daily.WithdrawalMGD = as.numeric(maxwithdrmgd)) %>%
        mutate(Watersystem = !!watersystem,
            PWSID = !!pwsid, Ownership = !!ownership,
            Date = my(paste(Month, "-", Year)))
}
```

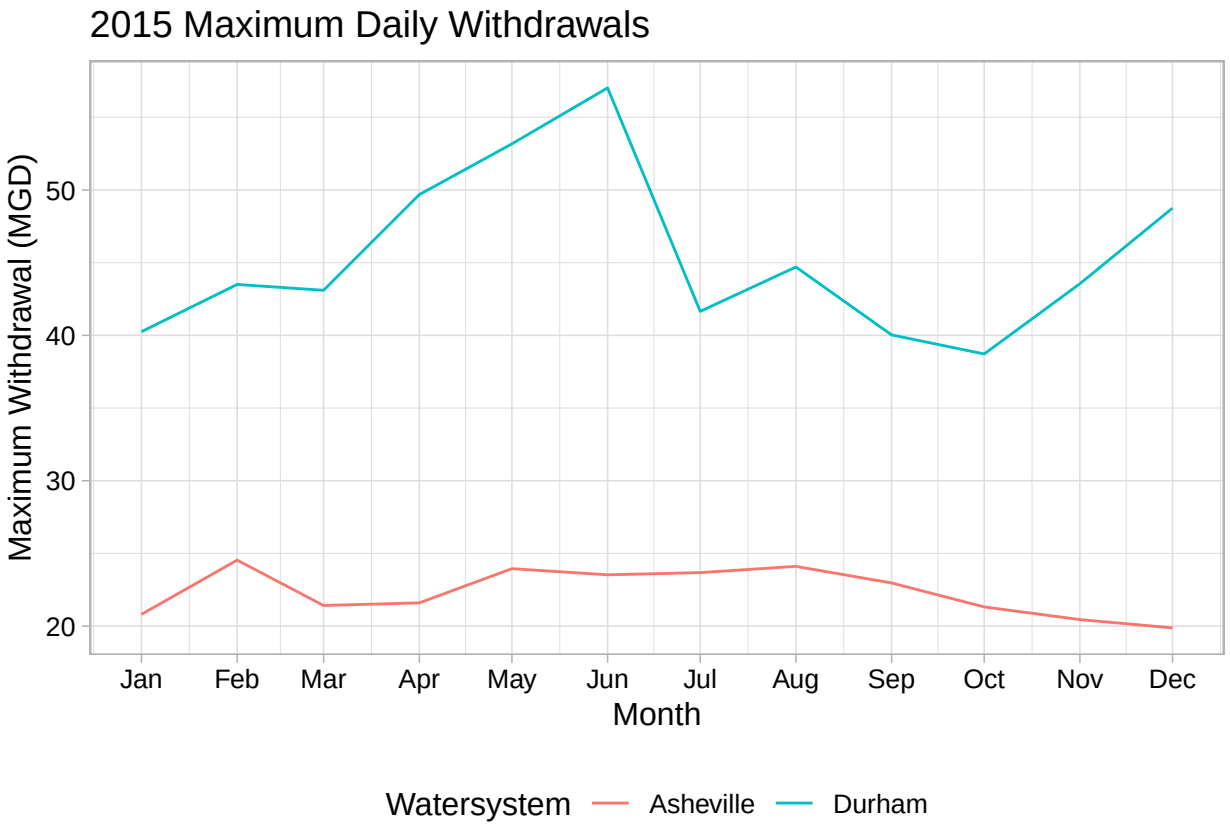
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID='03-32-010') for each month in 2015

```
# 7 Setting the year to 2015 and
# creating the unique scrape url
Durham2015df <- scrape.it("03-32-010", 2015)

Durham2015df <- arrange(Durham2015df, Date)

# plotting 2015 daily max withdrawal
# data
Durham2015plot <- ggplot(Durham2015df, aes(x = Month,
    y = Max.Daily.WithdrawalMGD, group = 1)) +
    geom_line() + scale_x_discrete(limits = month.abb) +
    labs(title = "Maximum Daily Withdrawals for Durham in 2015",
        y = "Maximum Daily Withdrawal (MGD)")
print(Durham2015plot)
```



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 Using the function to extract data
# for Asheville in 2015.
Asheville2015df <- scrape.it("01-11-010",
    2015)

Asheville2015df <- arrange(Asheville2015df,
    Date)


# Combining the two dataframes and
# creating a plot that compares the
# withdrawal of Asheville and Durham
AshevilleandDurham2015 <- bind_rows(Asheville2015df,
    Durham2015df)
AshevilleandDurhamplot <- ggplot(AshevilleandDurham2015,
    aes(x = Date, y = Max.Daily.WithdrawalMGD,
        color = Watersystem)) + geom_line() +
    scale_x_date(date_breaks = "1 month",
        date_labels = "%b") + labs(title = "2015 Maximum Daily Withdrawals",
    y = "Maximum Withdrawal (MGD)", x = "Month")

print(AshevilleandDurhamplot)
```

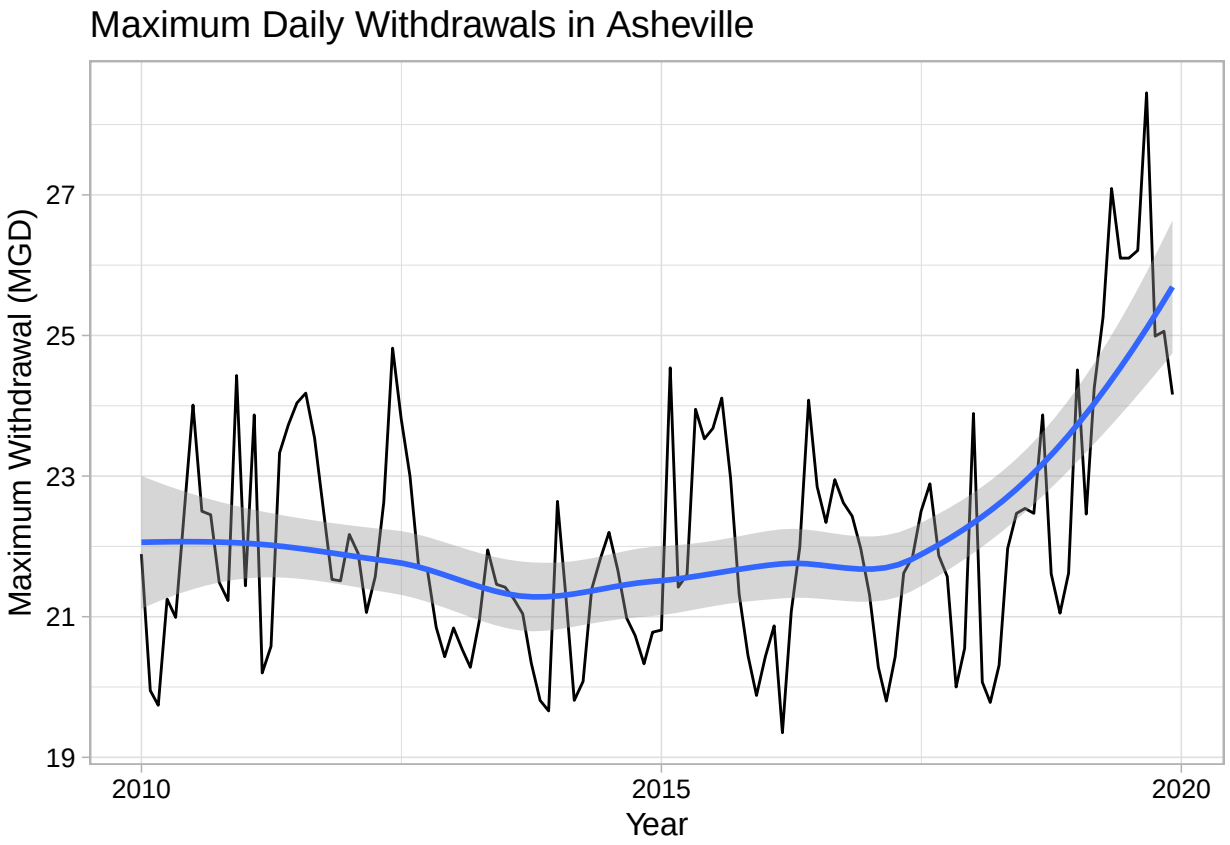


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

   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 Using the scape function I created
# to pull 2010-2019 data and plotting.

theyears <- rep(2010:2019)
Asheville_PWSID <- "01-11-010"

Asheville_NineYear <- map(theyears, scrape.it,
    PWSID = Asheville_PWSID)

Asheville_NineYear_df <- bind_rows(Asheville_NineYear)

Asheville_NineYear_plot <- ggplot(Asheville_NineYear_df,
    aes(x = Date, y = Max.Daily.WithdrawalMGD)) +
    geom_line() + geom_smooth() + labs(title = "Maximum Daily Withdrawals in Asheville",
    y = "Maximum Withdrawal (MGD)", x = "Year")

print(Asheville_NineYear_plot)
```

```
## `geom_smooth()` using method = 'loess' and 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 2009 to about 2017, it appears that Asheville's water usage remained relatively stable, but after around 2017 it appears to be trending upward.