Assignment 10: Data Scraping

Hanna Karnei

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 Load libraries

library(tidyverse)
library(dplyr)
library(here)
library(rvest)
getwd()
```

```
## [1] "/Users/hannakarnei/Desktop/EDA/EDE_Fall2023"
```

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

```
## {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)".

```
#3Scrape data

water_system_name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()

water_system_name
```

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

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


PWSID
```

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

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

ownership
```

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

```
max_day_use <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()

max_day_use
```

```
##  [1] "36.1000" "43.4200" "52.4900" "30.5000" "42.5900" "34.8800" "39.9100"
##  [8] "43.3200" "32.5300" "34.6600" "41.8000" "37.5300"
```

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.

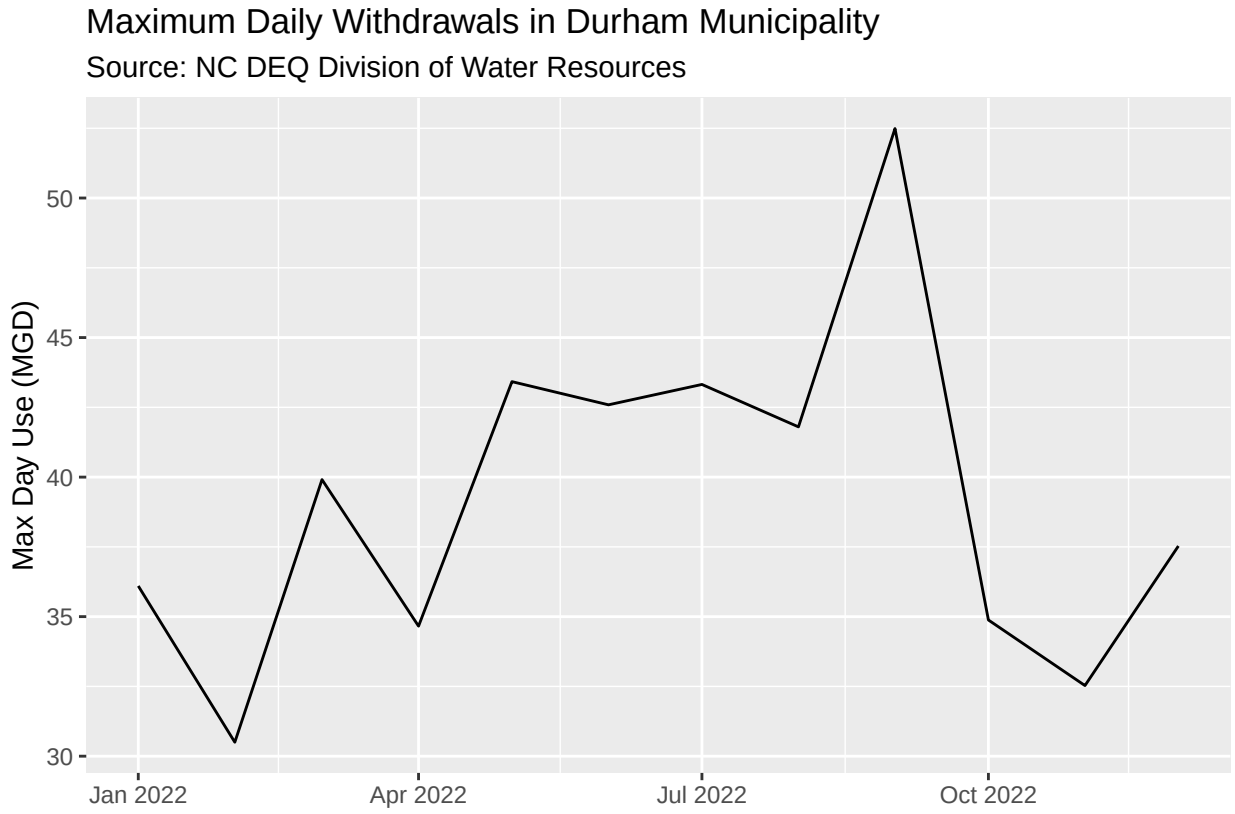
   NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc... Or, you could scrape month values from the web page...

5. Create a line plot of the maximum daily withdrawals across the months for 2022

```
#4 Create a df


df <- data.frame("Water_System_Name" = water_system_name,
                 "PWSID" = PWSID,
                 "Ownership" = ownership,
                 "Date" = as.Date(paste(2022, c("Jan", "May", "Sep", "Feb", "Jun", "Oct", "Mar", "Jul",
                 "Max_Day_Use_MGD" = as.numeric(max_day_use))

head(df)
```

```
##   Water_System_Name     PWSID    Ownership       Date Max_Day_Use_MGD
## 1            Durham 03-32-010 Municipality 2022-01-01           36.10
## 2            Durham 03-32-010 Municipality 2022-05-01           43.42
## 3            Durham 03-32-010 Municipality 2022-09-01           52.49
## 4            Durham 03-32-010 Municipality 2022-02-01           30.50
## 5            Durham 03-32-010 Municipality 2022-06-01           42.59
## 6            Durham 03-32-010 Municipality 2022-10-01           34.88
```

```
#5 Create a line plot

graph <- ggplot(df,aes(x=Date,y=Max_Day_Use_MGD)) +
  geom_line() +
  labs(title = ('Maximum Daily Withdrawals in Durham Municipality'),
       subtitle = "Source: NC DEQ Division of Water Resources",
       y="Max Day Use (MGD)",
       x=" ")

graph
```



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. Constrcut a scraping function

scrape.it <- function(the_pwsid, the_year){

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

  #Set the element address variables
  the_pwsid_tag <- ".sortable td:nth-child(1)"
  the_data_tag <- "th~ td+ td"

  #Scrape the data items
  the_pwsid <- the_website %>% html_nodes(the_pwsid_tag) %>% html_text()
  the_max_day_use <- the_website %>% html_nodes(the_data_tag) %>% html_text()

  #Convert to a dataframe
  df_max_day_use<- data.frame("Date" = as.Date(paste(the_year, c("Jan", "May", "Sep", "Feb", "Jun", "Oc
                              "Max_Day_Use_MGD" = as.numeric(the_max_day_use))

  Sys.sleep(1)
```

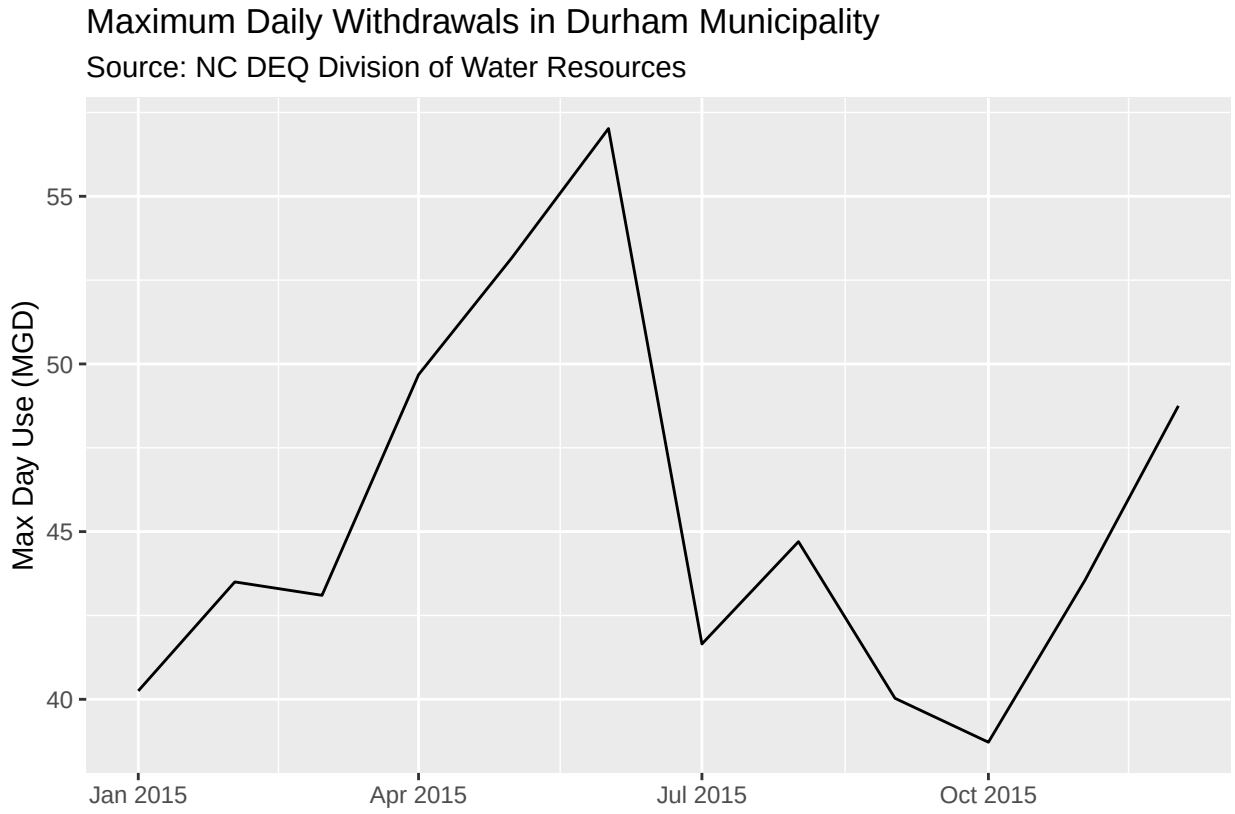
```
  #Return the dataframe
  return(df_max_day_use)
}
```

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 <- scrape.it('03-32-010', 2015)
head(durham)
```

```
##         Date Max_Day_Use_MGD
## 1 2015-01-01           40.25
## 2 2015-05-01           53.17
## 3 2015-09-01           40.03
## 4 2015-02-01           43.50
## 5 2015-06-01           57.02
## 6 2015-10-01           38.72
```

```
graph2 <- ggplot(durham,aes(x=Date,y=Max_Day_Use_MGD)) +
  geom_line() +
  labs(title = ('Maximum Daily Withdrawals in Durham Municipality'),
       subtitle = "Source: NC DEQ Division of Water Resources",
       y="Max Day Use (MGD)",
       x=" ")

graph2
```



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 <- scrape.it('01-11-010', 2015)
head(asheville)
```

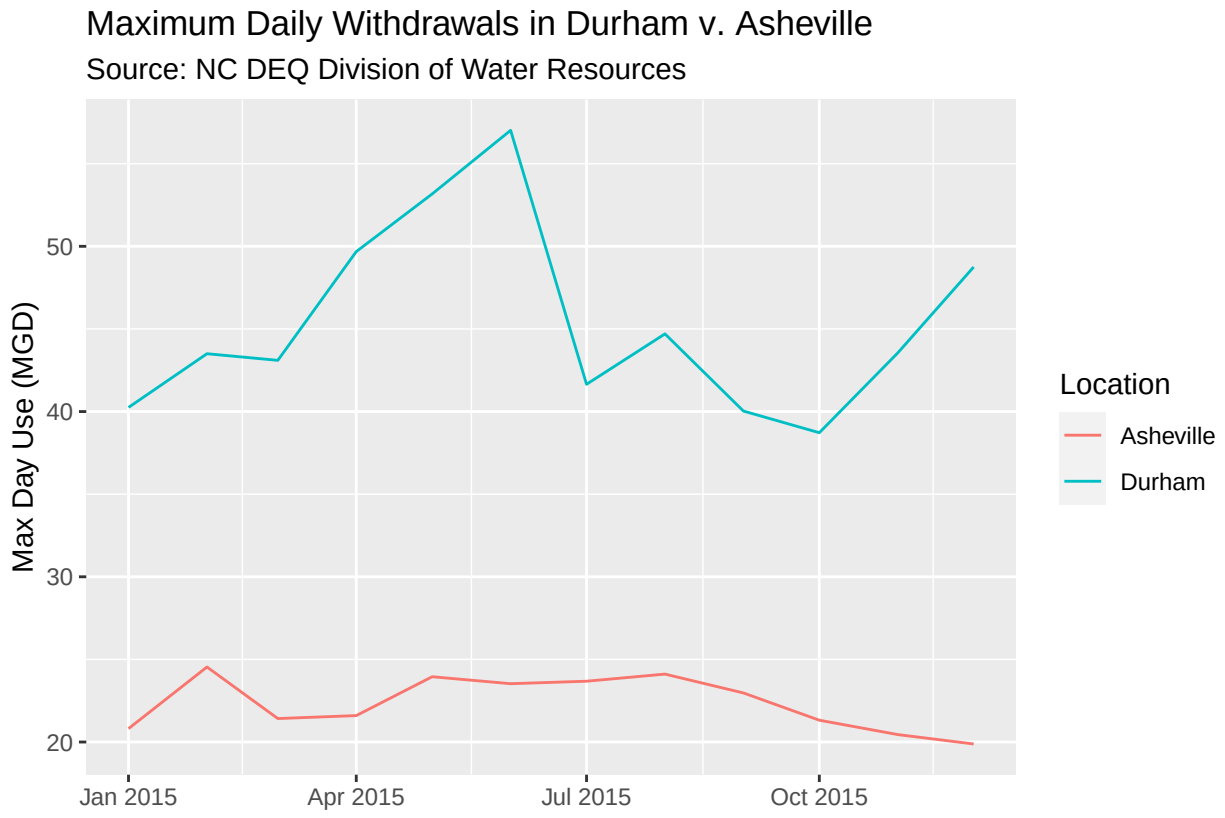
```
##         Date Max_Day_Use_MGD
## 1 2015-01-01           20.81
## 2 2015-05-01           23.95
## 3 2015-09-01           22.97
## 4 2015-02-01           24.54
## 5 2015-06-01           23.53
## 6 2015-10-01           21.32
```

```
combined_df <- bind_rows(
  mutate(durham, Location = "Durham"),
  mutate(asheville, Location = "Asheville")
)

head(combined_df)
```

```
##         Date Max_Day_Use_MGD Location
```

```
## 1 2015-01-01           40.25   Durham
## 2 2015-05-01           53.17   Durham
## 3 2015-09-01           40.03   Durham
## 4 2015-02-01           43.50   Durham
## 5 2015-06-01           57.02   Durham
## 6 2015-10-01           38.72   Durham
```

```
graph3 <- ggplot(combined_df,aes(x=Date,y=Max_Day_Use_MGD, color=Location)) +
  geom_line() +
  labs(title = ('Maximum Daily Withdrawals in Durham v. Asheville'),
       subtitle = "Source: NC DEQ Division of Water Resources",
       y="Max Day Use (MGD)",
       x=" ")

graph3
```



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').

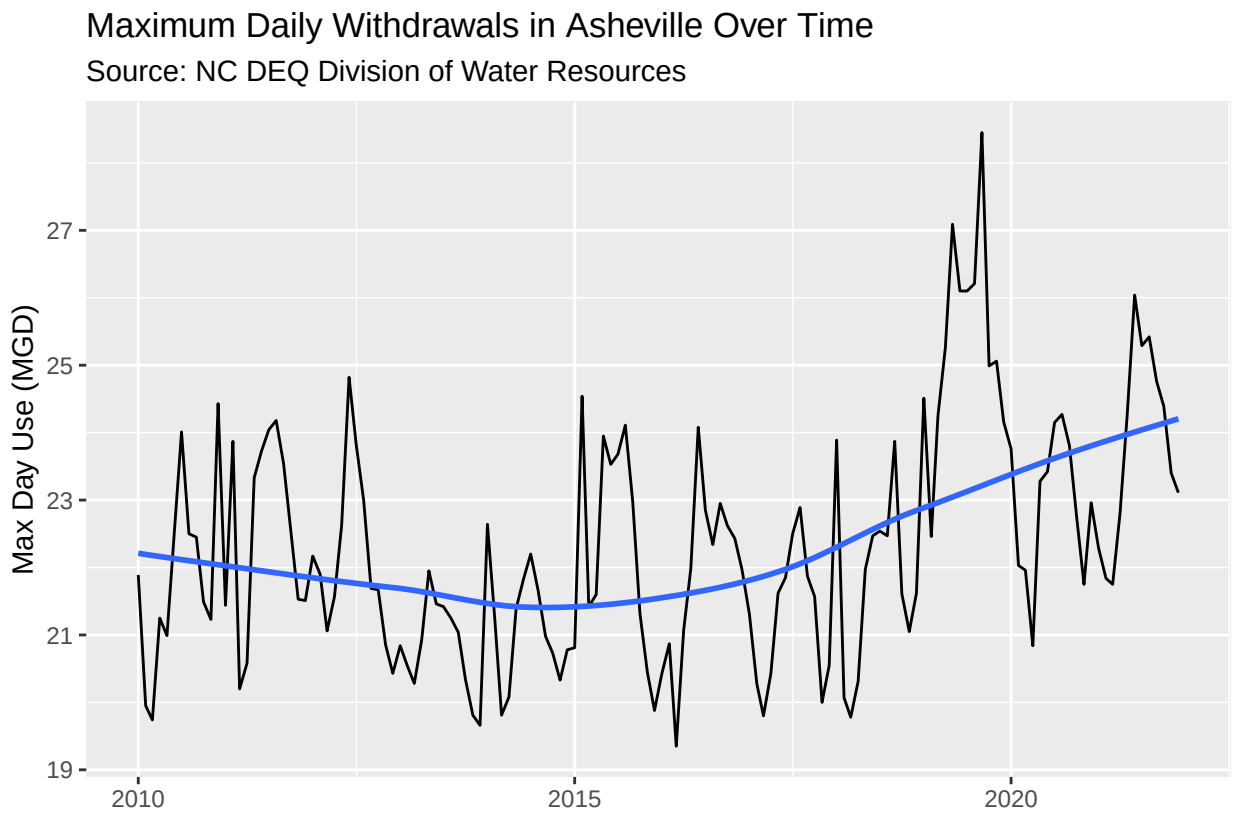
   TIP: See Section 3.2 in the "10_Data_Scraping.Rmd" where we apply "map2()" to iteratively run a function over two inputs. Pipe the output of the map2() function to `bindrows()` to combine the dataframes into a single one.

```r
#9

the_years = rep(2010:2021)
my_pwsid = '01-11-010'

the_dfs <- map(the_years,scrape.it,the_pwsid=my_pwsid)

#Conflate the returned dataframes into a single dataframe
asheville_2010_2022 <- bind_rows(the_dfs)
head(asheville_2010_2022)
```

```
##         Date Max_Day_Use_MGD
## 1 2010-01-01           21.89
## 2 2010-05-01           20.99
## 3 2010-09-01           22.45
## 4 2010-02-01           19.95
## 5 2010-06-01           22.53
## 6 2010-10-01           21.49
```

```r
graph4 <- ggplot(asheville_2010_2022,aes(x=Date,y=Max_Day_Use_MGD)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = ('Maximum Daily Withdrawals in Asheville Over Time'),
       subtitle = "Source: NC DEQ Division of Water Resources",
       y="Max Day Use (MGD)",
       x=" ")

graph4
```

```
## `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 is a clear trend of an increasing maximum daily water usage from 2015 to 2021. From 2010 to 2015, water usage declied slightly, but there is no clearly visible trend.