Assignment 10: Data Scraping

Tani Valdez Rivas

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

# Installing packages
library(tidyverse)
library(lubridate)
library(viridis)
library(here)
library(rvest)

# Checking working directory
here()
```

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

# Using the read_html() to read the contents
NCDEQ_Web <-
  read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
NCDEQ_Web
```

```
## {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)“.

```
#3

# Scraping the water system name
the_water_system <- NCDEQ_Web %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
the_water_system
```

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

```
# Scraping the PWSID
the_PWSID <- NCDEQ_Web %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
the_PWSID
```

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

```r
# Scraping the ownership
the_ownership <- NCDEQ_Web %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
the_ownership
```

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

```r
# Selecting the MGD of the water supply source for each month

max_monthly_MGD <- NCDEQ_Web %>%
  html_nodes('th~ td+ td') %>%
  html_text()
max_monthly_MGD
```

```
##  [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.

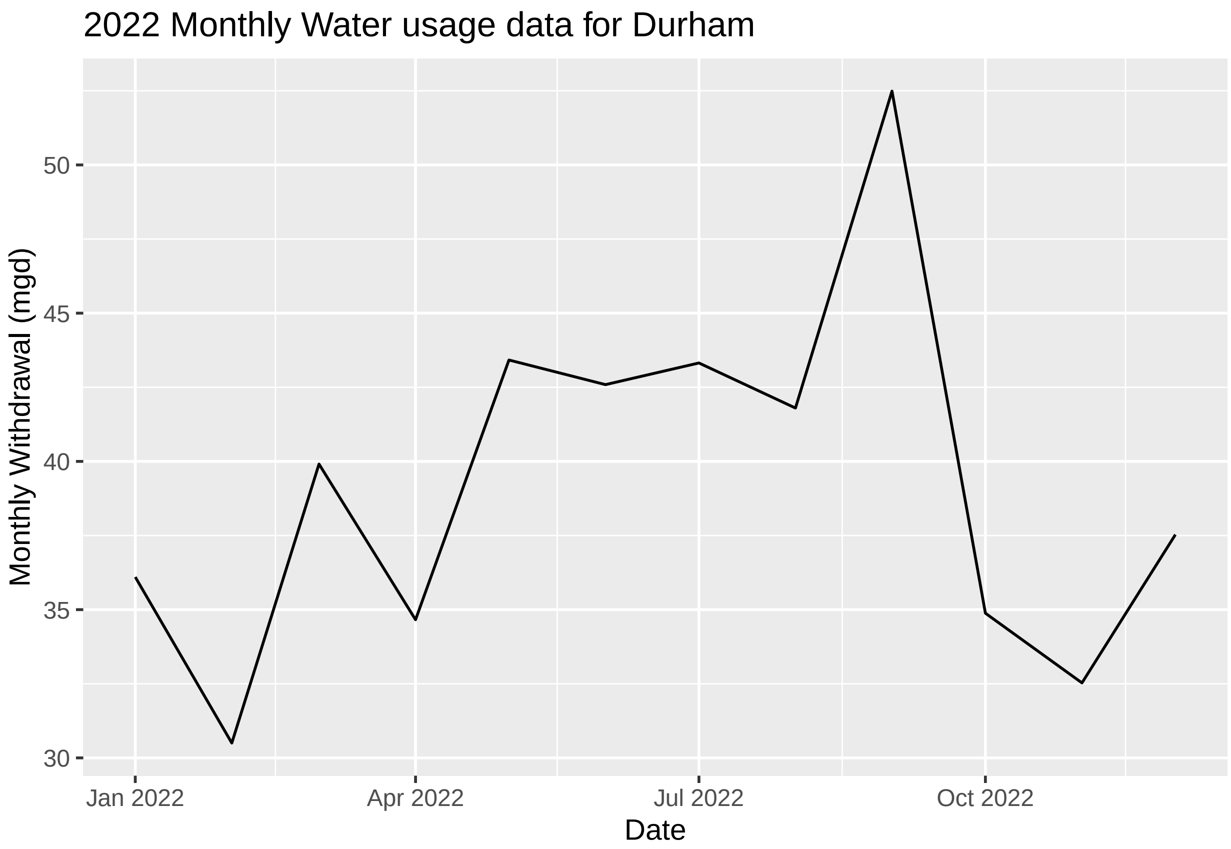
   NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc… Or, you could scrape month values from the web page…

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

```r
#4

# Creating Month and year
Months <- c("Jan", "May", "Sep", "Feb", "Jun", "Oct",
            "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")
Year <- 2022

# Using the data.frame function to scrap data into data frame
df_NCDEQ <- data.frame(Month = rep(Months, each = 1),
                       Year = rep(Year, 12),
                       "Monthly_Max_mgd" = as.numeric(max_monthly_MGD))%>%
  mutate(Water_System = !!the_water_system,
         PWSID = !!the_PWSID,
         Ownership = !!the_ownership,
         Date = my(paste(Month,"-",Year)))
df_NCDEQ
```

```
##    Month Year Monthly_Max_mgd Water_System     PWSID    Ownership       Date
## 1    Jan 2022           36.10       Durham 03-32-010 Municipality 2022-01-01
## 2    May 2022           43.42       Durham 03-32-010 Municipality 2022-05-01
## 3    Sep 2022           52.49       Durham 03-32-010 Municipality 2022-09-01
## 4    Feb 2022           30.50       Durham 03-32-010 Municipality 2022-02-01
## 5    Jun 2022           42.59       Durham 03-32-010 Municipality 2022-06-01
## 6    Oct 2022           34.88       Durham 03-32-010 Municipality 2022-10-01
## 7    Mar 2022           39.91       Durham 03-32-010 Municipality 2022-03-01
## 8    Jul 2022           43.32       Durham 03-32-010 Municipality 2022-07-01
## 9    Nov 2022           32.53       Durham 03-32-010 Municipality 2022-11-01
## 10   Apr 2022           34.66       Durham 03-32-010 Municipality 2022-04-01
## 11   Aug 2022           41.80       Durham 03-32-010 Municipality 2022-08-01
## 12   Dec 2022           37.53       Durham 03-32-010 Municipality 2022-12-01
```

```
#5

# Using ggplot to create line plot of dataframe
Max_Withdrawls_2022 <-
  ggplot(df_NCDEQ,aes(x=Date, y=Monthly_Max_mgd)) +
  geom_line() +
  labs(title = paste("2022 Monthly Water usage data for",the_water_system),
       y="Monthly Withdrawal (mgd)",
       x="Date")
Max_Withdrawls_2022
```



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.

# Using function to scrape any PWSID and year
scrape_NCDEQ <- function(the_PWSID, Year){
  NCDEQ_Web <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',
                                  the_PWSID, '&year=', Year))

# Using the code from question 3

# Scraping the water system name
the_water_system <- NCDEQ_Web %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
the_water_system

# Scraping the PWSID
the_PWSID <- NCDEQ_Web %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
the_PWSID

# Scraping the ownership
the_ownership <- NCDEQ_Web %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
the_ownership

# Selecting the MGD of the water supply source for each month
max_monthly_MGD <- NCDEQ_Web %>%
  html_nodes('th~ td+ td') %>%
  html_text()
max_monthly_MGD

#Convert to a dataframe using data
df_NCDEQ <- data.frame(Month = rep(Months, each = 1),
                       Year = rep(Year, 12),
                       "Monthly_Max_mgd" = as.numeric(max_monthly_MGD))%>%
  mutate(Water_System = !!the_water_system,
         PWSID = !!the_PWSID,
         Ownership = !!the_ownership,
         Date = my(paste(Month,"-",Year)))


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

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

```
#7

# Running the fetch and plot for Durham
```

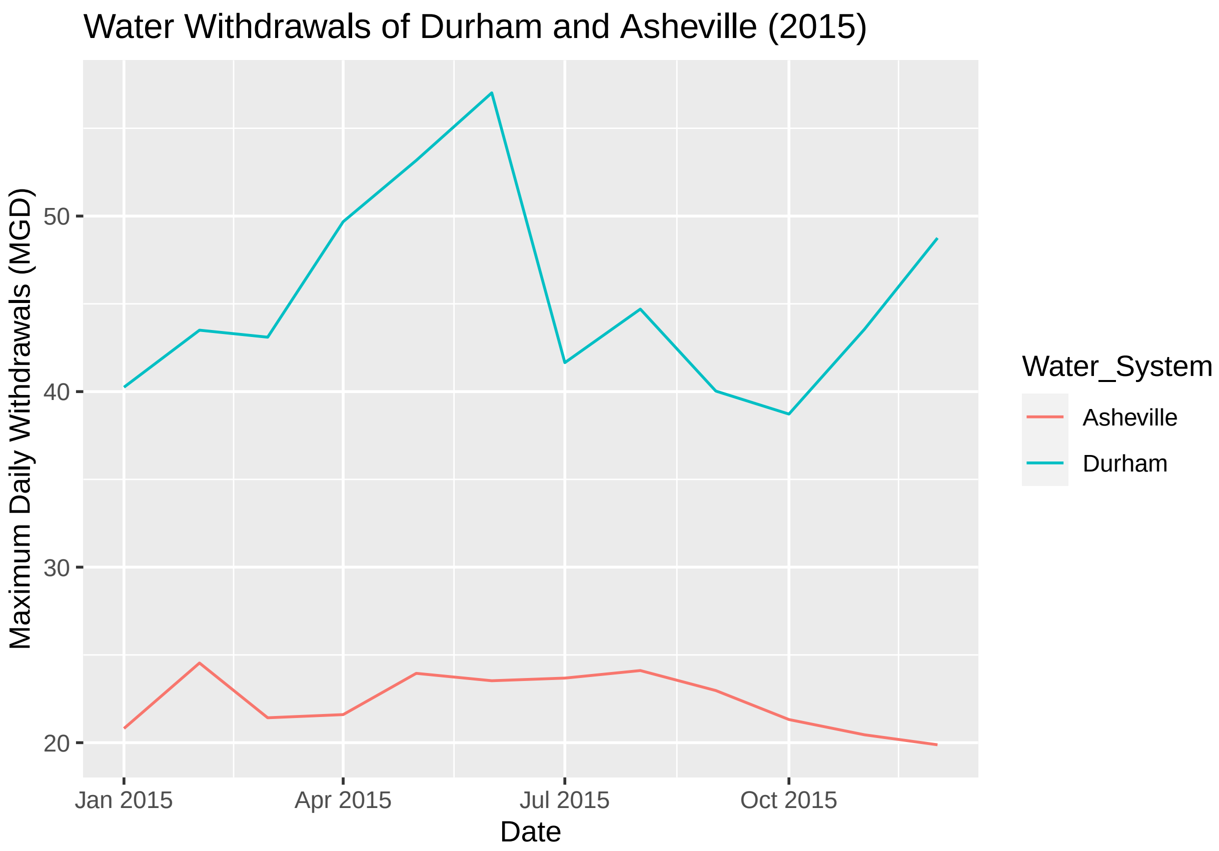
```
Durham2015 <- scrape_NCDEQ('03-32-010',2015)
view(Durham2015)
```

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

# Running the fetch and plot for Ashville
Asheville2015 <- scrape_NCDEQ('01-11-010',2015)
view(Asheville2015)

# Combining Asheville and Durham 2015 data
AshevilleDurham2015 <- bind_rows(Durham2015, Asheville2015)
AshevilleDurham2015
```

```
##    Month Year Monthly_Max_mgd Water_System     PWSID    Ownership       Date
## 1    Jan 2015           40.25       Durham 03-32-010 Municipality 2015-01-01
## 2    May 2015           53.17       Durham 03-32-010 Municipality 2015-05-01
## 3    Sep 2015           40.03       Durham 03-32-010 Municipality 2015-09-01
## 4    Feb 2015           43.50       Durham 03-32-010 Municipality 2015-02-01
## 5    Jun 2015           57.02       Durham 03-32-010 Municipality 2015-06-01
## 6    Oct 2015           38.72       Durham 03-32-010 Municipality 2015-10-01
## 7    Mar 2015           43.10       Durham 03-32-010 Municipality 2015-03-01
## 8    Jul 2015           41.65       Durham 03-32-010 Municipality 2015-07-01
## 9    Nov 2015           43.55       Durham 03-32-010 Municipality 2015-11-01
## 10   Apr 2015           49.68       Durham 03-32-010 Municipality 2015-04-01
## 11   Aug 2015           44.70       Durham 03-32-010 Municipality 2015-08-01
## 12   Dec 2015           48.75       Durham 03-32-010 Municipality 2015-12-01
## 13   Jan 2015           20.81    Asheville 01-11-010 Municipality 2015-01-01
## 14   May 2015           23.95    Asheville 01-11-010 Municipality 2015-05-01
## 15   Sep 2015           22.97    Asheville 01-11-010 Municipality 2015-09-01
## 16   Feb 2015           24.54    Asheville 01-11-010 Municipality 2015-02-01
## 17   Jun 2015           23.53    Asheville 01-11-010 Municipality 2015-06-01
## 18   Oct 2015           21.32    Asheville 01-11-010 Municipality 2015-10-01
## 19   Mar 2015           21.42    Asheville 01-11-010 Municipality 2015-03-01
## 20   Jul 2015           23.68    Asheville 01-11-010 Municipality 2015-07-01
## 21   Nov 2015           20.45    Asheville 01-11-010 Municipality 2015-11-01
## 22   Apr 2015           21.60    Asheville 01-11-010 Municipality 2015-04-01
## 23   Aug 2015           24.11    Asheville 01-11-010 Municipality 2015-08-01
## 24   Dec 2015           19.88    Asheville 01-11-010 Municipality 2015-12-01
```

```
# Creating ggplot for Durham and Asheville's water withdrawls
AshevilleDurham2015Plot <-
  ggplot(AshevilleDurham2015, aes(x = Date, y = Monthly_Max_mgd,
                                  color = Water_System)) +
  geom_line() +
  labs(title = "Water Withdrawals of Durham and Asheville (2015)",
       x = "Date",
       y = "Maximum Daily Withdrawals (MGD)")
AshevilleDurham2015Plot
```

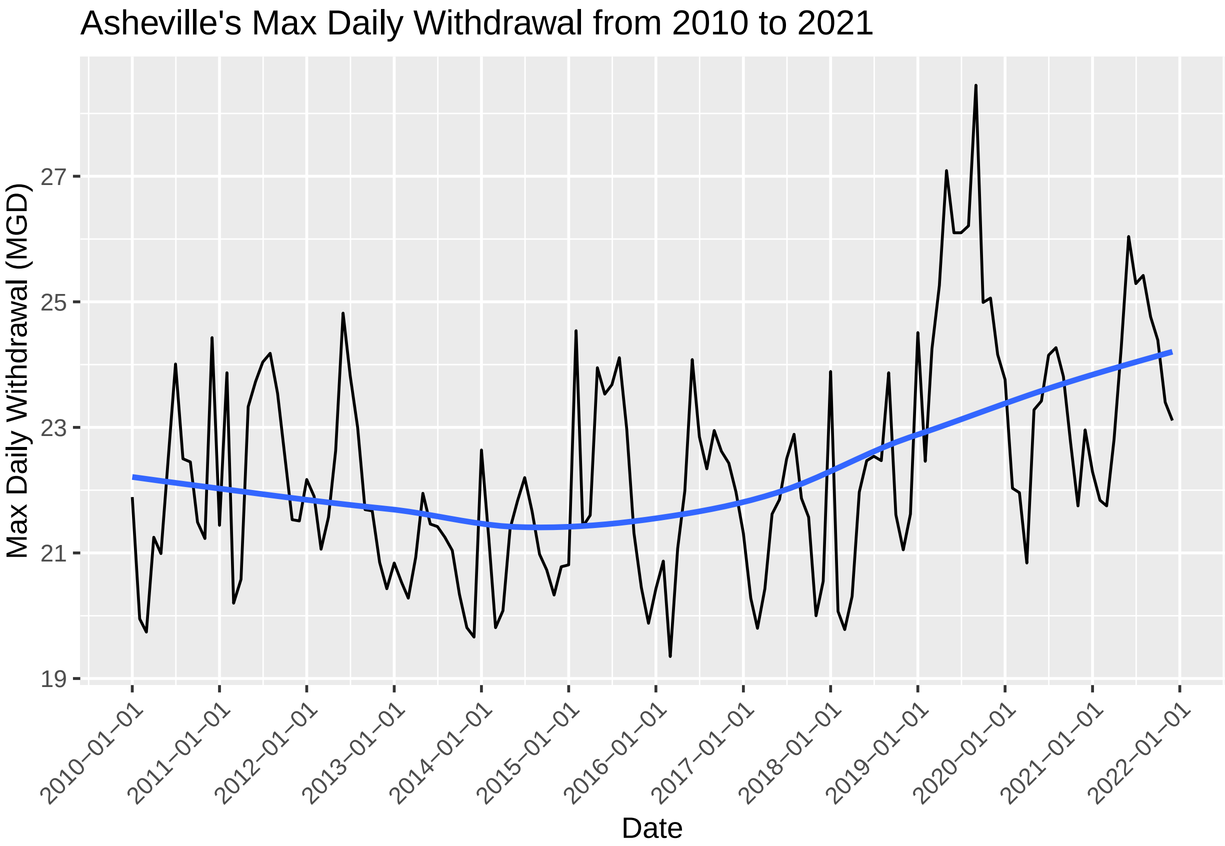


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   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.

```
#9

# Creating the sequence through 2010 to 2021
Ashevilles_df_years <- seq(2010,2021)

# Identifying the PWSID
Ashevilles_PWSID = '01-11-010'

#"Map" the scrape function to retrieve data for all these
dfs_Ashevilles_2010_2021 <- map2(Ashevilles_PWSID, Ashevilles_df_years, scrape_NCDEQ)

#Conflate the returned list of dataframes into a single one
single_df_Asheville_2010_2021<- bind_rows(dfs_Ashevilles_2010_2021)

#Plotting the dataframe
Asheville_2010_2021_plot <-
  ggplot(single_df_Asheville_2010_2021,aes(x = Date, y = Monthly_Max_mgd)) +
  geom_line() +
  scale_x_date(date_breaks="1 year") +
  theme(axis.text.x=element_text(angle = 45, hjust = 1)) +
  geom_smooth(method="loess",se=FALSE) +
    labs(title = paste("Asheville's Max Daily Withdrawal from 2010 to 2021"),
         x = "Date",
         y = "Max Daily Withdrawal (MGD)")


Asheville_2010_2021_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: From 2010 to 2018, the maximum day usage remained constant. But from 2019 and on, > water usage increased, with the peak occuring between 2019 and 2020.