Assignment 10: Data Scraping

Vicky Fong

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)
getwd()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2023 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=2023

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

```
#2
website <- read_html(
  'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2023'
)
```

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
water_system <- website %>%
  html_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>% html_text()
pwsid <- website %>%
  html_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html_text()
ownership <- website %>%
  html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>% html_text()
mgd <- 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.)

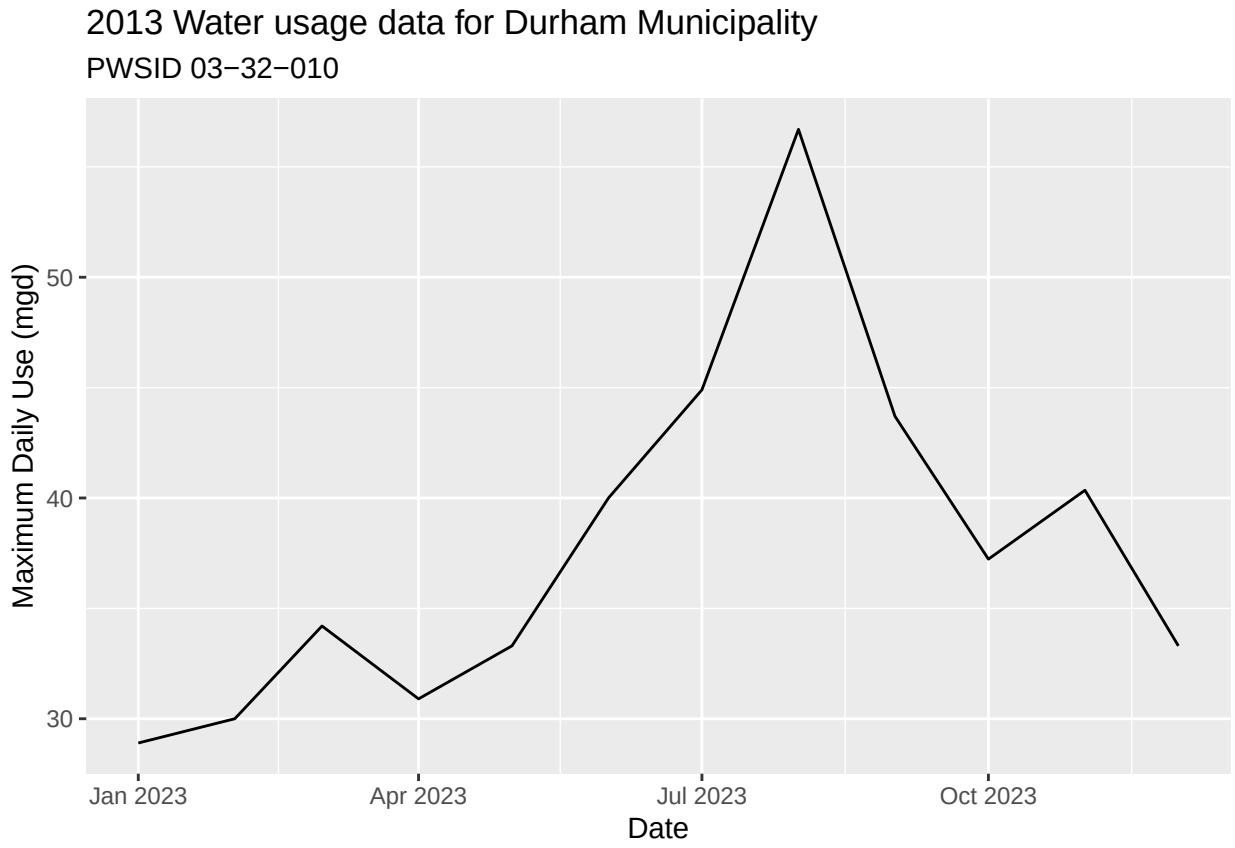
> TIP: Use `rep()` to repeat a value when creating a dataframe.

> 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 maximum daily withdrawals across the months for 2023, making sure, the months are presented in proper sequence.

```
#4
df <- data.frame("Month" =
 c("Jan","May","Sep","Feb","Jun","Oct","Mar","Jul","Nov","Apr","Aug","Dec"),
                  "Year" = rep(2023,12),
                  "Max_Day_Use_mgd" = as.numeric(mgd)) %>%
  mutate(Water_System = !!water_system,
         PWSID = !!pwsid,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year)))
```

```r
#5
ggplot(df,aes(x=Date,y=Max_Day_Use_mgd)) +
  geom_line() +
  labs(title = paste("2013 Water usage data for",water_system,ownership),
       subtitle = paste("PWSID",pwsid),
       y="Maximum Daily Use (mgd)",
       x="Date")
```



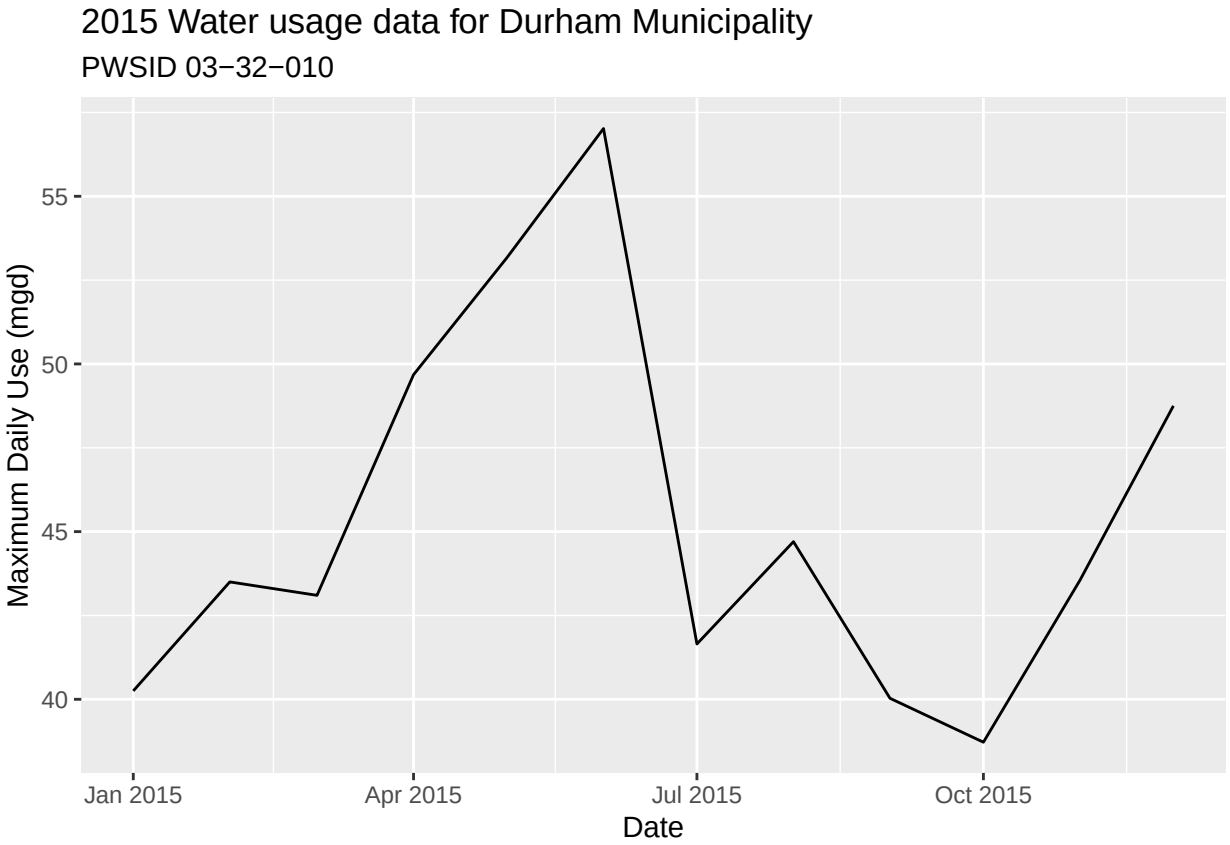
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data, returning a dataframe. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```r
#6.
#Retrieve the website contents
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 (determined in the previous step)
  the_water_system_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
  the_pwsid_tag <- 'td tr:nth-child(1) td:nth-child(5)'
```

```
  the_ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
  the_data_tag <- 'th~ td+ td'

  #Scrape the data items
  the_water_system <- the_website %>% html_nodes(the_water_system_tag) %>% html_text()
  the_pwsid <- the_website %>% html_nodes(the_pwsid_tag) %>% html_text()
  the_ownership <- the_website %>% html_nodes(the_ownership_tag) %>% html_text()
  mgd <- the_website %>% html_nodes(the_data_tag) %>% html_text()

  #Convert to a dataframe
  the_df <- data.frame("Month" =
              c("Jan","May","Sep","Feb","Jun","Oct","Mar","Jul","Nov","Apr","Aug","Dec"),
                  "Year" = rep(the_year,12),
                  "Max_Day_Use_mgd" = as.numeric(mgd)) %>%
  mutate(Water_System = !!the_water_system,
         PWSID = !!the_pwsid,
         Ownership = !!the_ownership,
         Date = my(paste(Month,"-",Year)))

  #Pause for a moment - scraping etiquette
  #Sys.sleep(1) #uncomment this if you are doing bulk scraping!

  #Return the plot
  return(the_df)
}
```

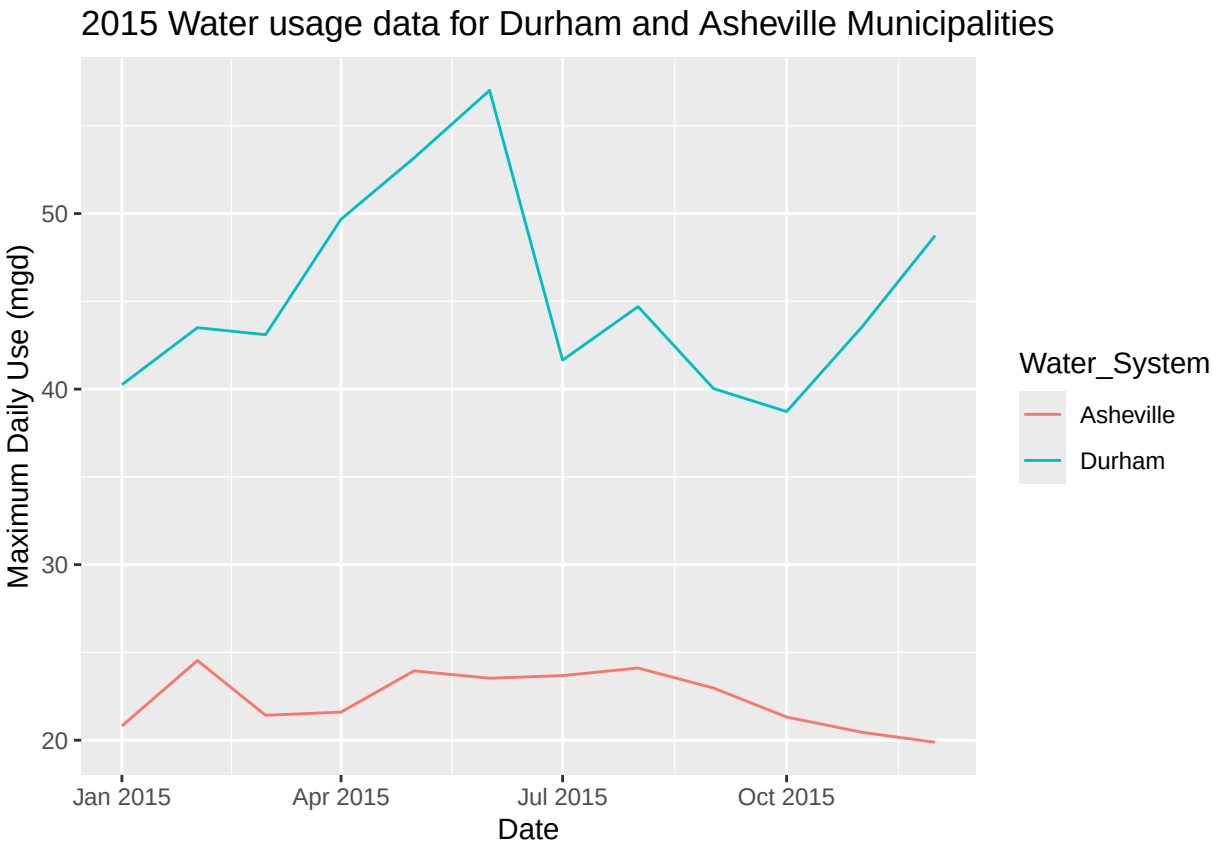
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

```
#7
the_pwsid = "03-32-010"
the_year = 2015
durham_2015_df <- scrape.it(the_pwsid, the_year)
durham_2015_df %>% ggplot(aes(x=Date,y=Max_Day_Use_mgd)) +
  geom_line() +
  labs(title = paste(the_year,"Water usage data for Durham Municipality"),
       subtitle = paste("PWSID",the_pwsid),
       y="Maximum Daily Use (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
#Scrape Asheville data
the_pwsid = "01-11-010"
the_year = 2015
asheville_2015_df <- scrape.it(the_pwsid, the_year)

#Combine data
combined_2015_df <-rbind(durham_2015_df,asheville_2015_df)

combined_2015_df %>% ggplot(aes(x=Date,y=Max_Day_Use_mgd, color=Water_System)) +
  geom_line() +
  labs(title = "2015 Water usage data for Durham and Asheville Municipalities",
       y="Maximum Daily Use (mgd)",
       x="Date")
```

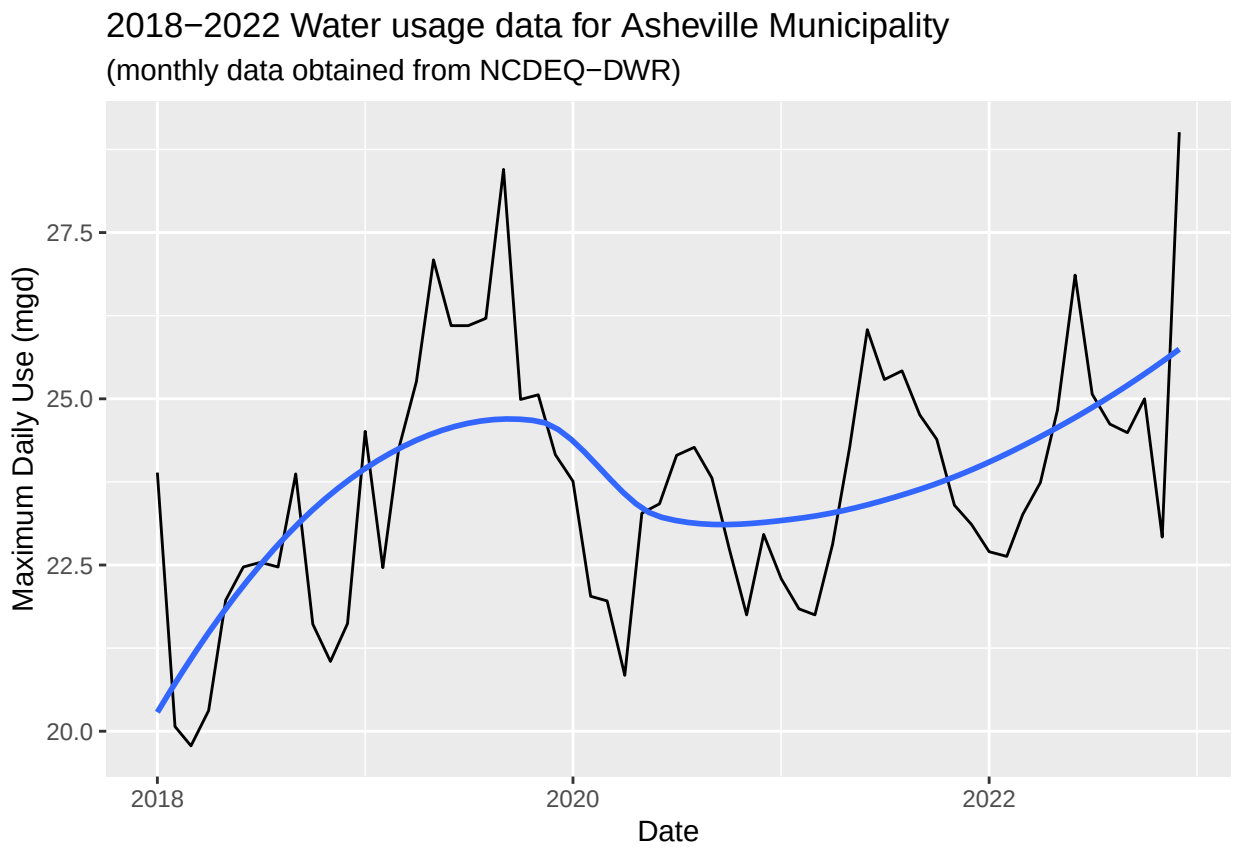


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2022.Add a smoothed line to the plot (method = 'loess').

   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
#Set tags
the_pwsid <- "01-11-010"
the_years = rep(2018:2022)

#"Map" the "scrape.it" function to retrieve data for all these
asheville_2018_2022_dfs <- map2(the_pwsid, the_years, scrape.it)

#Conflate the returned list of dataframes into a single one
asheville_2018_2022_df <- bind_rows(asheville_2018_2022_dfs)

#Plot data
asheville_2018_2022_df %>% ggplot(aes(x=Date,y=Max_Day_Use_mgd,)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = "2018-2022 Water usage data for Asheville Municipality",
       subtitle = "(monthly data obtained from NCDEQ-DWR)",
       y="Maximum Daily Use (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? > Answer: By visual analysis of the plot, Asheville's water usage has increased overall from 2018 to 2022. There was a strong increase from 2018 to 2020, decreasing slightly from 2020 to 2021, then increased again from 2021 to 2022. >