Assignment 09: Data Scraping

Laurel Cohen

Total points:

OVERVIEW

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

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Fay_09_Data_Scraping.Rmd") prior to submission.

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

```
#1
getwd()
```

```
## [1] "/Users/Laurel/Documents/Information to Keep/Graduate School/Second Year/Second Semester/Environ
```

```
library(tidyverse)
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
library(lubridate)
library(viridis)
library(rvest)
library(dataRetrieval)
```

```
## Warning: package 'dataRetrieval' was built under R version 4.0.5
```

```
library(tidycensus)
```

```
## Warning: package 'tidycensus' was built under R version 4.0.5
```

```
mytheme <- theme_classic() +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2020 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- Change the date from 2021 to 2020 in the upper right corner.
- 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=2020

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

```
#2
webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')
```

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:
- Max Daily Use (MGD) - for each month

In the code chunk below scrape these values, assigning them to three 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, with the first value being 36.0100.

```
#3
water.system.name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
pwsid <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
ownership <- webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
max.withdrawals.mgd <- webpage %>%
  html_nodes("th~ td+ td , th~ td+ td") %>%
  html_text()
```

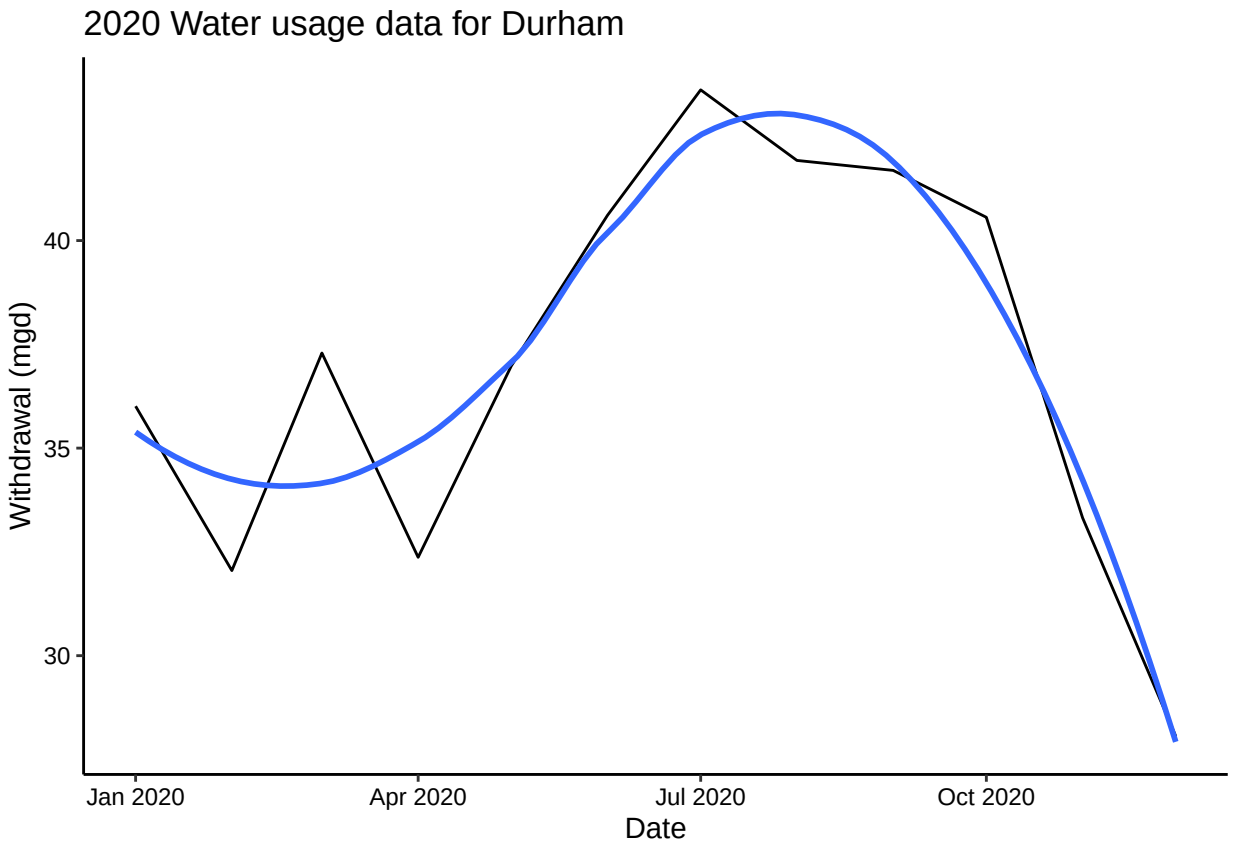
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   > 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 order. You can overcome this by creating a month column in the same order the data are scraped: Jan, May, Sept, Feb, etc. . .

5. Plot the max daily withdrawals across the months for 2020

```r
#4
df_withdrawals <- data.frame("Month" = c("Jan","May","Sept","Feb","June",
                                         "Oct","Mar","July","Nov","Apr","Aug","Dec"),
                             "Year" = rep(2020,12),
                             "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd),
                             "Water System Name" = water.system.name,
                             "PWS ID" = pwsid,
                             "Ownership" = ownership)

df_withdrawals <- df_withdrawals %>%
  mutate(Date = my(paste(Month,"-",Year)))

#5
ggplot(df_withdrawals,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2020 Water usage data for",water.system.name),
       y="Withdrawal (mgd)",
       x="Date")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



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 scraped**.

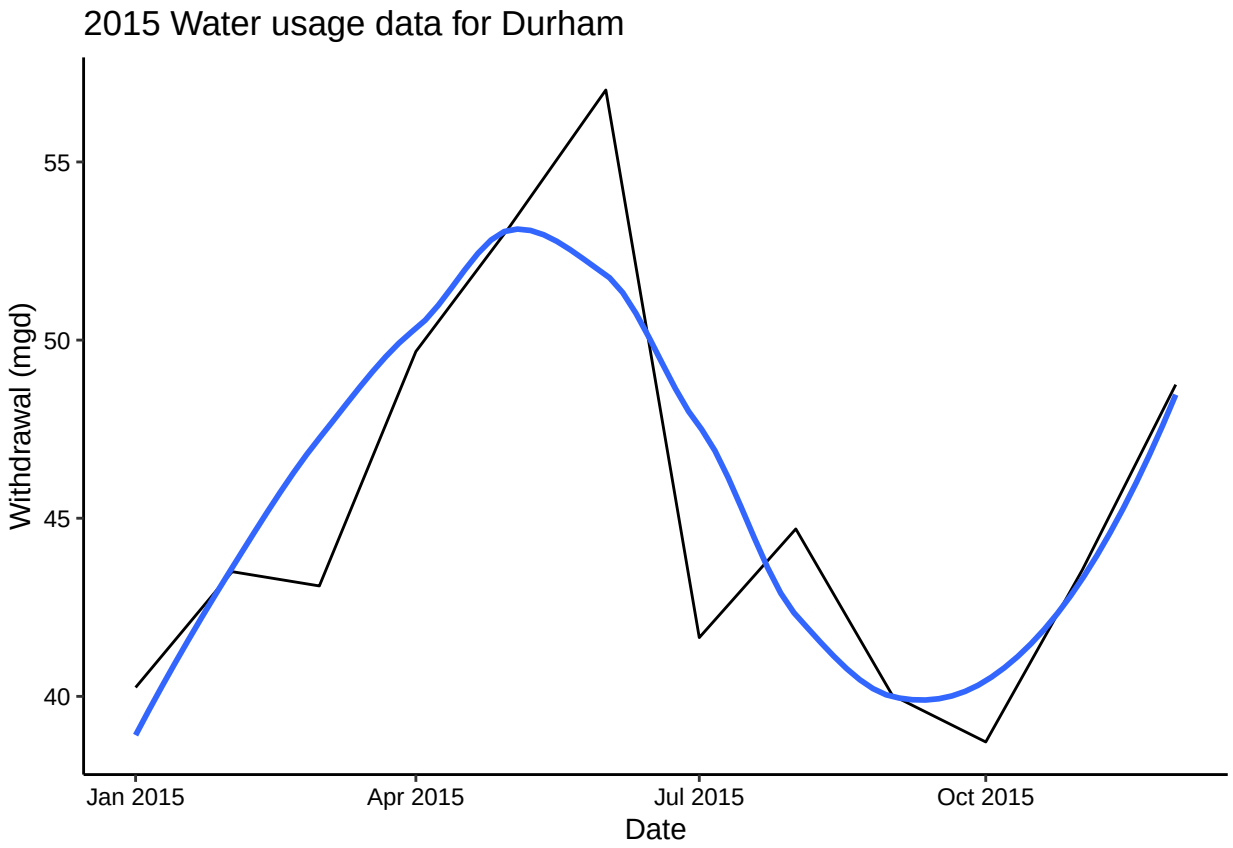
```r
#6.
scrape.it <- function(the_year, pwsid){
```

```
the_website <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php',
                                '?pwsid=', pwsid, '&year=', the_year))

water.system.name.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 , th~ td+ td'

water.system.name <- the_website %>% html_nodes(water.system.name.tag) %>% html_text()
pwsid <- the_website %>%  html_nodes(pwsid_tag) %>%  html_text()
ownership <- the_website %>% html_nodes(ownership_tag) %>% html_text()
max.withdrawals.mgd <- the_website %>% html_nodes(max.withdrawals.mgd.tag) %>% html_text()

df_withdrawals <- data.frame("Month" = c("Jan","May","Sept","Feb","June",
                                         "Oct","Mar","July","Nov","Apr","Aug","Dec"),
                             "Year" = rep(the_year,12),
                             "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd),
                             "Water System Name" = water.system.name,
                             "PWS ID" = pwsid,
                             "Ownership" = ownership)

df_withdrawals <- df_withdrawals %>%
  mutate(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
durham_2015 <- scrape.it(2015,'03-32-010')

ggplot(durham_2015,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2015 Water usage data for",water.system.name),
       y="Withdrawal (mgd)",
       x="Date")
```

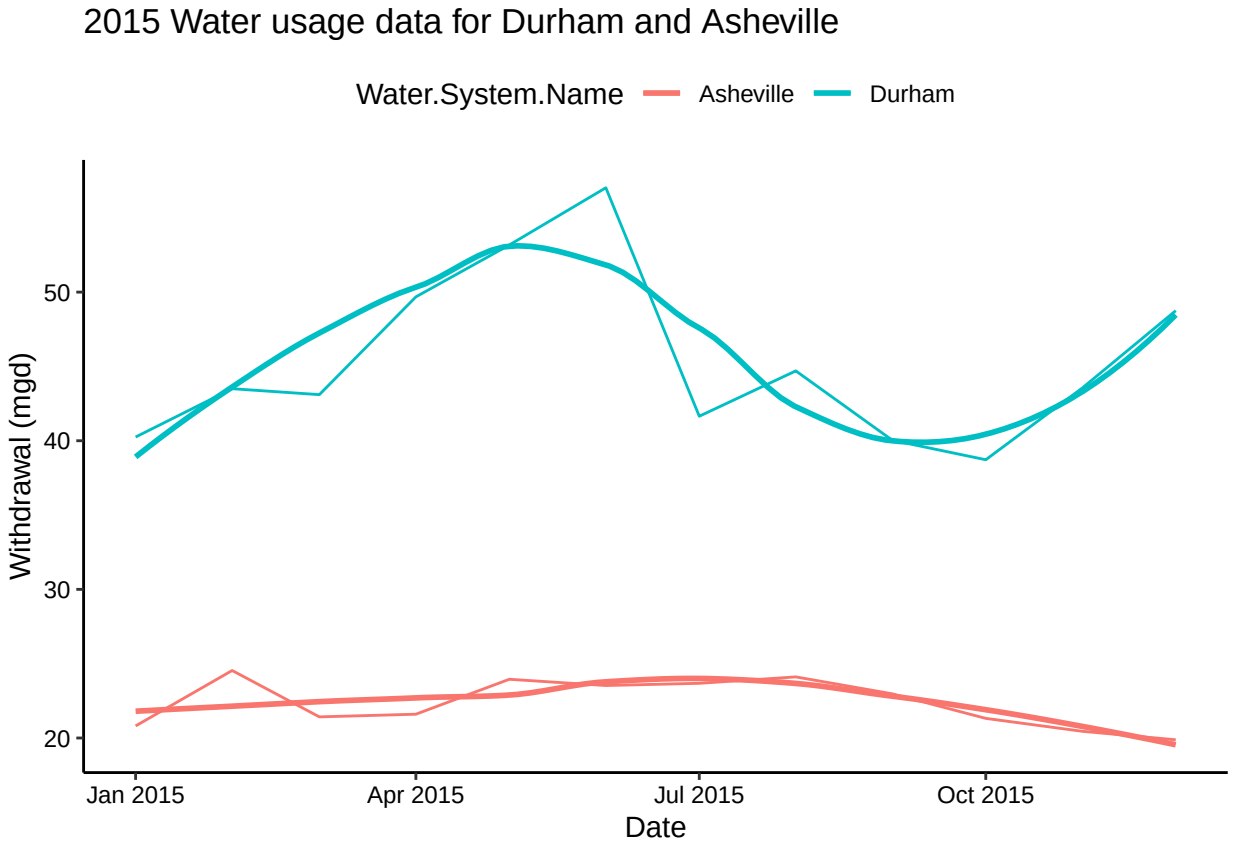
```
## `geom_smooth()` using formula 'y ~ x'
```



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 the Asheville to Durham's water withdrawals.

```
#8
asheville_df <- scrape.it(2015,'01-11-010')

combined_df <- rbind(durham_2015, asheville_df)

ggplot(combined_df,aes(x=Date,y=Max_Withdrawals_mgd,color=Water.System.Name)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = "2015 Water usage data for Durham and Asheville",
       y="Withdrawal (mgd)",
       x="Date")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

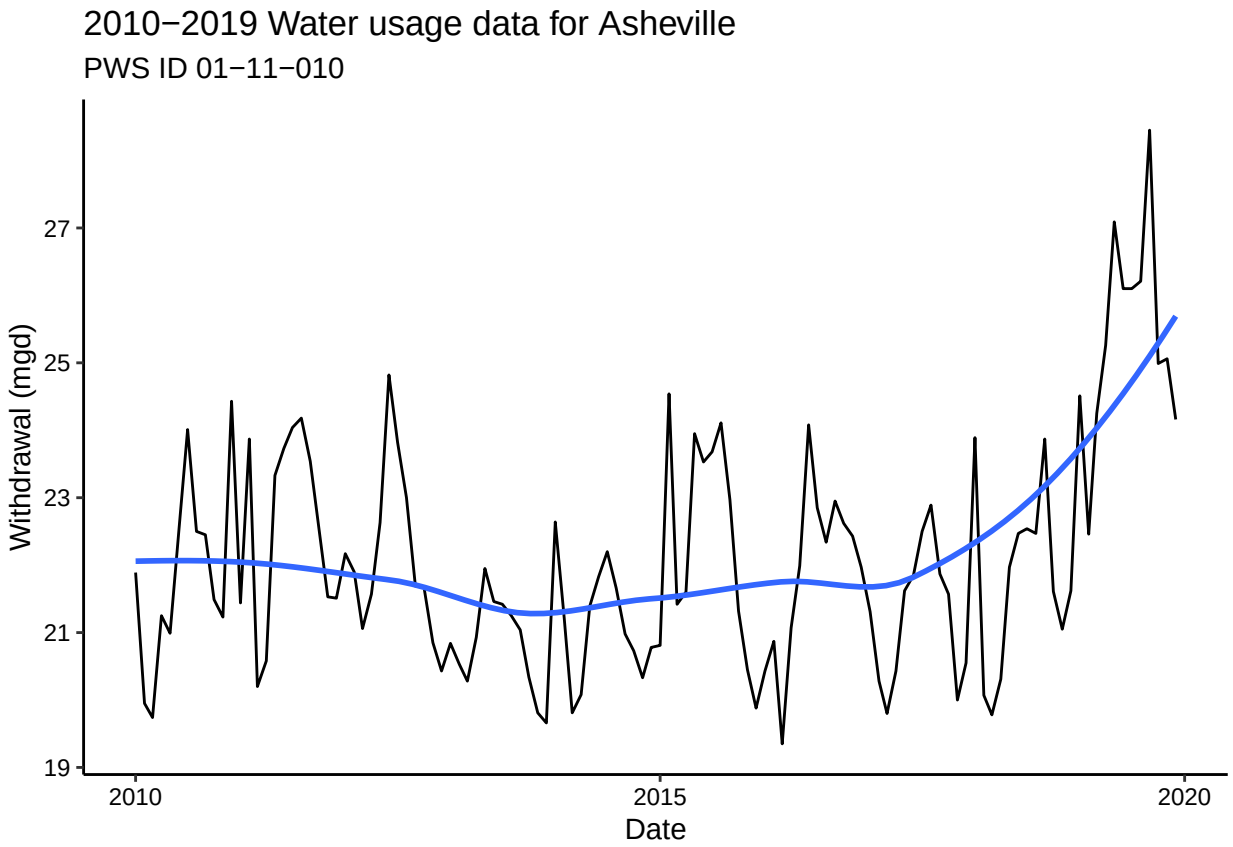


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019. Add a smoothed line to the plot.

```
#9
the_years = rep(2010:2019)
my_facility = '01-11-010'

df_asheville_years <- lapply(X = the_years,
                  FUN = scrape.it,
                  pwsid=my_facility)

df_single_asheville_years <- bind_rows(df_asheville_years,.id="Water.System.Name")

#Replacing numbers in "Water System Name" variable column with "Asheville"
df_single_asheville_years <- df_single_asheville_years %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 1, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 2, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 3, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 4, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 5, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 6, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 7, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 8, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 9, "Asheville")) %>%
  mutate(Water.System.Name = replace(Water.System.Name, Water.System.Name == 10, "Asheville"))

ggplot(df_single_asheville_years,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
```

```
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2010-2019 Water usage data for",
                     df_single_asheville_years$Water.System.Name),
       subtitle = paste("PWS ID",df_single_asheville_years$PWS.ID),
       y="Withdrawal (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?

Yes, the trend in Asheville's water usage over time is that it is increasing.