Assignment 09: Data Scraping

Yunting Wang

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] "\\\\homedir.oit.duke.edu/users/y/yw448/RforENV872/Environmental_Data_Analytics_2022/Assignments
```

```
library(tidyverse)
library(rvest)
library(lubridate)
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 2019 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- Change the date from 2020 to 2019 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
the.url=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
- PSWID
- Ownership
- From the "3. Water Supply Sources" section:
- Average 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 <- the.url %>%
  html_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>%
  html_text()
water.system.name
```

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

```
pswid <- the.url %>%
  html_nodes('td tr:nth-child(1) td:nth-child(5)') %>%
  html_text()
pswid
```

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

```
ownership <- the.url %>%
  html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>%
  html_text()
ownership
```

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

```
max.withdrawals.mgd <- the.url %>%
  html_nodes('th~ td+ td') %>%
  html_text()
max.withdrawals.mgd
```

```
##  [1] "36.0100" "36.9800" "41.6900" "32.0500" "40.6100" "40.5600" "37.2900"
##  [8] "43.6300" "33.3200" "32.3700" "41.9300" "28.0600"
```

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.)

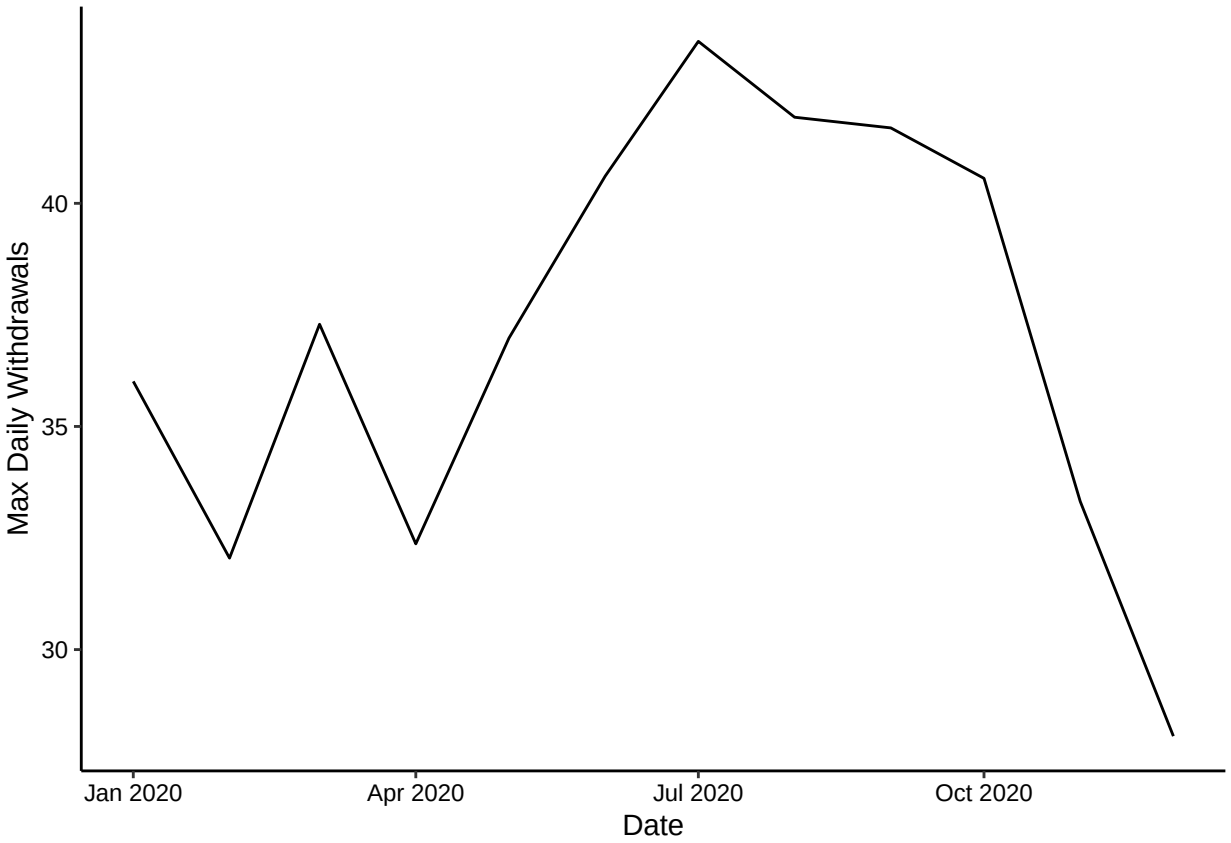
   > TIP: Use `rep()` to repeat a value when creating a dataframe.
   >
   > NOTE: It's likely you won't be able to scrape the monthly widthrawal data in order. You can overcome this by creating a month column in the same order the data are scraped: Jan, May, Sept, Feb, etc. . .

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

```
#4
the.month=c(1,5,9,2,6,10,03,7,11,4,8,12)
the.year=rep(2020,12)
Date=paste0(the.year,"-",the.month)
the.date=my(paste(the.month,"-",the.year))

the.ownership=rep(ownership,12)
pwsid=rep(pswid,12)
ex4=data.frame(Water.System.Name=rep(water.system.name,12),
               Ownership=rep(ownership,12),
               PWSID=rep(pswid,12),
               Max.Day.Use=as.numeric(max.withdrawals.mgd),
               Date=the.date)
ex4
```

```
##    Water.System.Name    Ownership     PWSID Max.Day.Use       Date
## 1             Durham Municipality 03-32-010       36.01 2020-01-01
## 2             Durham Municipality 03-32-010       36.98 2020-05-01
## 3             Durham Municipality 03-32-010       41.69 2020-09-01
## 4             Durham Municipality 03-32-010       32.05 2020-02-01
## 5             Durham Municipality 03-32-010       40.61 2020-06-01
## 6             Durham Municipality 03-32-010       40.56 2020-10-01
## 7             Durham Municipality 03-32-010       37.29 2020-03-01
## 8             Durham Municipality 03-32-010       43.63 2020-07-01
## 9             Durham Municipality 03-32-010       33.32 2020-11-01
## 10            Durham Municipality 03-32-010       32.37 2020-04-01
## 11            Durham Municipality 03-32-010       41.93 2020-08-01
## 12            Durham Municipality 03-32-010       28.06 2020-12-01
```

```
#5
ex5=ggplot(ex4,aes(x=Date,y=Max.Day.Use))+
  geom_line()+
  ylab("Max Daily Withdrawals")

print(ex5)
```



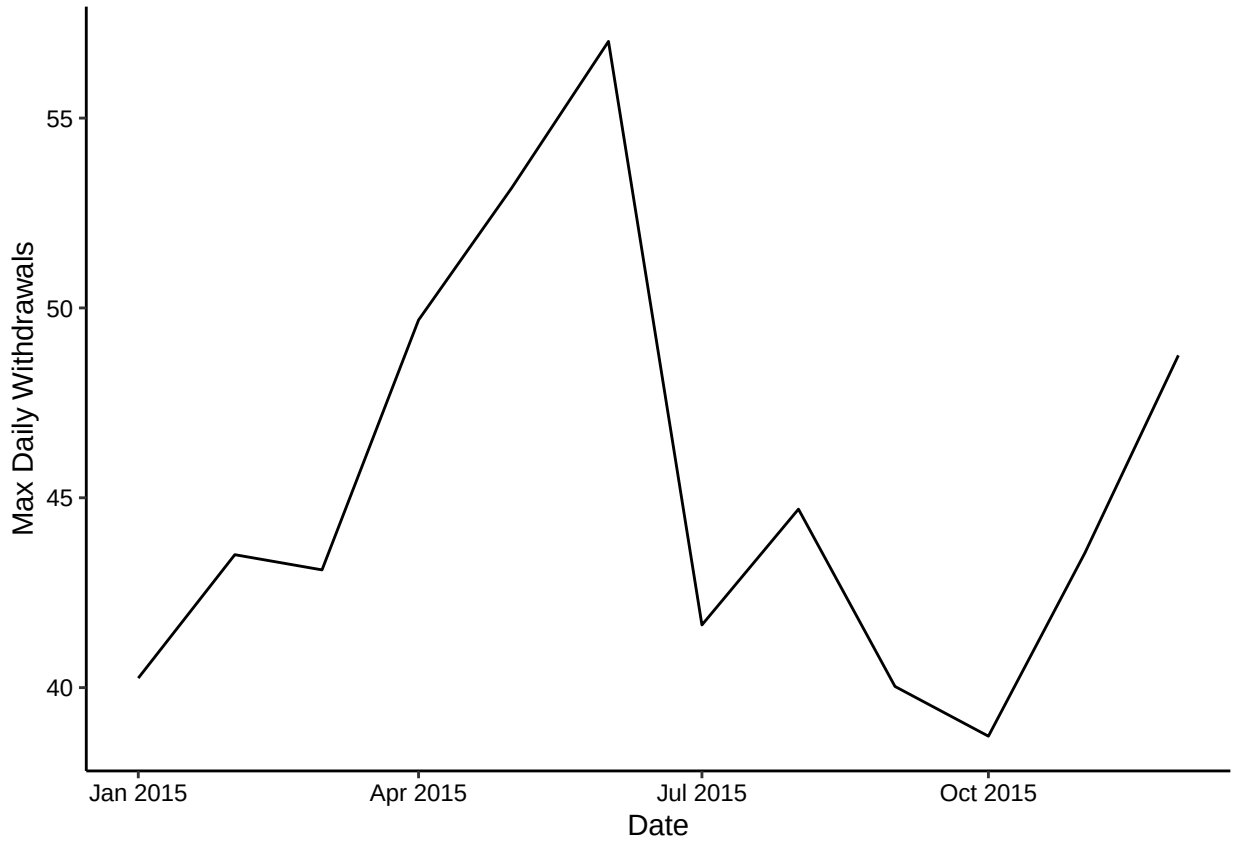
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site scraped**.

```
#6.
scrap.it=function(the_pwsid,the_year){
  the_website= read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',
                                the_pwsid,'&year=',the_year))

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


ownership <- the_website %>%
  html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>%
  html_text()

max.withdrawals.mgd <- the_website %>%
  html_nodes('th~ td+ td') %>%
  html_text()

the.month=c(1,5,9,2,6,10,03,7,11,4,8,12)
the.year=rep(the_year,12)
the.date=my(paste(the.month,"-",the_year))

the_df=data.frame(Water.System.Name=rep(water.system.name,12),
```

```
                Ownership=rep(ownership,12),
                PWSID=rep(the_pwsid,12),
                Max.Day.Use=as.numeric(max.withdrawals.mgd),
                Date=the.date)
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

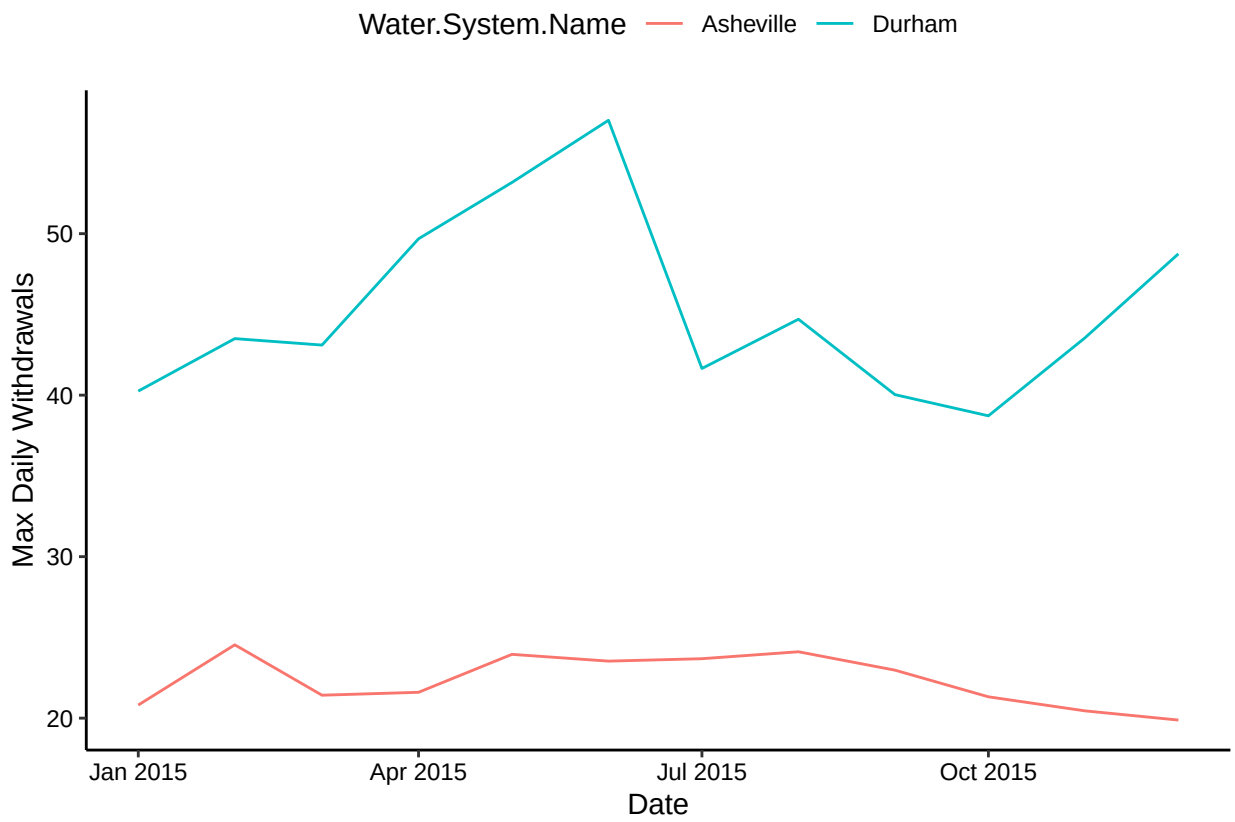
```
#7
ex7=scrap.it('03-32-010',2015)
ex7.plot=ggplot(ex7,aes(x=Date,y=Max.Day.Use))+
  geom_line()+
  ylab("Max Daily Withdrawals")
print(ex7.plot)
```



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
ex8=scrap.it('01-11-010',2015)
ex8.combined=rbind(ex7,ex8)
ex8.plot=ggplot(ex8.combined,
                aes(x=Date,
                    y=Max.Day.Use,
```

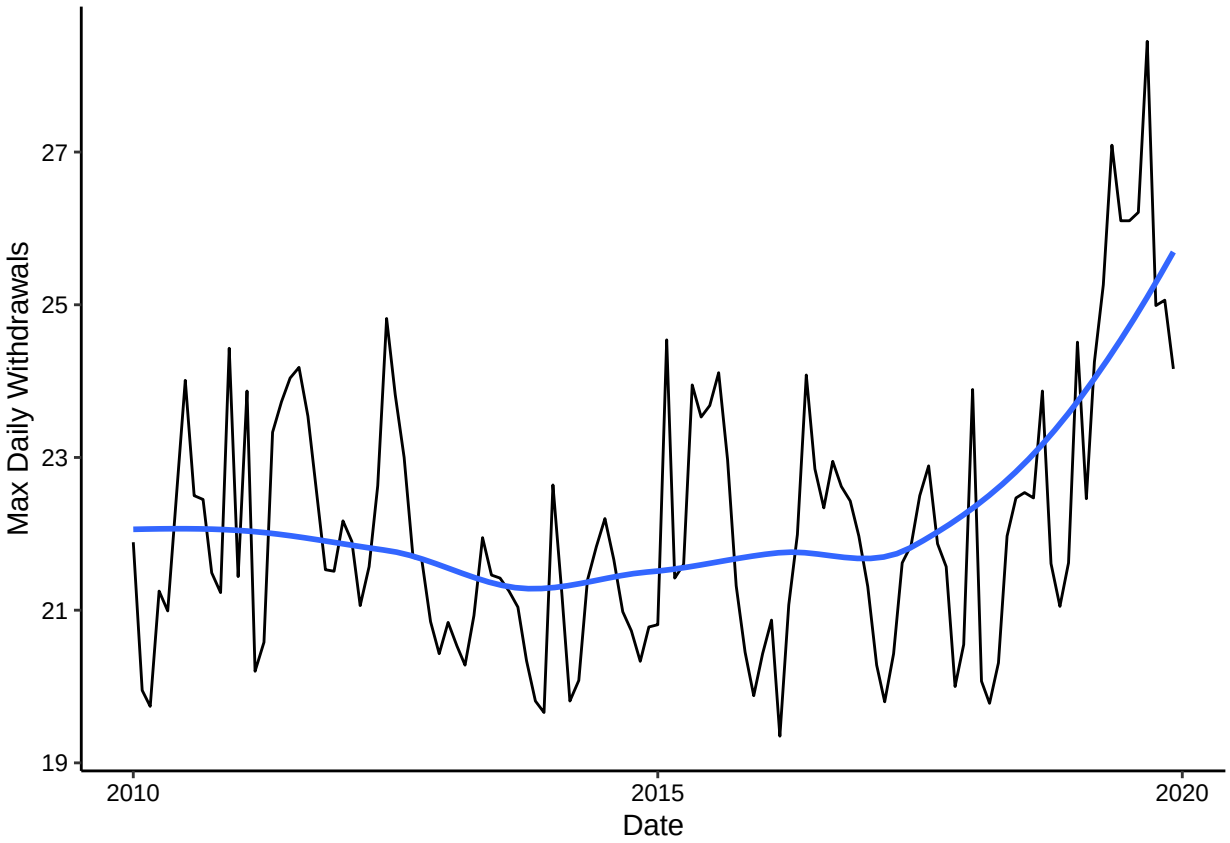
```
                    color=Water.System.Name))+
  geom_line()+
  ylab("Max Daily Withdrawals")
print(ex8.plot)
```



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=seq(2010,2019)
my_pwsid='01-11-010'
ex9=lapply(X=the_years,
           FUN=scrap.it,
           the_pwsid=my_pwsid) %>% bind_rows()
ex9.plot=ggplot(ex9,aes(x=Date,
                    y=Max.Day.Use))+
  geom_line()+
  geom_smooth(method="loess",se=FALSE) +
  ylab("Max Daily Withdrawals")
print(ex9.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? Yes, the water usage in Asheville has a rising trend.