Assignment 09: Data Scraping

Sam Saltman

# 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/samsaltman/Documents/R/Environmental\_Data\_Analytics\_2022"

library(tidyverse)  
library(lubridate)  
library(viridis)  
library(rvest)  
library(dataRetrieval)  
library(tidycensus)  
  
# Set theme  
mytheme <- theme\_classic() +  
 theme(axis.text = element\_text(color = "black"),   
 legend.position = "top")  
theme\_set(mytheme)

1. 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\_website\_2020 <- read\_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')  
the\_website\_2020

## {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= ...

1. 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\_website\_2020 %>% html\_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>% html\_text()  
pswid <- the\_website\_2020 %>% html\_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html\_text()  
ownership <- the\_website\_2020 %>% html\_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>% html\_text()  
average.daily.mgd <- the\_website\_2020 %>% html\_nodes('.fancy-table:nth-child(31) th+ td') %>% html\_text()  
max.withdrawals.mgd <- the\_website\_2020 %>% html\_nodes('th~ td+ td') %>% html\_text()

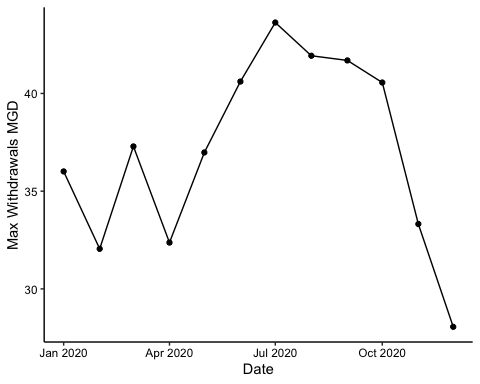
1. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

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…

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

#4  
  
the\_df\_2020 <- data.frame("Water System Name" = water.system.name,  
 "PWSID" = pswid,  
 "Ownership" = ownership,  
 "Max Withdrawals MGD" = as.numeric(max.withdrawals.mgd),  
 "Month" = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12),  
 "Year" = rep(2020, 12))  
the\_df\_2020 <- the\_df\_2020 %>%  
 mutate(Date = my(paste(Month,"-",Year))) %>%  
 select("Water.System.Name", "PWSID","Ownership", "Max.Withdrawals.MGD", "Date")  
  
  
#5  
plot\_2020 <- ggplot(the\_df\_2020, aes(x = Date, y = Max.Withdrawals.MGD)) +  
 geom\_point() +  
 geom\_line() +  
 ylab("Max Withdrawals MGD")  
print(plot\_2020)

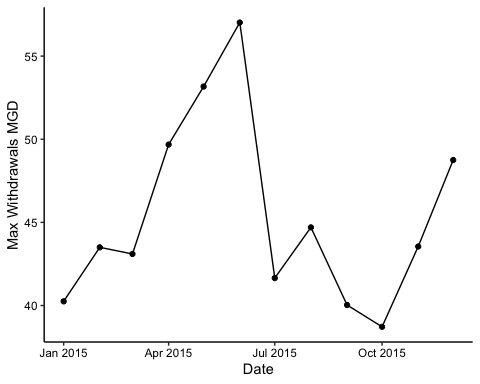


1. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site scraped**.

#6.  
scrape.it <- function(pwsid, year\_1){  
   
URL <- read\_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',pwsid,'&year=',year\_1))  
  
water.system.name <- URL %>% html\_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>% html\_text()  
pswid <- URL %>% html\_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html\_text()  
ownership <- URL %>% html\_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>% html\_text()  
average.daily.mgd <- URL %>% html\_nodes('.fancy-table:nth-child(31) th+ td') %>% html\_text()  
max.withdrawals.mgd <- URL %>% html\_nodes('th~ td+ td') %>% html\_text()  
  
the\_df\_all <- data.frame("Water System Name" = water.system.name,  
 "PWSID" = pswid,  
 "Ownership" = ownership,  
 "Max Withdrawals MGD" = as.numeric(max.withdrawals.mgd),  
 "Month" = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12),  
 "Year" = rep(year\_1, 12))  
the\_df\_all <- the\_df\_all %>%  
 mutate(Date = my(paste(Month,"-",Year))) %>%  
 select("Water.System.Name", "PWSID","Ownership", "Max.Withdrawals.MGD", "Date")  
  
return(the\_df\_all)}

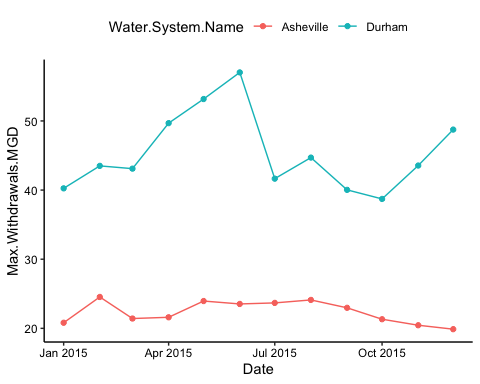
1. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

#7  
Durham\_PWSID0332010\_Year2015 <- scrape.it('03-32-010',2015)  
  
view(Durham\_PWSID0332010\_Year2015)  
  
Plot\_Durham\_PWSID0332010\_Year2015 <- ggplot(Durham\_PWSID0332010\_Year2015, aes(x = Date, y = Max.Withdrawals.MGD)) +  
 geom\_line() +  
 geom\_point() +  
 ylab("Max Withdrawals MGD")  
print(Plot\_Durham\_PWSID0332010\_Year2015)



1. 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.

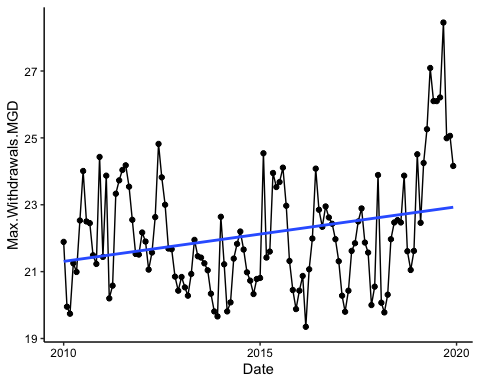
#8  
Asheville\_PWSID0111010\_Year2015 <- scrape.it('01-11-010',2015)  
view(Asheville\_PWSID0111010\_Year2015)  
  
Asheville\_Durham\_2015 <- rbind(Asheville\_PWSID0111010\_Year2015,Durham\_PWSID0332010\_Year2015)  
  
Plot\_Asheville\_Durham\_2015 <- ggplot(Asheville\_Durham\_2015,aes(x = Date, y = Max.Withdrawals.MGD, color = Water.System.Name)) +  
 geom\_point()+  
 geom\_line()  
print(Plot\_Asheville\_Durham\_2015)



1. 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  
PWSID\_iterate <- '01-11-010'  
Years\_iterate <- rep(2010:2019)  
  
Asheville\_2010\_2019 <- map2(PWSID\_iterate,Years\_iterate, scrape.it)  
view(Asheville\_2010\_2019)  
  
Asheville\_Bind\_Years <- bind\_rows(Asheville\_2010\_2019)  
  
Plot\_Asheville\_2010\_2019 <- ggplot(Asheville\_Bind\_Years,aes(x = Date, y = Max.Withdrawals.MGD)) +  
 geom\_point()+  
 geom\_line() +  
 geom\_smooth(se = FALSE, method = "lm")  
print(Plot\_Asheville\_2010\_2019)

## `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 plot suggests that water usage has been increasing since 2010