Assignment 10: Data Scraping

Samantha White-Murillo

## 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)  
library(lubridate)  
library(viridis)  
library(here)  
library(dataRetrieval)  
library(tidycensus)  
library(dplyr)  
  
here()

## [1] "/home/guest/EDA\_Spring2024\_SamanthaWM"

1. 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 URL to be scraped. (In other words, read the contents into an rvest webpage object.)

#2   
Durham.LWSP <- read\_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')  
  
Durham.LWSP

## {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
* 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.name <- Durham.LWSP %>%  
 html\_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%  
 html\_text()  
Water.system.name

## [1] "Durham"

PWSID <- Durham.LWSP %>%  
 html\_nodes("td tr:nth-child(1) td:nth-child(5)") %>%  
 html\_text()  
PWSID

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

Ownership <- Durham.LWSP %>%  
 html\_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%  
 html\_text()  
Ownership

## [1] "Municipality"

MGD <- Durham.LWSP %>%  
 html\_nodes("th~ td+ td") %>%  
 html\_text()  
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"

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.

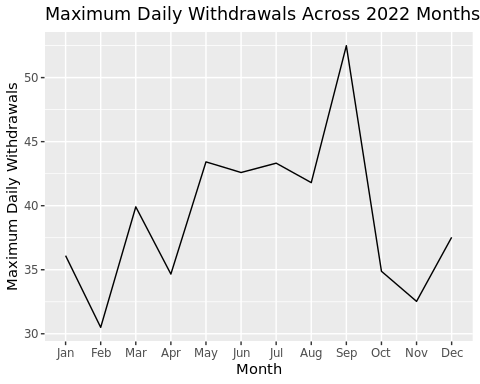
NOTE: It’s likely you won’t be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: “Jan”, “May”, “Sept”, “Feb”, etc… Or, you could scrape month values from the web page…

month.order <- c("Jan", "May", "Sep", "Feb", "Jun", "Oct", "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")  
  
Durham.LWSP.data <- data.frame("Month" = rep(1:12),  
 "Year" = rep(2022,12),  
 "Maximum.Day.Use" = as.numeric(MGD)) %>%  
 mutate('Water System' = !!Water.system.name,  
 Ownership = !!Ownership,  
 PWSID = !!PWSID,  
 Month = factor(Month,   
 levels = 1:12,   
 labels = month.order),  
 Date = my(paste(Month,"-",Year)))  
  
  
month.order.2 <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")  
  
Durham.LWSP.data <- Durham.LWSP.data %>%  
 mutate(Month = factor(Month, levels = month.order.2)) %>%  
 arrange(Month)  
  
view(Durham.LWSP.data)

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

#5  
ggplot(Durham.LWSP.data, aes(x = Month, y = `Maximum.Day.Use`)) +  
 geom\_line(aes(group = 1)) +   
 geom\_smooth(method="loess", se=FALSE) +  
 labs(title = "Maximum Daily Withdrawals Across 2022 Months ",  
 x = "Month",  
 y = "Maximum Daily Withdrawals")

## `geom\_smooth()` using formula = 'y ~ x'



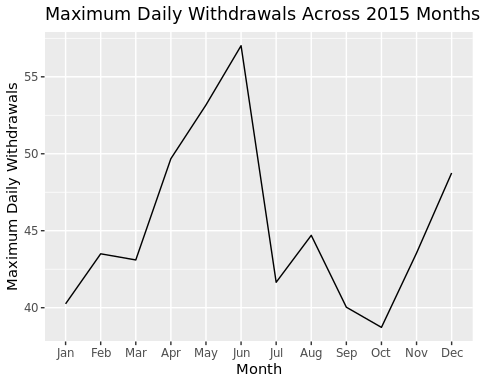
1. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

#6.   
Durham.LWSP.scraped <- function(PWSID, the\_year){  
   
 Durham.LWSP.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)'  
 MGD\_tag <- 'th~ td+ td'  
   
 Water.system.name\_val <- Durham.LWSP.website %>% html\_nodes(Water.system.name\_tag) %>% html\_text()  
 PWSID\_val <- Durham.LWSP.website %>% html\_nodes(PWSID\_tag) %>% html\_text()  
 Ownership\_val <- Durham.LWSP.website %>% html\_nodes(Ownership\_tag) %>% html\_text()  
 MGD\_val <- Durham.LWSP.website %>% html\_nodes(MGD\_tag) %>% html\_text()  
   
 month.order <- c("Jan", "May", "Sep", "Feb", "Jun", "Oct", "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")  
   
 S.Durham.LWSP.data <- data.frame("Month" = rep(1:12),  
 "Year" = rep(the\_year,12),  
 "Maximum.Day.Use" = as.numeric(MGD\_val)) %>%   
 mutate('Water System' = Water.system.name\_val,  
 Ownership = Ownership\_val,  
 PWSID = PWSID\_val,  
 Month = factor(Month,  
 levels = 1:12,   
 labels = month.order),  
 Date = paste(Month,"-",the\_year, sep=""))  
   
 month.order.2 <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")  
   
 S.Durham.LWSP.data <- S.Durham.LWSP.data %>%  
 mutate(Month = factor(Month, levels = month.order.2)) %>%  
 arrange(Month)  
   
 return(S.Durham.LWSP.data)  
}

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

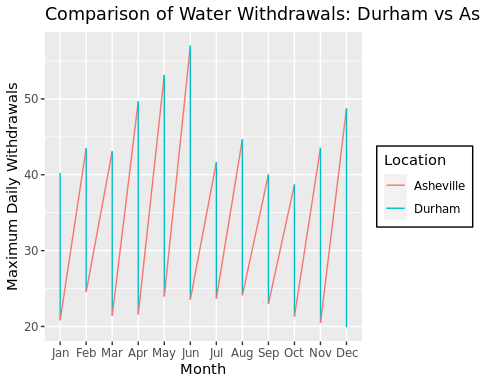
#7   
Durham.LWSP.Data.2015 <- Durham.LWSP.scraped(PWSID = "03-32-010", the\_year = 2015)  
  
ggplot(Durham.LWSP.Data.2015, aes(x = Month, y = `Maximum.Day.Use`)) +  
 geom\_line(aes(group = 1)) +   
 geom\_smooth(method="loess", se=FALSE) +  
 labs(title = paste("Maximum Daily Withdrawals Across 2015 Months"),  
 x = "Month",  
 y = "Maximum Daily Withdrawals")

## `geom\_smooth()` using formula = 'y ~ x'



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 Asheville’s to Durham’s water withdrawals.

#8   
Ashville.LWSP.Data.2015 <- Durham.LWSP.scraped(PWSID = "01-11-010", the\_year = 2015)  
  
Combined <- rbind(Durham.LWSP.Data.2015, Ashville.LWSP.Data.2015)  
  
ggplot(Combined, aes(x = Month, y = Maximum.Day.Use, color = `Water System`)) +  
 geom\_line(aes(group = 1)) +  
 labs(title = "Comparison of Water Withdrawals: Durham vs Asheville",  
 x = "Month",  
 y = "Maximum Daily Withdrawals",  
 color = "Location") +  
 theme(legend.position = "right",  
 legend.background = element\_rect(fill = "white", colour = 1))

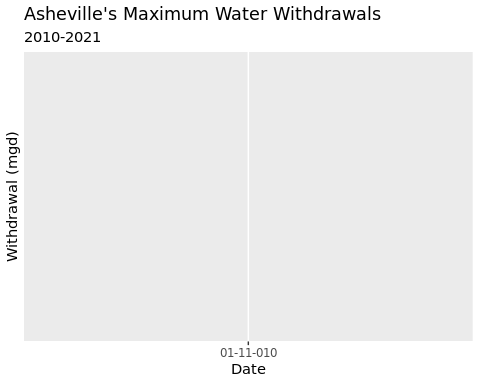


1. Use the code & function you created above to plot Asheville’s max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = ‘loess’).

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   
PWSID <- '01-11-010'  
the\_year <- rep(2010:2021)  
the\_months = rep(1:12)  
  
the\_dfs <- lapply(X = the\_year, FUN = Durham.LWSP.scraped, PWSID = PWSID) %>%  
 bind\_rows()  
  
MDW <- map2(the\_year, PWSID, Durham.LWSP.scraped) %>%   
 bind\_rows()  
  
 ggplot(MDW, aes(x = Year, y = Maximum.Day.Use)) +  
 geom\_line() +  
 geom\_smooth(method = "loess", se = FALSE) +  
 labs(title = paste("Asheville's Maximum Water Withdrawals"),  
 subtitle = "2010-2021",  
 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? > Answer: > Yes, it there is a growth of water use since 2014-2015.