Assignment 10: Data Scraping

Reed Leon-Hinton

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_10_Data_Scraping.Rmd") prior to submission.

The completed exercise is due on Tuesday, April 6 at 11:59 pm.

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

```
# clearing the environment (It's a pet peeve)
remove(list = ls())

# clearing the console and plots so looking at the code when ran
# from start to finish is cleaner.
graphics.off()
cat("\f")
```

```
#1

# checking the working directory
getwd()
```

```
## [1] "C:/Users/shado/Documents/Graduate School Stuff/ENVIRON 872 - Environmental Data Analytics/Envir
```

```
# adding the necessary packages to the library
library(tidyverse)
library(rvest)

# the ever-classic making a comeback
super_cool_theme_mk3 <- theme_classic(base_size = 13) +
  theme(axis.text = element_text(color = "firebrick3"),
        legend.position = "top",
        axis.title = element_text(color = "black"),
        panel.background = element_rect(fill = "white", color = "black"),
        panel.grid.major = element_line(size = 0.5, color = "gray93"))
theme_set(super_cool_theme_mk3)
```

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=2019

Indicate this website as the as the URL to be scraped.

```
#2 setting the scraping URL
theURL <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2019'
webpage <- read_html(theURL)
```

3. The data we want to collect are listed below:

- From the "System Information" section:
- Water system name
- PSWID
- Ownership
- From the "Water Supply Sources" section:
- Maximum monthly withdrawals (MGD)

In the code chunk below scrape these values into the supplied variable names.

```
#3

# scraping the Water System Name
water_sys_name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water_sys_name
```

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

```
# scraping the PSWID
pswid <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
pswid
```

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

```
# scraping the Ownership
ownership <- webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
ownership
```

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

```
# scraping the Maximum monthly withdrawals (MGD)
mgd <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text() %>%
  as.numeric()
mgd
```

```
##  [1] 29.62 35.73 54.07 32.39 37.86 44.35 36.43 46.02 36.06 32.60 42.05 31.20
```

```
# scraping the ordered months for creation of the dataframe
month <- webpage %>%
  html_nodes(".fancy-table:nth-child(31) tr+ tr th") %>%
  html_text()
month
```

```
##  [1] "Jan" "May" "Sep" "Feb" "Jun" "Oct" "Mar" "Jul" "Nov" "Apr" "Aug" "Dec"
```

```
# fun fact, this is the only field for which the "tag" changes between pages!
# therefore, you cannot use this in the function creation later,
# which I learned after many errors... Just an FYI
```

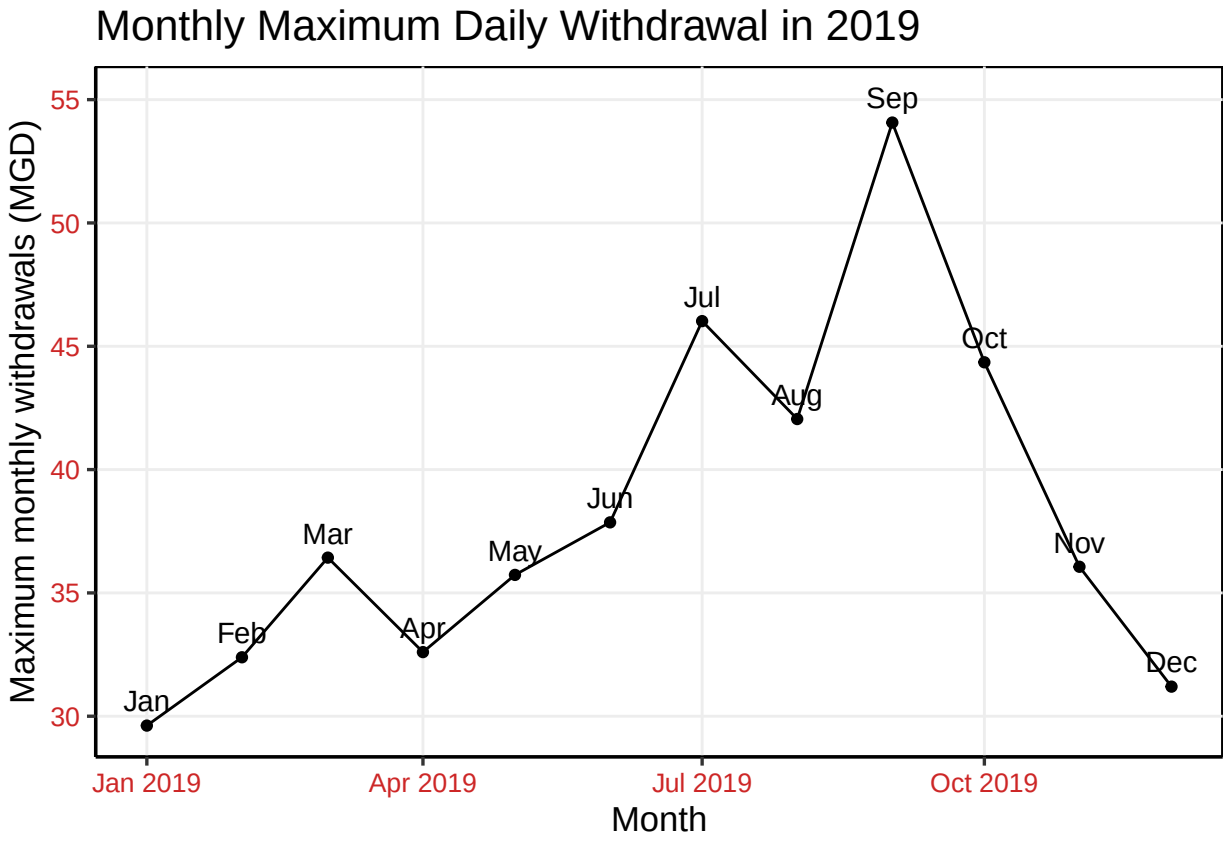
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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 2019.

```
#4

# creating the year column
year <- 2019

# creating a day column in order for the date to work
day <- 1

# creating the data frame
durham_water <- data.frame(water_sys_name, pswid, ownership,
                           mgd, month, year, day)
```

```
durham_water$date <- as.Date(with(durham_water, paste(year, month, day, sep = "-")),"%Y-%b-%d")

# removing the unnecessary columns
durham_water <-
  durham_water %>%
  select(date, month, water_sys_name, pswid, ownership, mgd)

#5
mgd_graph <- ggplot(data = durham_water, aes(x = date, y = mgd)) +
  geom_line() +
  geom_point() +
  geom_text(aes(label = month), position = position_nudge(y = 1)) +
  labs(y = "Maximum monthly withdrawals (MGD)", x = "Month", title = "Monthly Maximum Daily Withdrawal i
mgd_graph
```



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 data scraped.

```
#6.
water_scrape <- function(pwsid, Year){

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

  water_sys_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_sys_name <- website %>% html_nodes(water_sys_name_tag) %>% html_text()
  pwsid <- website %>% html_nodes(pwsid_tag) %>%  html_text()
  ownership <- website %>% html_nodes(ownership_tag) %>% html_text()
  mgd <- website %>% html_nodes(mgd_tag) %>% html_text() %>% as.numeric()

  month <- c("Jan","May","Sep","Feb","Jun","Oct","Mar","Jul","Nov","Apr","Aug","Dec")

  day <- 1

  water <- data.frame(water_sys_name, pwsid, ownership,
                          mgd, month, Year, day)
  water$date <- as.Date(with(water, paste(Year, month, day, sep = "-")),"%Y-%b-%d")

  water <- water %>%
  select(date, month, water_sys_name, pwsid, ownership, mgd)

  #Return the dataframe
  return(water)
}

# testing the function to see if it can replicate what we did before
water_scrape("03-32-010", "2019")
```

```
##          date month water_sys_name     pwsid    ownership   mgd
## 1  2019-01-01   Jan         Durham 03-32-010 Municipality 29.62
## 2  2019-05-01   May         Durham 03-32-010 Municipality 35.73
## 3  2019-09-01   Sep         Durham 03-32-010 Municipality 54.07
## 4  2019-02-01   Feb         Durham 03-32-010 Municipality 32.39
## 5  2019-06-01   Jun         Durham 03-32-010 Municipality 37.86
## 6  2019-10-01   Oct         Durham 03-32-010 Municipality 44.35
## 7  2019-03-01   Mar         Durham 03-32-010 Municipality 36.43
## 8  2019-07-01   Jul         Durham 03-32-010 Municipality 46.02
## 9  2019-11-01   Nov         Durham 03-32-010 Municipality 36.06
## 10 2019-04-01   Apr         Durham 03-32-010 Municipality 32.60
## 11 2019-08-01   Aug         Durham 03-32-010 Municipality 42.05
## 12 2019-12-01   Dec         Durham 03-32-010 Municipality 31.20
```

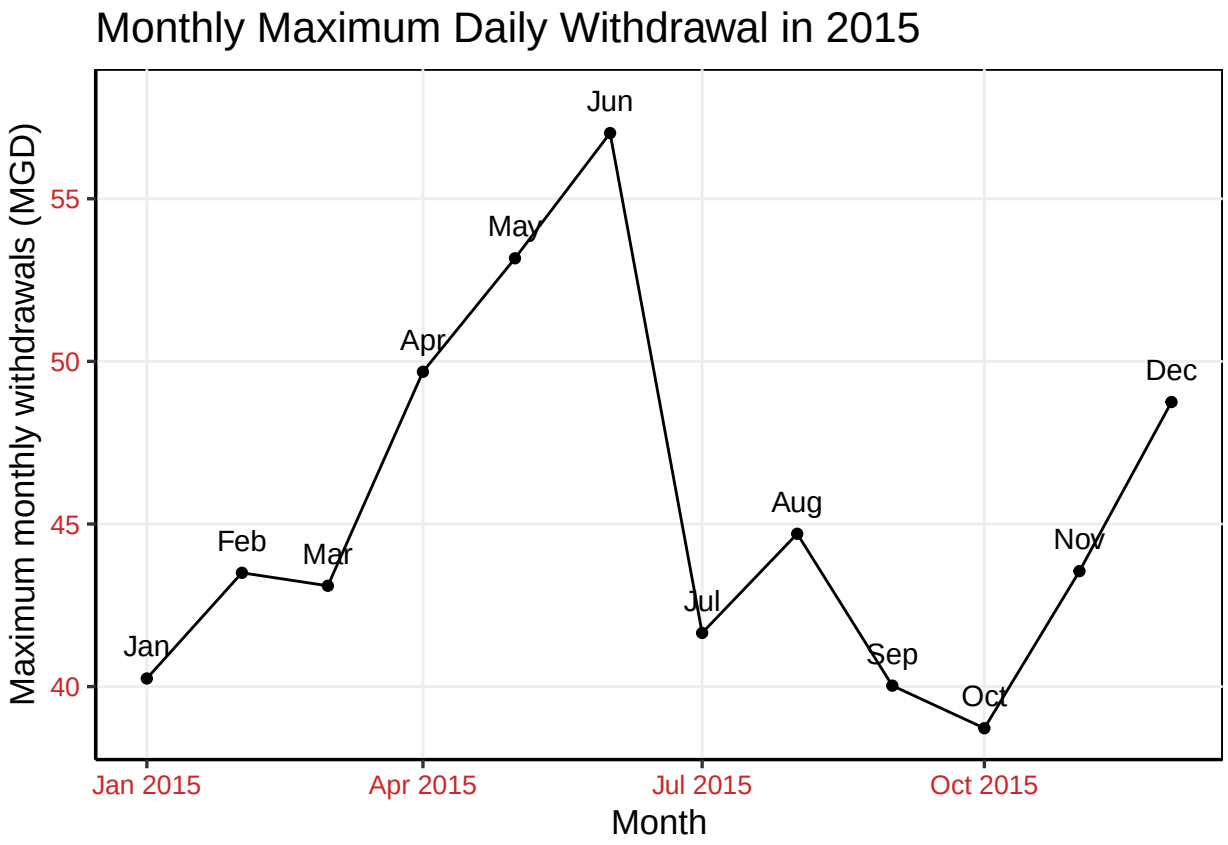
```
durham_water
```

```
##          date month water_sys_name     pswid    ownership   mgd
## 1  2019-01-01   Jan         Durham 03-32-010 Municipality 29.62
## 2  2019-05-01   May         Durham 03-32-010 Municipality 35.73
## 3  2019-09-01   Sep         Durham 03-32-010 Municipality 54.07
## 4  2019-02-01   Feb         Durham 03-32-010 Municipality 32.39
## 5  2019-06-01   Jun         Durham 03-32-010 Municipality 37.86
## 6  2019-10-01   Oct         Durham 03-32-010 Municipality 44.35
## 7  2019-03-01   Mar         Durham 03-32-010 Municipality 36.43
## 8  2019-07-01   Jul         Durham 03-32-010 Municipality 46.02
## 9  2019-11-01   Nov         Durham 03-32-010 Municipality 36.06
## 10 2019-04-01   Apr         Durham 03-32-010 Municipality 32.60
```

```
## 11 2019-08-01    Aug          Durham 03-32-010 Municipality 42.05
## 12 2019-12-01    Dec          Durham 03-32-010 Municipality 31.20
```

7. Use the function above to extract and plot max daily withdrawals for Durham for each month in 2015

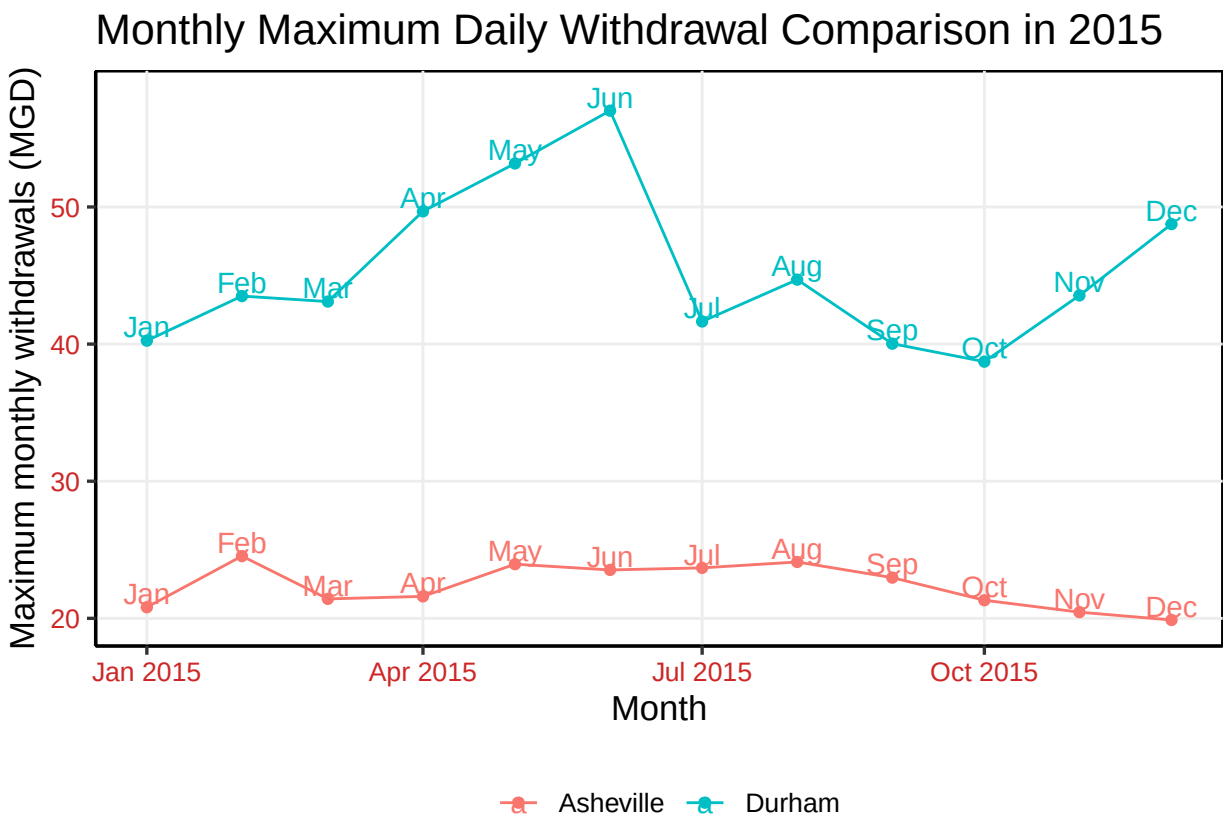
```
#7
durham_2015 <- water_scrape("03-32-010", "2015")

mgd2015_graph <- ggplot(data = durham_2015, aes(x = date, y = mgd)) +
  geom_line() +
  geom_point() +
  geom_text(aes(label = month), position = position_nudge(y = 1)) +
  labs(y = "Maximum monthly withdrawals (MGD)", x = "Month", title = "Monthly Maximum Daily Withdrawal
mgd2015_graph
```



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_2015 <- water_scrape("01-11-010", "2015")

water_2015 <- rbind(durham_2015, asheville_2015)

water2015_graph <- ggplot(data = water_2015, aes(x = date, y = mgd, color = water_sys_name)) +
  geom_line() +
  geom_point() +
  geom_text(aes(label = month), position = position_nudge(y = 1)) +
```

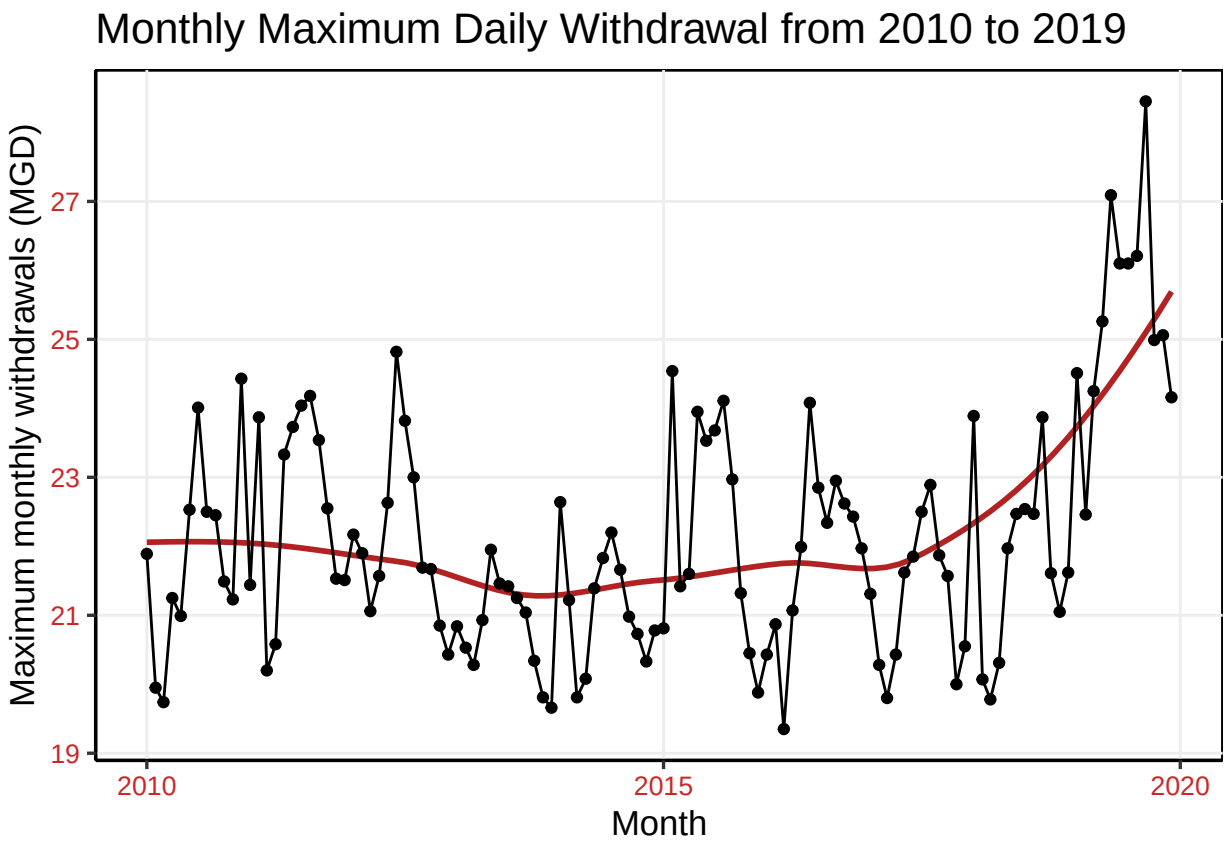
```
  labs(y = "Maximum monthly withdrawals (MGD)", x = "Month", title = "Monthly Maximum Daily Withdrawal C
  theme(legend.title = element_blank(), legend.position = "bottom")
water2015_graph
```



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

# could do it this way, but below I do it a slightly different way
# asheville_2010 <- water_scrape("01-11-010", "2010")
# asheville_2011 <- water_scrape("01-11-010", "2011")
# asheville_2012 <- water_scrape("01-11-010", "2012")
# asheville_2013 <- water_scrape("01-11-010", "2013")
# asheville_2014 <- water_scrape("01-11-010", "2014")
# asheville_2015 <- water_scrape("01-11-010", "2015")
# asheville_2016 <- water_scrape("01-11-010", "2016")
# asheville_2017 <- water_scrape("01-11-010", "2017")
# asheville_2018 <- water_scrape("01-11-010", "2018")
# asheville_2019 <- water_scrape("01-11-010", "2019")

# making the data frame
# using the functions inside the data frame to prevent creation of 10 other frames
asheville_water <- rbind(water_scrape("01-11-010", "2010"), water_scrape("01-11-010", "2011"),
                         water_scrape("01-11-010", "2012"), water_scrape("01-11-010", "2013"),
                         water_scrape("01-11-010", "2014"), water_scrape("01-11-010", "2015"),
```

```
                        water_scrape("01-11-010", "2016"), water_scrape("01-11-010", "2017"),
                        water_scrape("01-11-010", "2018"), water_scrape("01-11-010", "2019"))

asheville_graph <- ggplot(data = asheville_water, aes(x = date, y = mgd)) +
  geom_smooth(formula = y ~ x, method="loess",se=FALSE,color = "firebrick")+
  geom_line() +
  geom_point() +
  labs(y = "Maximum monthly withdrawals (MGD)", x = "Month", title = "Monthly Maximum Daily Withdrawal
  theme(legend.title = element_blank(), legend.position = "bottom")
asheville_graph
```



Question: Just by looking at the plot (i.e. not running statistics), does Asheville have a trend in water usage over time?

Answer: Yes, if one just looks at the plot it appears that recently there has been an increase in water usage since around 2017 in Asheville. Before that the water usage appears to have remained relatively constant.