Assignment 09: Data Scraping

Clara Fast

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()
require(tidyverse)
require(rvest)
require(lubridate)

#Set theme
mytheme <- theme_classic(base_size = 14) +
  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 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
water_webpage<- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')
water_webpage
```

```
## {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= ...
```

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
#Scrape data for water system name
water.system.name <- water_webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water.system.name
```

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

```
#Scrape data for PWSID
pwsid <- water_webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
pwsid
```

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

```r
#Scrape data for ownership
ownership <- water_webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
ownership
```

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

```r
#Scrape data for max daily use
max.withdrawals.mgd <- water_webpage %>%
  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.)

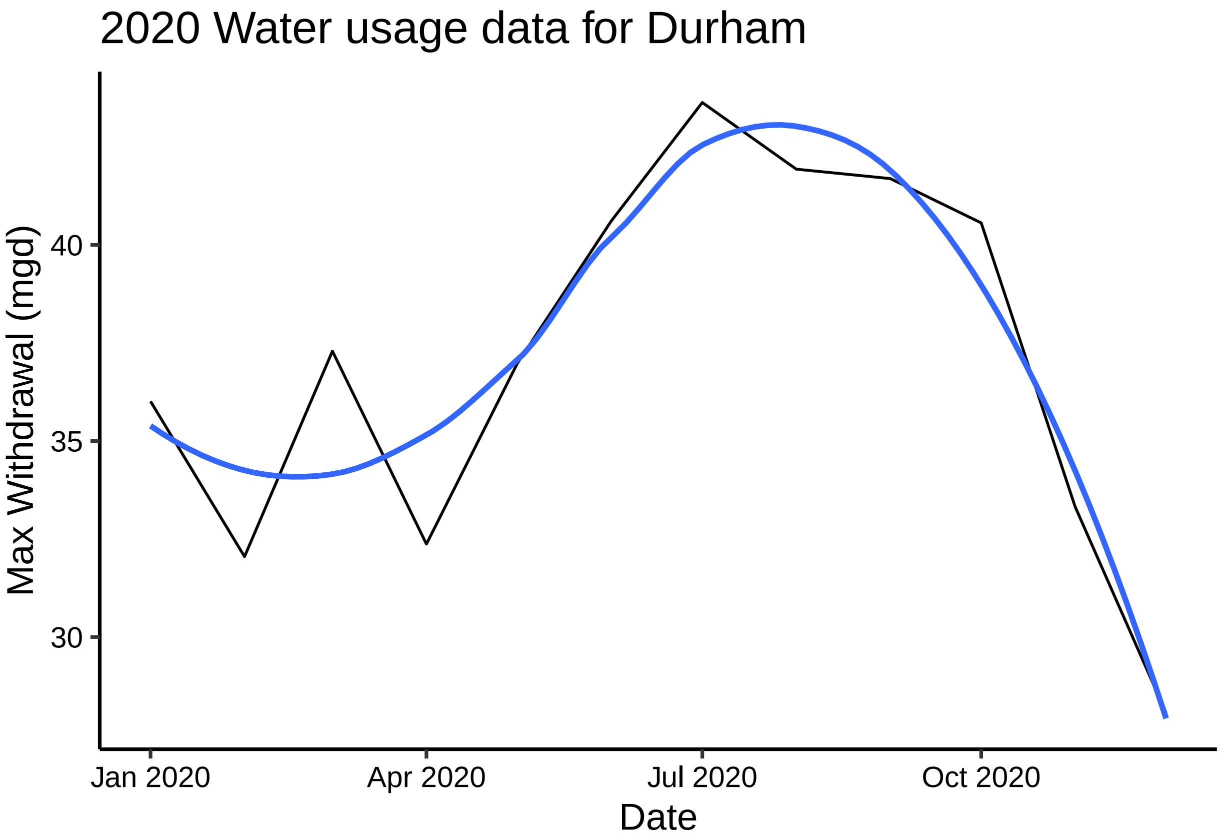
   TIP: Use `rep()` to repeat a value when creating a dataframe.

   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

```r
#4

#Create new month dataframe
month<- c(1,5,9,2,6,10,3,7,11,4,8,12)

#Create Dataframe
df_withdrawals <- data.frame("Month" = as.numeric(month),
                             "Year" = rep(2020,12),
                             "Ownership"= as.character(ownership),
                             "Water System Name" = as.character(water.system.name),
                             "PWSID" = as.character(pwsid),
                             "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd))

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

#5
#Plot max daily withdrawals across the months for 2020
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 Durham"),
       y="Max 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**.

```
#6.
#Create scraping function
the_year<-2020

scrape.it <- function(the_year, the_pwsid){

   #Retrieve the website contents
  the_website<-read_html(paste0
                        ('https://www.ncwater.org/WUDC/app/LWSP/report.php?','pwsid=',the_pwsid,'&year=

  #Scrape the data items
  water.system.name <- the_website %>%
    html_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>% html_text()
  pwsid <- the_website %>%
    html_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html_text()
  max.withdrawals.mgd <- the_website %>%
    html_nodes('th~ td+ td') %>% html_text()
  ownership <- the_website %>%
    html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>% html_text()

  #Convert to a dataframe
  df_withdrawals_new <- data.frame("Month" = as.numeric(month),
                               "Year" = rep(the_year,12),
                               "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd),
                              "Ownership"= as.character(ownership),
                             "Water System Name" = as.character(water.system.name),
                             "PWSID" = as.character(pwsid)) %>%
    mutate(Date = my(paste(Month,"-",Year)))

  Sys.sleep(1)

  return(df_withdrawals_new)

}
```

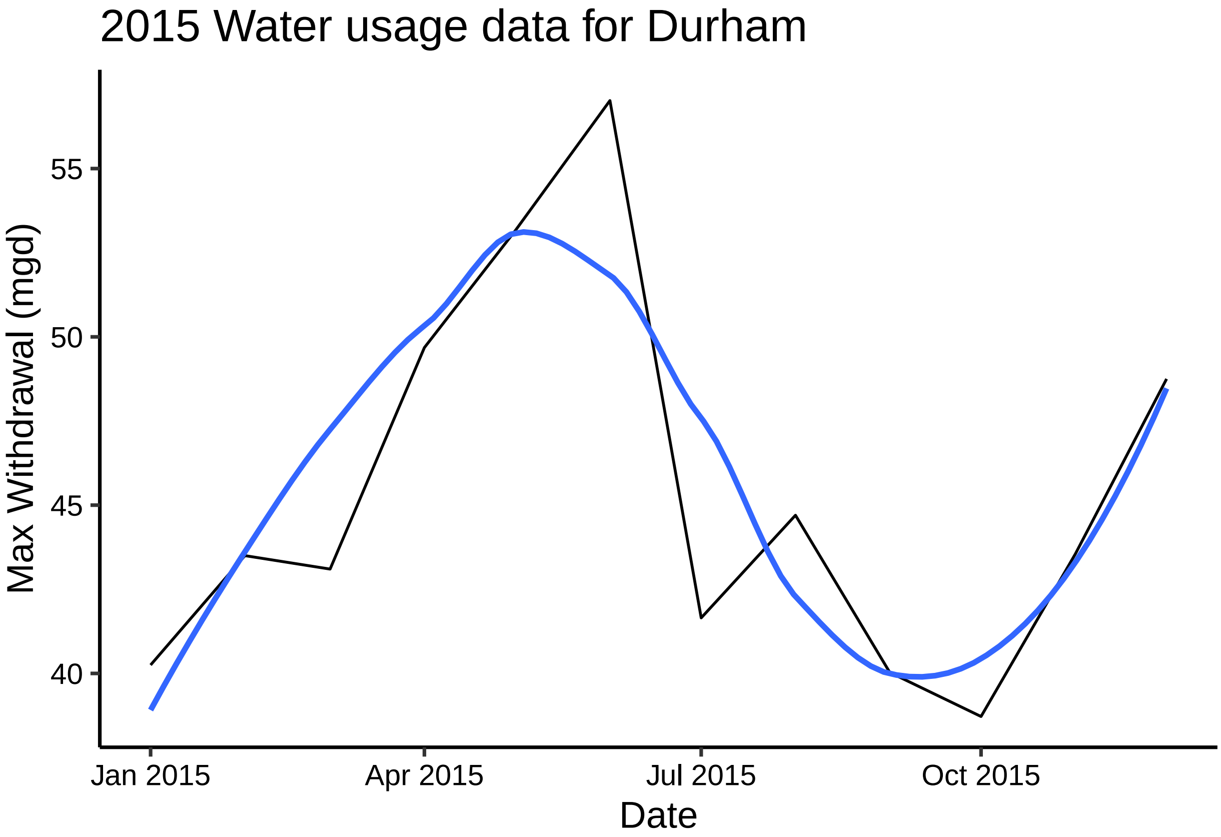
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID='03-32-010') for each month in 2015

```
#7
#Extract data for Durham, 2015
the_year<-2015
the_pwsid<-as.character('03-32-010')
#Assign to dataframe
the_df<-data.frame(scrape.it(the_year,the_pwsid))

print(the_df)
```

```
##    Month Year Max_Withdrawals_mgd    Ownership Water.System.Name     PWSID
## 1      1 2015               40.25 Municipality            Durham 03-32-010
## 2      5 2015               53.17 Municipality            Durham 03-32-010
## 3      9 2015               40.03 Municipality            Durham 03-32-010
## 4      2 2015               43.50 Municipality            Durham 03-32-010
## 5      6 2015               57.02 Municipality            Durham 03-32-010
## 6     10 2015               38.72 Municipality            Durham 03-32-010
## 7      3 2015               43.10 Municipality            Durham 03-32-010
## 8      7 2015               41.65 Municipality            Durham 03-32-010
## 9     11 2015               43.55 Municipality            Durham 03-32-010
## 10     4 2015               49.68 Municipality            Durham 03-32-010
## 11     8 2015               44.70 Municipality            Durham 03-32-010
## 12    12 2015               48.75 Municipality            Durham 03-32-010
##          Date
## 1  2015-01-01
## 2  2015-05-01
## 3  2015-09-01
## 4  2015-02-01
## 5  2015-06-01
## 6  2015-10-01
## 7  2015-03-01
## 8  2015-07-01
## 9  2015-11-01
## 10 2015-04-01
## 11 2015-08-01
## 12 2015-12-01
```

```
#Plot max daily withdrawals for Durham, 2015
ggplot(the_df,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2015 Water usage data for Durham"),
       y="Max 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.

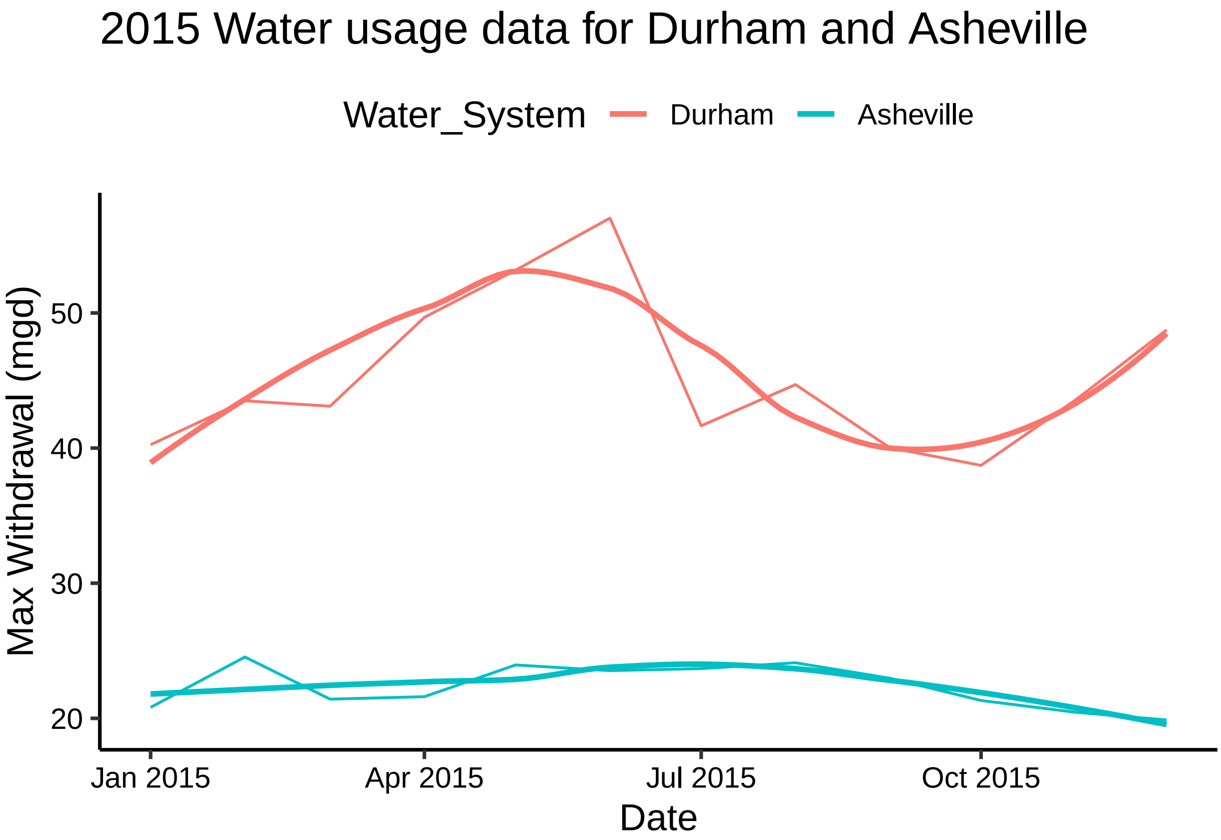
```
#8
#Extract data for Asheville, 2015
the_year<-2015
the_pwsid<-as.character('01-11-010')
#Assign to dataframe
the_df_asheville<-data.frame(scrape.it(the_year,the_pwsid))
```

```
print(the_df_asheville)
```

```
##    Month Year Max_Withdrawals_mgd    Ownership Water.System.Name     PWSID
## 1      1 2015               20.81 Municipality         Asheville 01-11-010
## 2      5 2015               23.95 Municipality         Asheville 01-11-010
## 3      9 2015               22.97 Municipality         Asheville 01-11-010
## 4      2 2015               24.54 Municipality         Asheville 01-11-010
## 5      6 2015               23.53 Municipality         Asheville 01-11-010
## 6     10 2015               21.32 Municipality         Asheville 01-11-010
## 7      3 2015               21.42 Municipality         Asheville 01-11-010
## 8      7 2015               23.68 Municipality         Asheville 01-11-010
## 9     11 2015               20.45 Municipality         Asheville 01-11-010
## 10     4 2015               21.60 Municipality         Asheville 01-11-010
## 11     8 2015               24.11 Municipality         Asheville 01-11-010
## 12    12 2015               19.88 Municipality         Asheville 01-11-010
##          Date
## 1  2015-01-01
## 2  2015-05-01
## 3  2015-09-01
## 4  2015-02-01
## 5  2015-06-01
## 6  2015-10-01
## 7  2015-03-01
## 8  2015-07-01
## 9  2015-11-01
## 10 2015-04-01
## 11 2015-08-01
## 12 2015-12-01
```

```
#Create plot comparing Asheville to Durham's water withdrawals
 #Join dataframes
water_join<-merge(x = the_df,
                        y = the_df_asheville,
                        by = c("Date", "Month", "Year"))

#Plot
ggplot_water<-water_join%>%
  gather(Water_System, City, Max_Withdrawals_mgd.x, Max_Withdrawals_mgd.y) %>%
  ggplot(aes(x=Date, y=City, colour=Water_System)) +
  geom_line()+
  geom_smooth(method="loess",se=FALSE) +
    scale_shape_discrete(labels = c("Durham", "Asheville")) +
  scale_colour_discrete(labels = c("Durham", "Asheville")) +
  labs(title = paste("2015 Water usage data for Durham and Asheville"),
       y="Max Withdrawal (mgd)",
       x="Date")

print(ggplot_water)
```

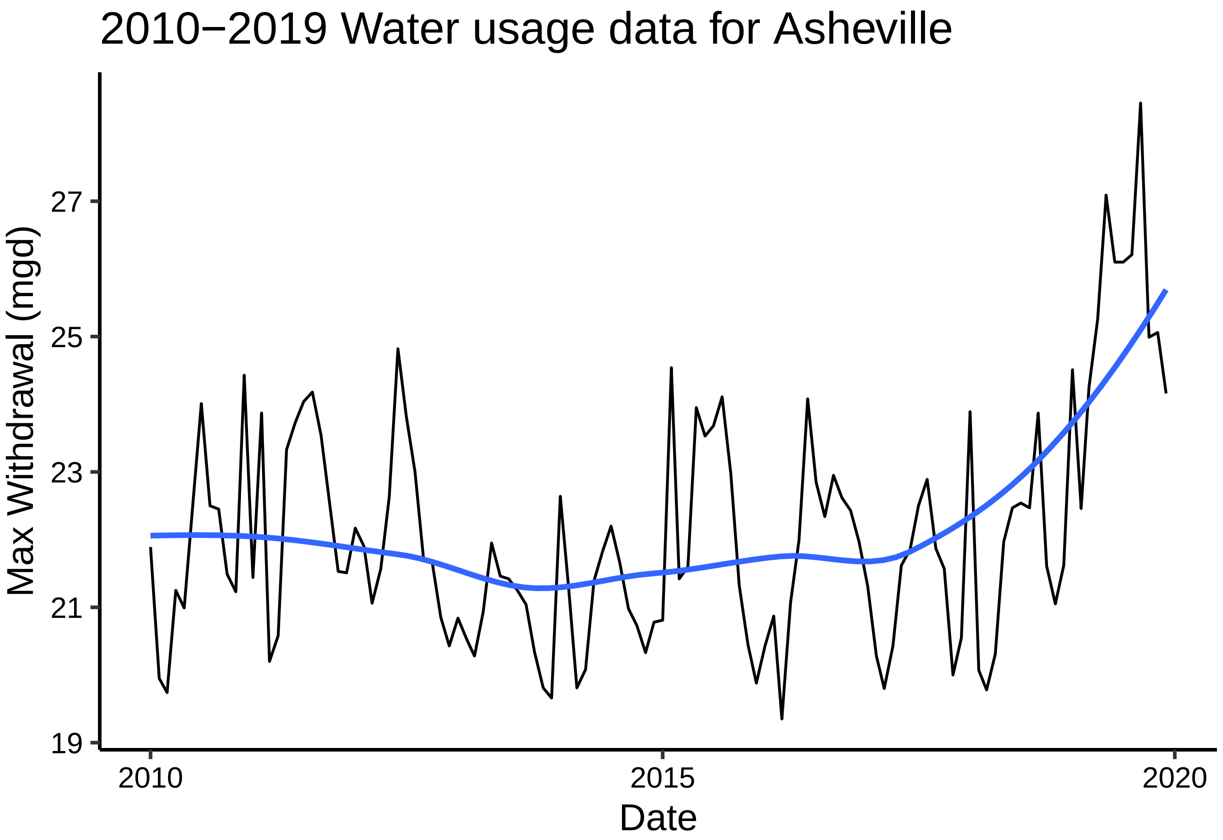
```
## `geom_smooth()` using formula 'y ~ x'
```



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_year<-seq(2010,2019)
the_pwsid<-as.character('01-11-010')

#Apply scrape function
asheville_df_new <- lapply(X = the_year,
                  FUN = scrape.it,
                  the_pwsid=the_pwsid)

#Conflate into single dataframe
asheville_df_final <- bind_rows(asheville_df_new)

#Plot Asheville's max daily withdrawal from 2010-2019
ggplot(asheville_df_final,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2010-2019 Water usage data for Asheville"),
       y="Max 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, there appears to be an upward trend in water usage over time. The plot shows there was a dramatic incline in the most recent years.