IreneChang_A10_DataScraping

Irene Chang

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)

getwd()
```

```
## [1] "/home/guest/EDA/EDA-Spring2023"
```

```
#set theme

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 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 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')
```

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:
- 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), with the first value being "27.6400".

```
#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"
```

```
max.withdrawals.mgd <-
  Durham.LWSP %>%
  html_nodes("th~ td+ td") %>%
  html_text()
max.withdrawals.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"
```

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.)

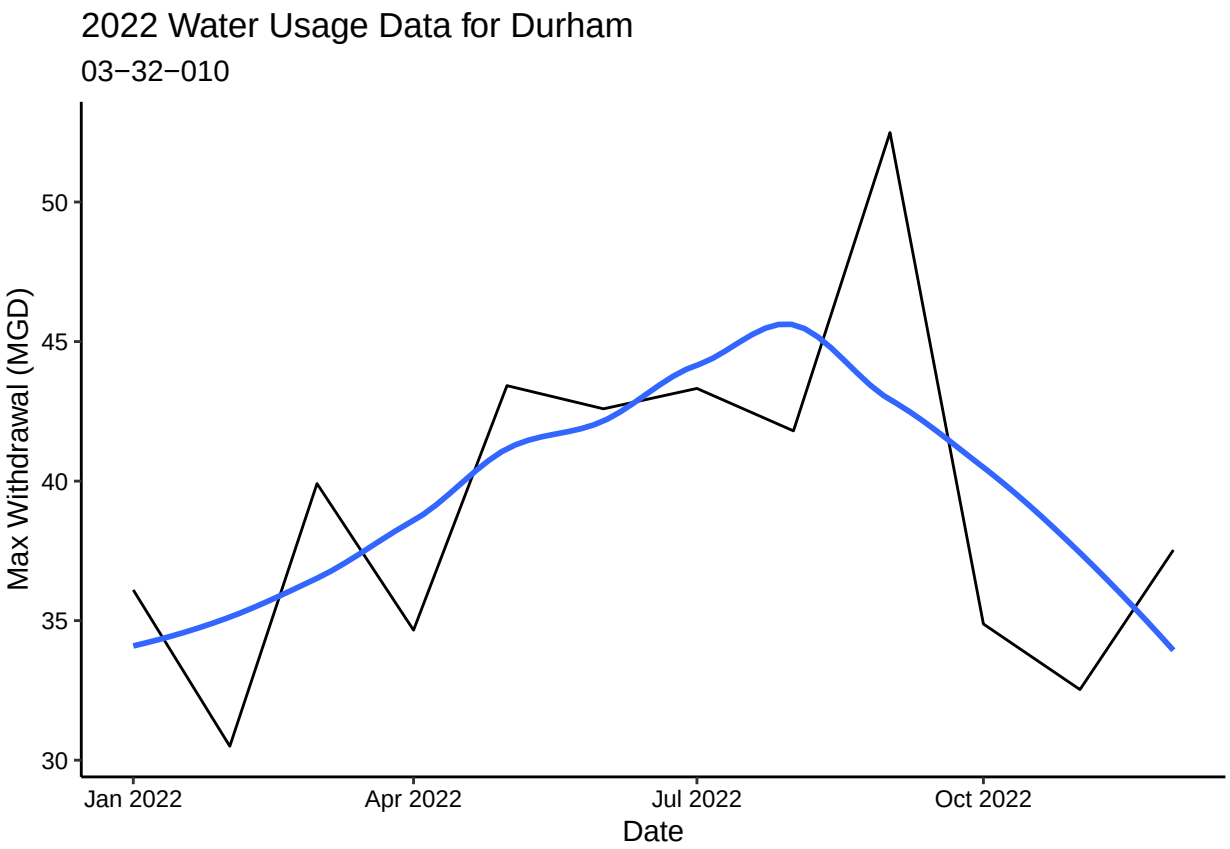
   TIP: Use `rep()` to repeat a value when creating a dataframe.

   NOTE: It's likely you won't be able to scrape the monthly widthrawal data in 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...

5. Create a line plot of the average daily withdrawals across the months for 2022

```
#4

df.withdrawals <-
  data.frame("Month" = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12),
             "Year" = rep(2022,12),
             "Max_Withdrawal_MGD" = as.numeric(max.withdrawals.mgd)) %>%
  mutate("Water_System_Name" = !!water.system.name,
         "PWSID" = !!PWSID,
         "Ownership" = !!ownership,
         "Date" = my(paste(Month, "-", Year)))

df.withdrawals <- arrange(df.withdrawals, Month)

#5

ggplot(df.withdrawals,
       aes(x=Date,
           y= Max_Withdrawal_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(
    title = paste("2022 Water Usage Data for", water.system.name),
    subtitle = PWSID,
    x= "Date",
    y= "Max Withdrawal (MGD)")
```

```
## `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 (pwsid) scraped**.

```
#6.

the.baseurl <-'https://www.ncwater.org/WUDC/app/LWSP/report.php?'
the.pwsid <- '03-32-010'
the.year <- '2022'
the.scrape.url <- paste0(the.baseurl, "pwsid=", the.pwsid, '&year=', the.year)

print(the.scrape.url)
```

```
## [1] "https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022"
```

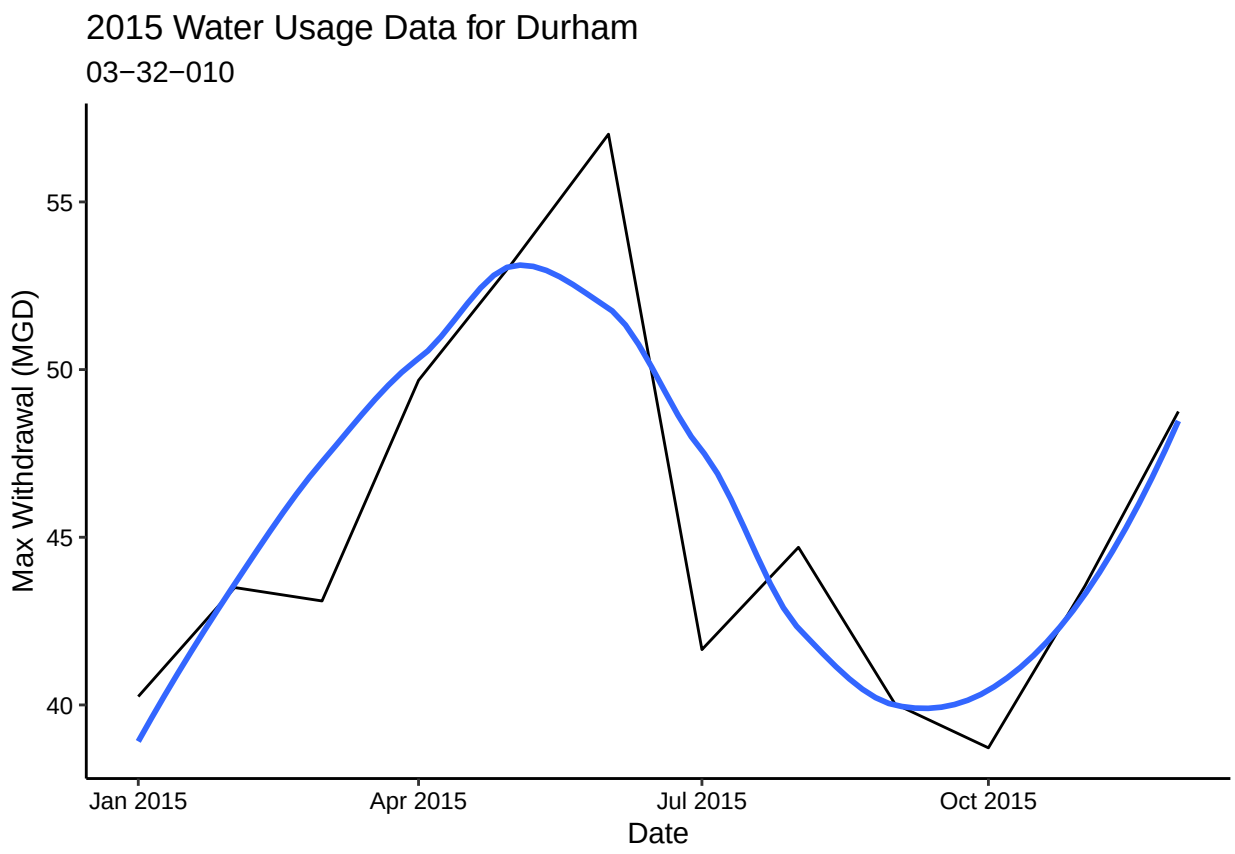
```
LWSP.website <- read_html(the.scrape.url)

scrape.it <- function(the.year, the.pwsid){

  LWSP.website <- read_html(paste0(the.baseurl, "pwsid=", the.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)'
  max.withdrawals.mgd.tag <- 'th~ td+ td'
```

```
  water.system.name <-
    LWSP.website %>%
    html_nodes(water.system.name.tag) %>%
    html_text()

  PWSID <-
    LWSP.website %>%
    html_nodes(PWSID.tag) %>%
    html_text()

  ownership <-
    LWSP.website %>%
    html_nodes(ownership.tag) %>%
    html_text()

  max.withdrawals.mgd <-
    LWSP.website %>%
    html_nodes(max.withdrawals.mgd.tag) %>%
    html_text()

  df.withdrawals <-
    data.frame("Month" = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12),
               "Year" = rep(the.year,12),
               "Max_Withdrawal_MGD" = as.numeric(max.withdrawals.mgd)) %>%
    mutate("Water_System_Name" = !!water.system.name,
           "PWSID" = !!PWSID,
           "Ownership" = !!ownership,
           "Date" = my(paste(Month, "-", Year)))

  df.withdrawals <- arrange(df.withdrawals, Month)

  return(df.withdrawals)
  }
```

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

```
#7

durham2015.df <- scrape.it(2015, '03-32-010')


ggplot(durham2015.df,
       aes(x=Date,
           y= Max_Withdrawal_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(
    title = paste(durham2015.df$Year, "Water Usage Data for", durham2015.df$Water_System_Name),
    subtitle = PWSID,
    x= "Date",
    y= "Max Withdrawal (MGD)")
```

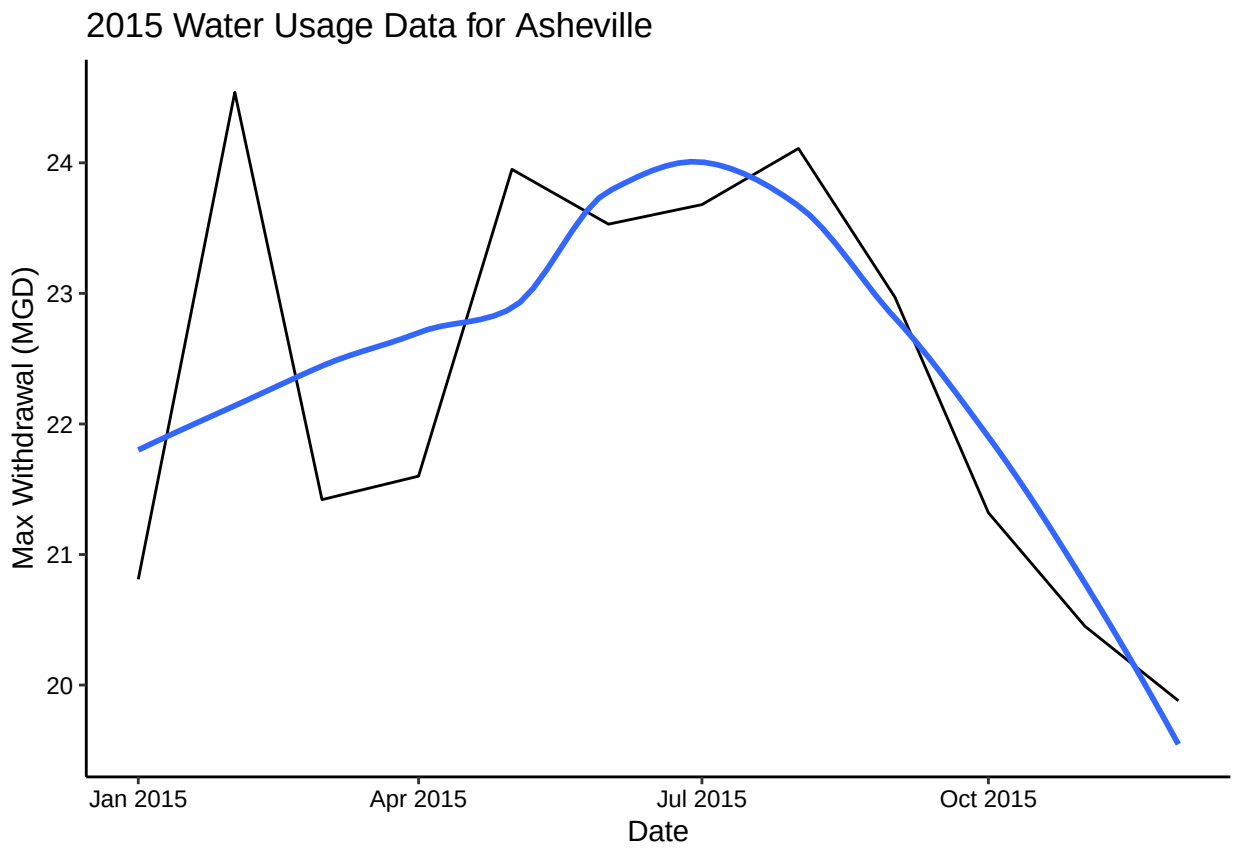
```
## `geom_smooth()` using formula = 'y ~ x'
```



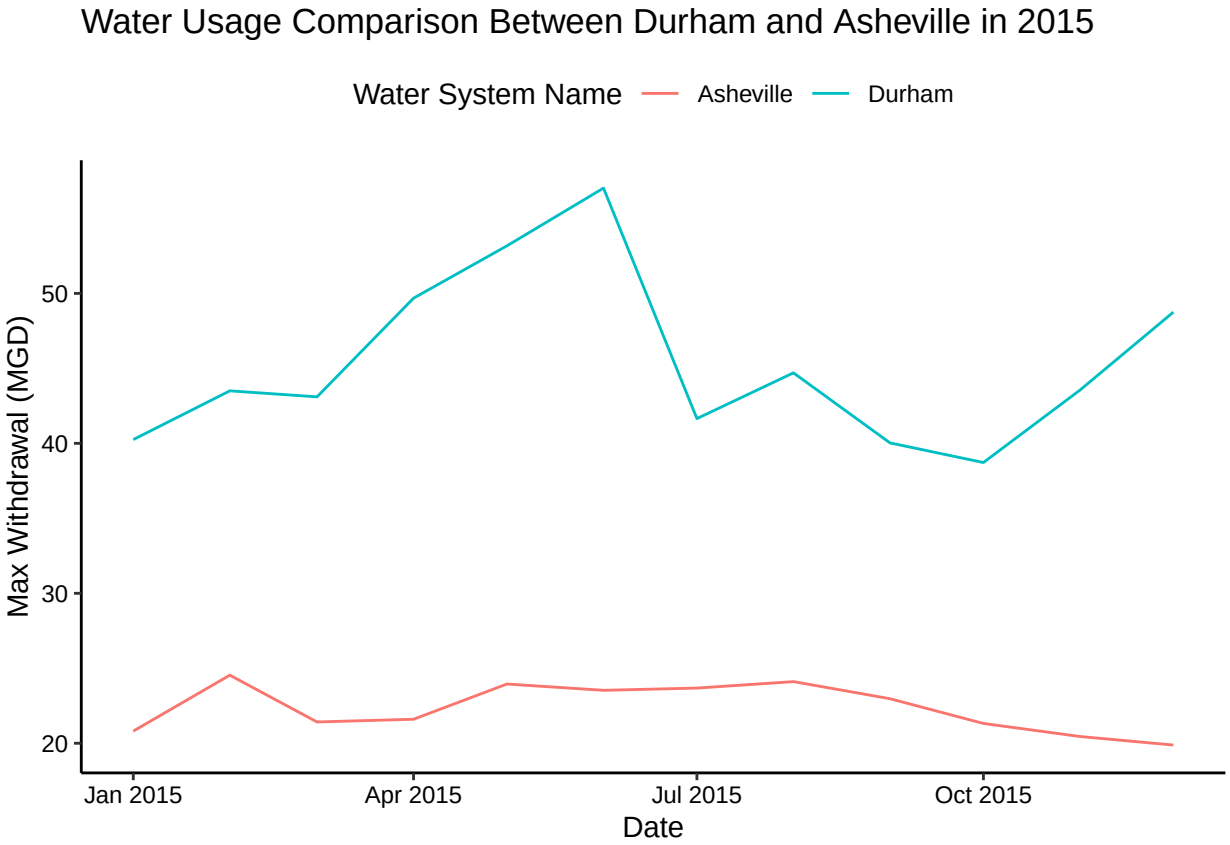
8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8

asheville2015.df <- scrape.it(2015, '01-11-010')


ggplot(asheville2015.df,
       aes(x=Date,
           y= Max_Withdrawal_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(
    title = paste(asheville2015.df$Year, "Water Usage Data for", asheville2015.df$Water_System_Name),
    x= "Date",
    y= "Max Withdrawal (MGD)")
```

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




```
durham.asheville.2015.df <-
  rbind(durham2015.df, asheville2015.df)

ggplot(durham.asheville.2015.df,
       aes(x= Date,
           y= Max_Withdrawal_MGD,
           color = Water_System_Name)) +
         geom_line() +
         labs(
           title = "Water Usage Comparison Between Durham and Asheville in 2015",
           x = "Date",
           y = "Max Withdrawal (MGD)",
           color = "Water System Name")
```



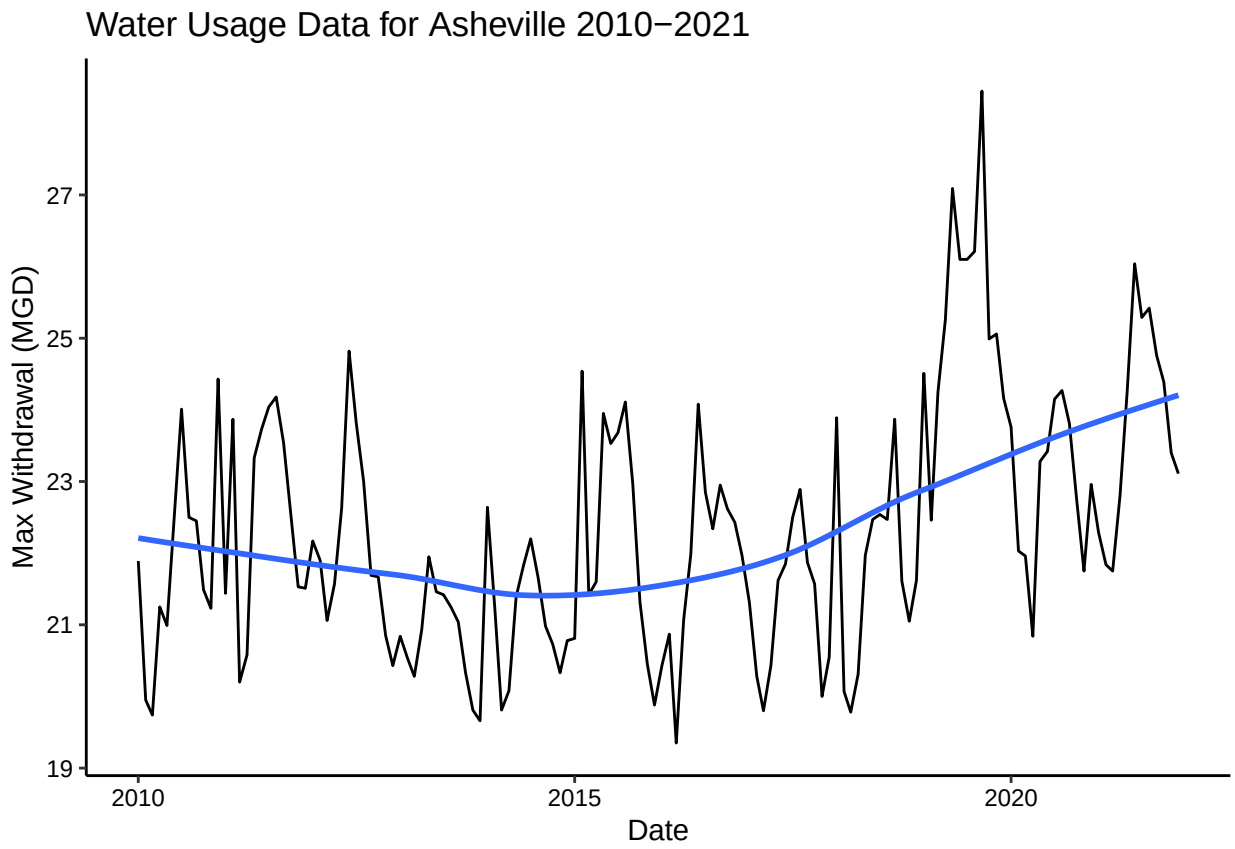
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   TIP: See Section 3.2 in the "09_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

the.years <- rep(2010:2021)
new.PSWID <- '01-11-010'

asheville.dfs <-
  map2(the.years, new.PSWID, scrape.it)

asheville.df <-
  bind_rows(asheville.dfs)

ggplot(asheville.df,
       aes(x=Date,
           y= Max_Withdrawal_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(
    title = "Water Usage Data for Asheville 2010-2021",
```

```
    x= "Date",
    y= "Max Withdrawal (MGD)")
```

```
## `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: In general, it looks like over time, the maximum withdrawl (MGD) has steadily increased starting from around 2015. From 2010-2015, maximum withdrawal was on a slight decline.