

Assignment 09: Data Scraping

GuruBandaa Khalsa

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on data scraping.

Directions

1. Rename this file `<FirstLast>_A09_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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

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. I checked my working directory using the getwd() function, loaded the  
# the packages `tidyverse` and `rvest`, set my ggplot theme, and set knitting  
# knitting settings.  
getwd()
```

```
## [1] "/Users/survivormangb/Desktop/Masters at Duke/Second Year/Fall Semester/Environmental Data Analy
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.2
```

```
## Warning: package 'ggplot2' was built under R version 4.1.2
```

```
## Warning: package 'tibble' was built under R version 4.1.2
```

```
## Warning: package 'tidyr' was built under R version 4.1.2
```

```
## Warning: package 'readr' was built under R version 4.1.2
```

```
## Warning: package 'dplyr' was built under R version 4.1.2

## Warning: package 'stringr' was built under R version 4.1.2

## Warning: package 'forcats' was built under R version 4.1.2
```

```
library(lubridate)
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.1.2
```

```
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=80), tidy=TRUE)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2021 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=2021>

Indicate this website as the as the URL to be scraped. (In other words, read the contents into an `rvest` webpage object.)

```
# 2. I indicated the website as the URL to be scraped by reading the contents
# into an `rvest` webpage object.
webpage <- read_html("https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2021")
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:
 - Maximum Daily 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. I scraped the desired values and assigned them to four separate variables.
```

```
water.system.name <- webpage %>%  
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%  
  html_text()  
  
pwsid <- webpage %>%  
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%  
  html_text()  
  
ownership <- webpage %>%  
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%  
  html_text()  
  
max.withdrawals.mgd <- webpage %>%  
  html_nodes("th~ td+ td") %>%  
  html_text()
```

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 withdrawal 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...

5. Create a line plot of the maximum daily withdrawals across the months for 2021

```
# 4. I converted my scraped data into a dataframe and manually made a column  
# for the months associated with the MGD.
```

```
Month <- c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12)  
Year <- c(2021)
```

```
the_df <- data.frame(WaterSystem = water.system.name, PWSID = pwsid, Ownership = ownership,  
  `Max Day Use` = as.numeric(max.withdrawals.mgd)) %>%  
  mutate(Month = Month, Year = Year, Date = my(paste0(Month, "-", Year)), PWSID = !!pwsid,  
    Ownership = !!ownership)
```

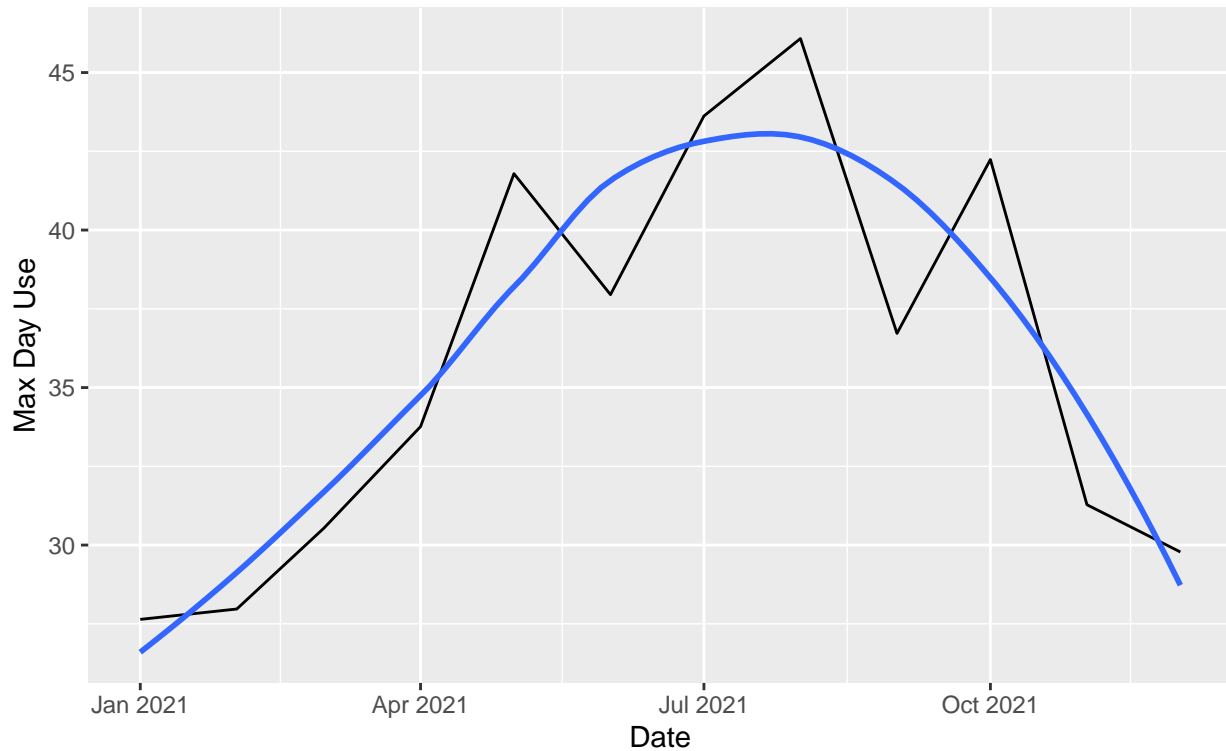
```
# 5. I created a line plot of the maximum daily withdrawals across the months  
# for 2021.
```

```
ggplot(the_df, aes(x = Date, y = Max.Day.Use)) + geom_line(aes(group = 1)) + geom_smooth(method = "loess",  
  se = FALSE) + labs(title = paste("2021 Durham Monthly Maximum Daily Withdrawals"),  
  subtitle = "GuruBandaa Khalsa", y = "Max Day Use", x = "Date")
```

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

2021 Durham Monthly Maximum Daily Withdrawals

GuruBandaa Khalsa



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. I constructed a function using my code from above that can scrape data for
# any PWSID and year for which the NC DEQ has data.
base_url <- "https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid="
pwsid <- "03-32-010"
Year <- 2015
scrape_url <- paste0(base_url, pwsid, "&year=", Year)
website <- read_html(scrape_url)

scrape.it <- function(pwsid, Year) {

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

  water.system.name <- website %>%
    html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
    html_text()
  pwsid <- website %>%
    html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
    html_text()
  ownership <- website %>%
    html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
```

```

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

    the_df2 <- data.frame(WaterSystem = water.system.name, PWSID = pwsid, Ownership = ownership,
      `Max Day Use` = as.numeric(max.withdrawals.mgd)) %>%
      mutate(Month = Month, Year = Year, Date = my(paste0(Month, "-", Year)), PWSID = !!pwsid,
        Ownership = !!ownership)

    return(the_df2)
}

```

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

```

# 7. I used the above function to extract and plot max daily withdrawals for
# Durham (PWSID='03-32-010') for each month in 2015.
df <- scrape.it("03-32-010", 2015)
print(df)

```

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

```

ggplot(df, aes(x = Date, y = Max.Day.Use)) + geom_line() + geom_smooth(method = "loess",
  se = FALSE) + labs(title = paste("2015 Durham Monthly Maximum Daily Withdrawals"),
  subtitle = "GuruBandaa Khalsa", y = "Max Day Use", x = "Date")

```

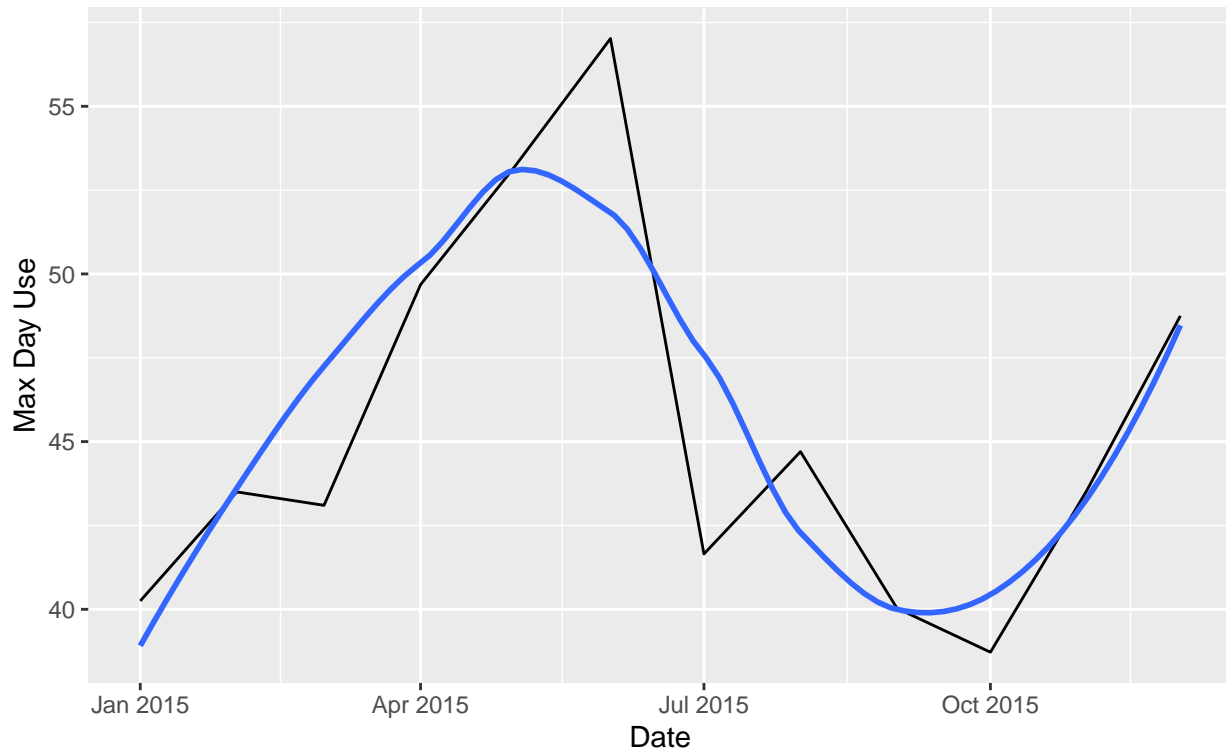
```

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

```

2015 Durham Monthly Maximum Daily Withdrawals

GuruBandaa Khalsa



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. I used the function above to extract data for Asheville (PWSID =
# 01-11-010) in 2015.
base_url <- "https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid="
pwsid2 <- "01-11-010"
Year <- 2015
scrape_url <- paste0(base_url, pwsid, "&year=", Year)
website <- read_html(scrape_url)

ashevillescrape <- scrape.it(pwsid2, 2015)
print(ashevillescrape)
```

| ## | WaterSystem | PWSID | Ownership | Max.Day.Use | Month | Year | Date |
|------|-------------|-----------|--------------|-------------|-------|------|------------|
| ## 1 | Asheville | 01-11-010 | Municipality | 20.81 | 1 | 2015 | 2015-01-01 |
| ## 2 | Asheville | 01-11-010 | Municipality | 23.95 | 5 | 2015 | 2015-05-01 |
| ## 3 | Asheville | 01-11-010 | Municipality | 22.97 | 9 | 2015 | 2015-09-01 |
| ## 4 | Asheville | 01-11-010 | Municipality | 24.54 | 2 | 2015 | 2015-02-01 |
| ## 5 | Asheville | 01-11-010 | Municipality | 23.53 | 6 | 2015 | 2015-06-01 |
| ## 6 | Asheville | 01-11-010 | Municipality | 21.32 | 10 | 2015 | 2015-10-01 |
| ## 7 | Asheville | 01-11-010 | Municipality | 21.42 | 3 | 2015 | 2015-03-01 |
| ## 8 | Asheville | 01-11-010 | Municipality | 23.68 | 7 | 2015 | 2015-07-01 |
| ## 9 | Asheville | 01-11-010 | Municipality | 20.45 | 11 | 2015 | 2015-11-01 |

```
## 10 Asheville 01-11-010 Municipality 21.60 4 2015 2015-04-01
## 11 Asheville 01-11-010 Municipality 24.11 8 2015 2015-08-01
## 12 Asheville 01-11-010 Municipality 19.88 12 2015 2015-12-01
```

```
# Merge dataframes
```

```
durhamANDasheville <- rbind(df, ashevillescrape)
print(durhamANDasheville)
```

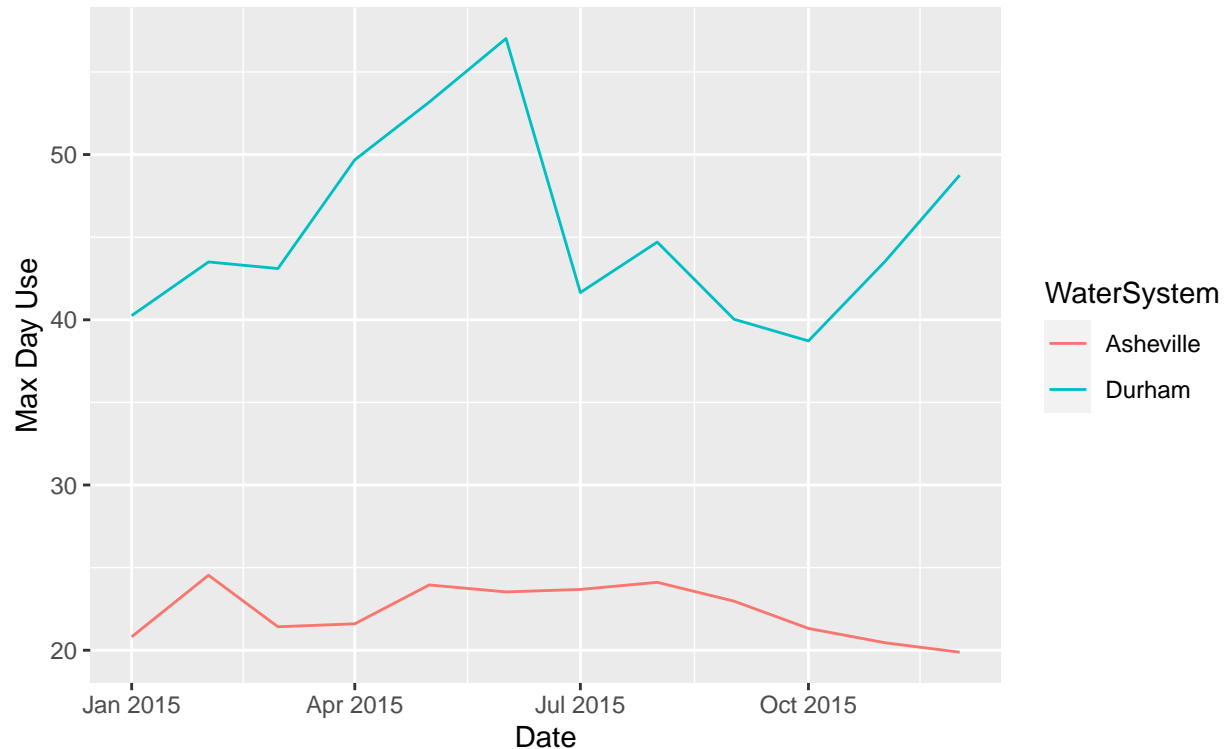
```
## WaterSystem PWSID Ownership Max.Day.Use Month Year Date
## 1 Durham 03-32-010 Municipality 40.25 1 2015 2015-01-01
## 2 Durham 03-32-010 Municipality 53.17 5 2015 2015-05-01
## 3 Durham 03-32-010 Municipality 40.03 9 2015 2015-09-01
## 4 Durham 03-32-010 Municipality 43.50 2 2015 2015-02-01
## 5 Durham 03-32-010 Municipality 57.02 6 2015 2015-06-01
## 6 Durham 03-32-010 Municipality 38.72 10 2015 2015-10-01
## 7 Durham 03-32-010 Municipality 43.10 3 2015 2015-03-01
## 8 Durham 03-32-010 Municipality 41.65 7 2015 2015-07-01
## 9 Durham 03-32-010 Municipality 43.55 11 2015 2015-11-01
## 10 Durham 03-32-010 Municipality 49.68 4 2015 2015-04-01
## 11 Durham 03-32-010 Municipality 44.70 8 2015 2015-08-01
## 12 Durham 03-32-010 Municipality 48.75 12 2015 2015-12-01
## 13 Asheville 01-11-010 Municipality 20.81 1 2015 2015-01-01
## 14 Asheville 01-11-010 Municipality 23.95 5 2015 2015-05-01
## 15 Asheville 01-11-010 Municipality 22.97 9 2015 2015-09-01
## 16 Asheville 01-11-010 Municipality 24.54 2 2015 2015-02-01
## 17 Asheville 01-11-010 Municipality 23.53 6 2015 2015-06-01
## 18 Asheville 01-11-010 Municipality 21.32 10 2015 2015-10-01
## 19 Asheville 01-11-010 Municipality 21.42 3 2015 2015-03-01
## 20 Asheville 01-11-010 Municipality 23.68 7 2015 2015-07-01
## 21 Asheville 01-11-010 Municipality 20.45 11 2015 2015-11-01
## 22 Asheville 01-11-010 Municipality 21.60 4 2015 2015-04-01
## 23 Asheville 01-11-010 Municipality 24.11 8 2015 2015-08-01
## 24 Asheville 01-11-010 Municipality 19.88 12 2015 2015-12-01
```

```
# Plot data
```

```
ggplot(durhamANDasheville) + geom_line(aes(x = Date, y = Max.Day.Use, color = WaterSystem)) +
  labs(title = paste("2015 Durham vs. Asheville Monthly Maximum Daily Withdrawals"),
        subtitle = "GuruBandaa Khalsa", y = "Max Day Use", x = "Date")
```

2015 Durham vs. Asheville Monthly Maximum Daily Withdrawals

GuruBandaa Khalsa



- 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.

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 `bind_rows()` to combine the dataframes into a single one.

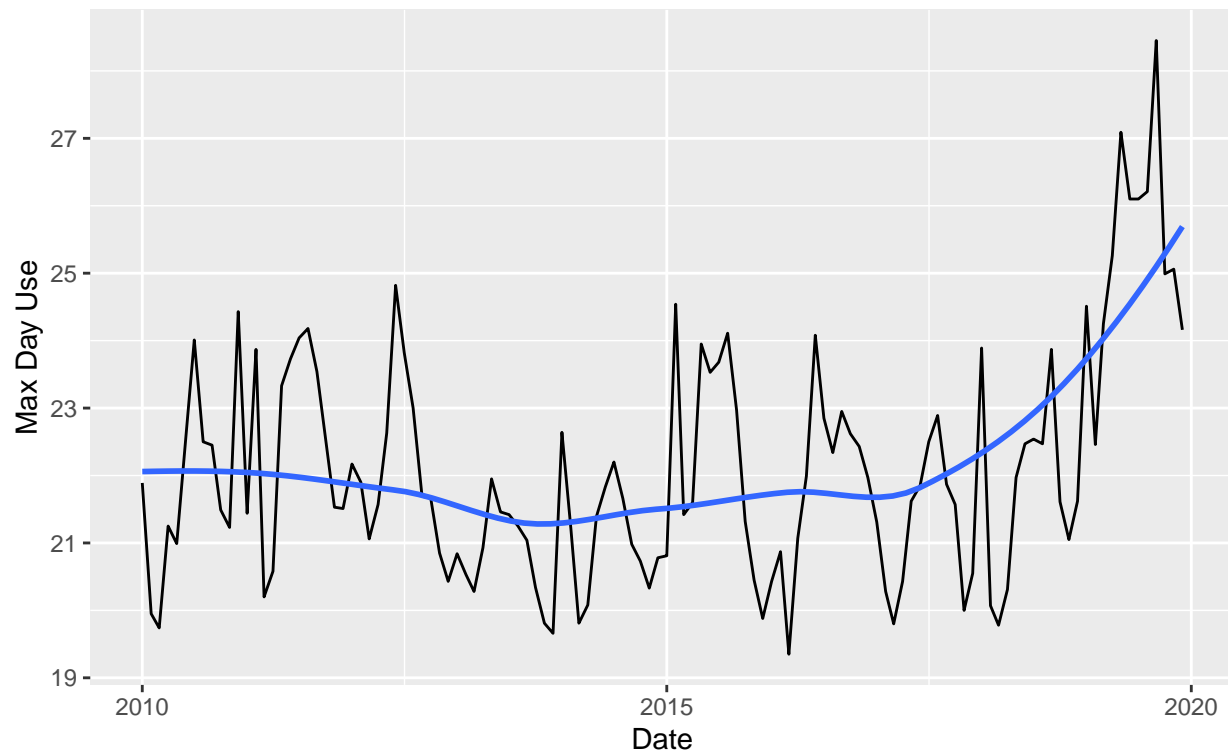
```
# 9. I used the code and function that I created above to plot Asheville's max
# daily withdrawal by months for the years 2010 through 2019 with a smooth line
# on the plot. Create a list of the year we want, the same length as the vector
# above.
years = rep(2010:2019)
pwsidcode = "01-11-010"
the_dfs <- years %>%
  map(scrape.it, pwsid = pwsidcode) %>%
  bind_rows()

# Plot the data.
ggplot(the_dfs, aes(y = Max.Day.Use, x = Date)) + geom_line() + geom_smooth(method = "loess",
  se = FALSE) + labs(title = paste("2010-2019 Asheville Monthly Maximum Daily Withdrawals"),
  subtitle = "GuruBandaa Khalsa", y = "Max Day Use", x = "Date")
```

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


2010–2019 Asheville Monthly Maximum Daily Withdrawals

GuruBandaa Khalsa



Question: Just by looking at the plot (i.e. not running statistics), does Asheville have a trend in water usage over time? Just by looking at the plot, Asheville appears to have a positive trend in water usage over time. It looks like it starts to increase more rapidly in 2018.