Assignment 09: Data Scraping

Student Name

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
getwd()
```

```
## [1] "C:/Users/kalli/OneDrive/Desktop/Grad_School/Environmental Data Analytics_ENV_872/EDA-Fall2022/As
```

```
library(tidyverse)
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.2.2
```

```
library(lubridate)
library(cowplot)

mytheme <- theme_bw(base_size = 13)+
  theme(plot.title = element_text(hjust = 0),
        axis.text = element_text(color = "black"))

theme_set(mytheme)
```

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
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
- PSWID
- 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
water.system.name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water.system.name
```

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

```
pwsid <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
pwsid
```

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

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

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

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

```
##  [1] "27.6400" "41.7900" "36.7200" "27.9700" "37.9500" "42.2400" "30.5400"
##  [8] "43.6200" "31.2800" "33.7600" "46.0800" "29.7800"
```

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…

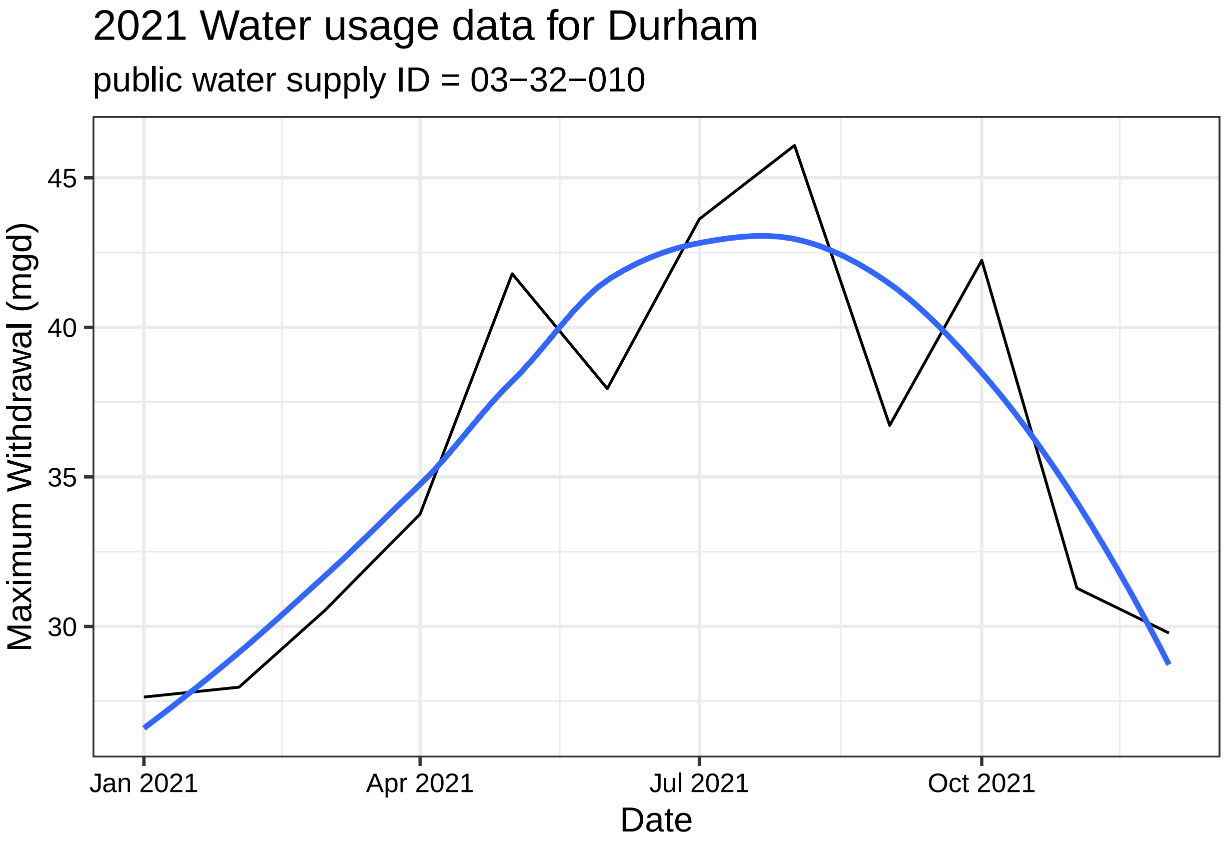
5. Create a line plot of the maximum daily withdrawals across the months for 2021

```r
#4
df_withdrawals <- data.frame("Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                             "Year" = rep(2021,12),
                            "Water.System.Name" = water.system.name,
                             "PWSID" = pwsid,
                             "Ownership" = ownership,
                             "Max.Withdrawal.mgd" = as.numeric(max.withdrawals.mgd)) %>%
  mutate(Date = my(paste(Month,"-",Year)))

head(df_withdrawals)
```

```
##   Month Year Water.System.Name     PWSID    Ownership Max.Withdrawal.mgd
## 1     1 2021            Durham 03-32-010 Municipality              27.64
## 2     5 2021            Durham 03-32-010 Municipality              41.79
## 3     9 2021            Durham 03-32-010 Municipality              36.72
## 4     2 2021            Durham 03-32-010 Municipality              27.97
## 5     6 2021            Durham 03-32-010 Municipality              37.95
## 6    10 2021            Durham 03-32-010 Municipality              42.24
##         Date
## 1 2021-01-01
## 2 2021-05-01
## 3 2021-09-01
## 4 2021-02-01
## 5 2021-06-01
## 6 2021-10-01
```

```
#5
durham.plot.2021 <- ggplot(df_withdrawals, aes(x=Date, y=Max.Withdrawal.mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2021 Water usage data for", water.system.name),
       subtitle = paste("public water supply ID =", pwsid),
       y="Maximum Withdrawal (mgd)",
       x="Date")

print(durham.plot.2021)
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



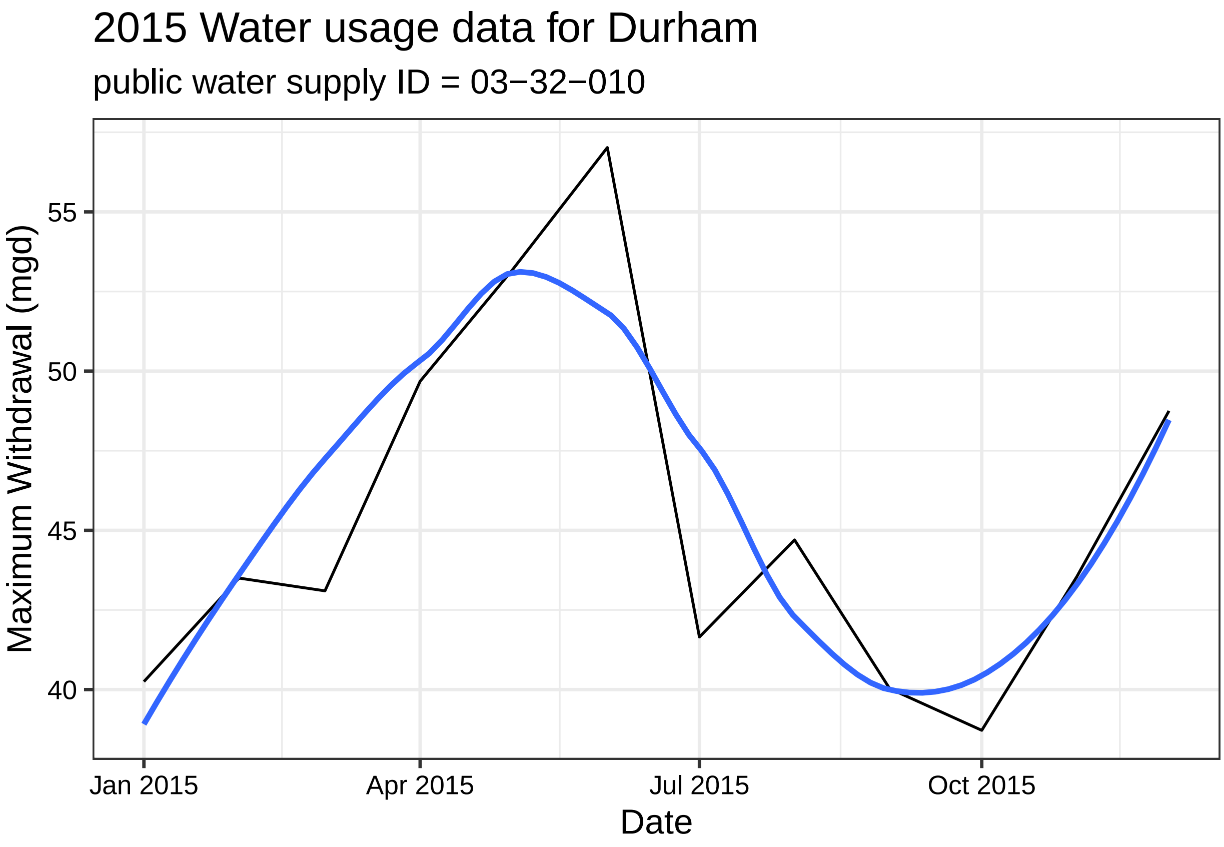
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped.**

```
#6.
scrape.it <- function(the_year, the_pwsid){

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

  #Set the element address variables (determined in the previous step)
  the_water.system.name_tag <- "div+ table tr:nth-child(1) td:nth-child(2)"
  the_pwsid_tag <- "td tr:nth-child(1) td:nth-child(5)"
  the_ownership_tag <- "div+ table tr:nth-child(2) td:nth-child(4)"
  the_max.withdrawal.mgd_tag <- "th~ td+ td"

  #Scrape the data items
  the_water.system.name <- the_website %>% html_nodes(the_water.system.name_tag) %>% html_text()
  the_pwsid <- the_website %>%   html_nodes(the_pwsid_tag) %>%  html_text()
  the_ownership <- the_website %>% html_nodes(the_ownership_tag) %>% html_text()
  the_max.withdrawal.mgd <- the_website %>% html_nodes(the_max.withdrawal.mgd_tag) %>% html_text()

  #create dataframe
  df_withdrawals <- data.frame("Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                               "Year" = the_year,
                               "Water.System.Name" = the_water.system.name,
                               "PWSID" = the_pwsid,
                               "Ownership" = the_ownership,
                               "Max.Withdrawal.mgd" = as.numeric(the_max.withdrawal.mgd)) %>%
  mutate(Date = my(paste(Month,"-",the_year)))

  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
the_year <- 2015
the_pwsid <- '03-32-010'

the_df <- scrape.it(the_year,the_pwsid)

durham.plot.2015 <- ggplot(the_df, aes(x=Date, y=Max.Withdrawal.mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste(the_year, "Water usage data for", the_df$Water.System.Name),
       subtitle = paste("public water supply ID =", the_df$PWSID),
       y="Maximum Withdrawal (mgd)",
       x="Date")

print(durham.plot.2015)
```

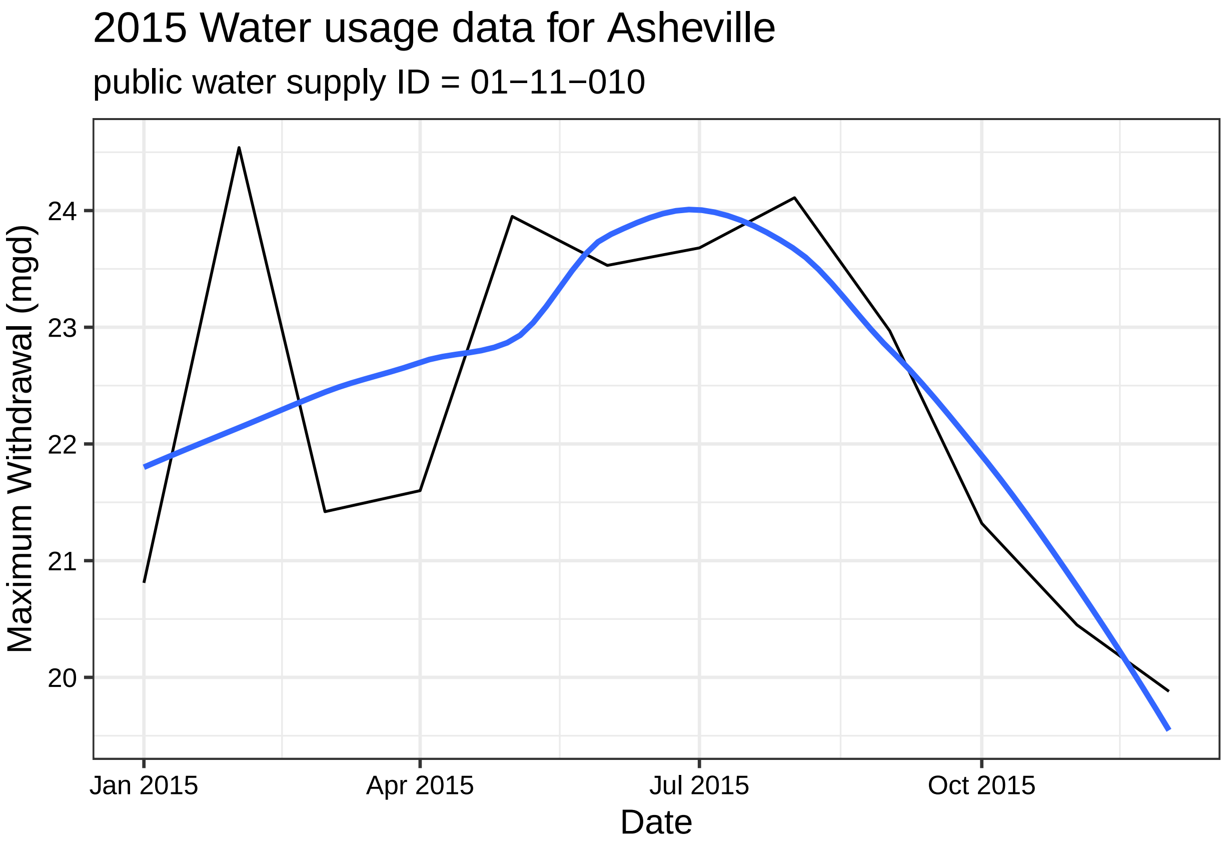
```
## `geom_smooth()` using formula 'y ~ x'
```



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
the_year <- 2015
the_pwsid <- '01-11-010'

the_df <- scrape.it(the_year,the_pwsid)

ashville.plot.2015 <- ggplot(the_df, aes(x=Date, y=Max.Withdrawal.mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste(the_year, "Water usage data for", the_df$Water.System.Name),
       subtitle = paste("public water supply ID =", the_df$PWSID),
       y="Maximum Withdrawal (mgd)",
       x="Date")

print(ashville.plot.2015)
```

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



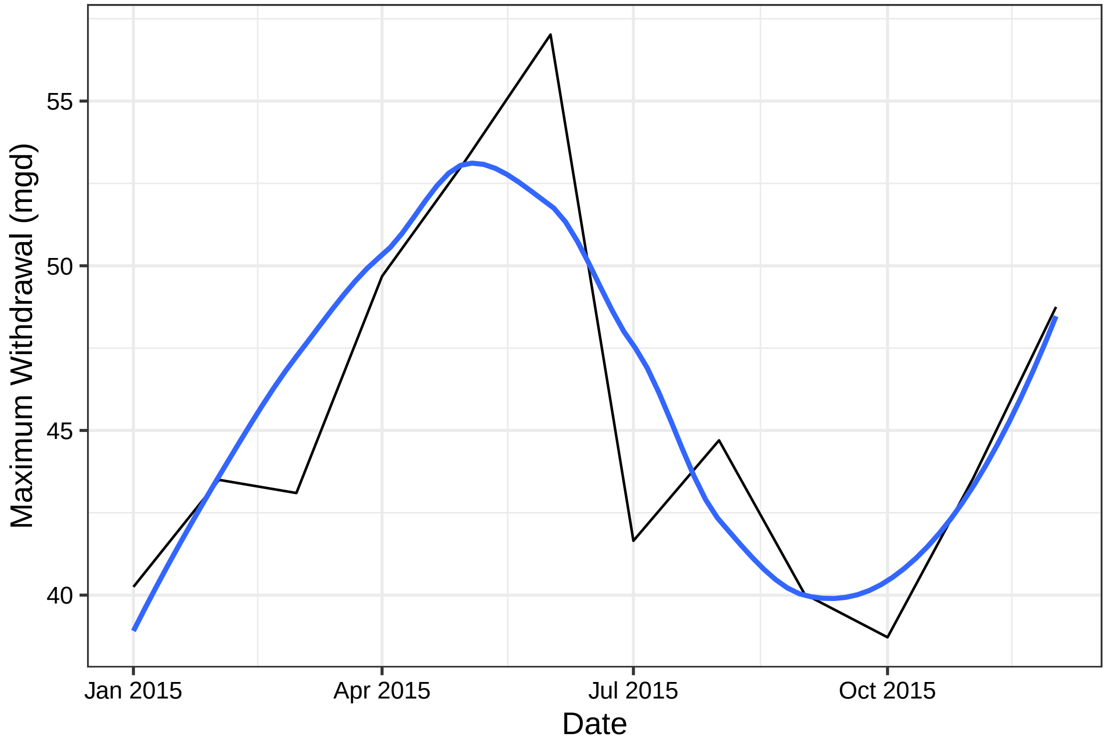
```
durham.ashville.combined <- plot_grid(durham.plot.2015,
                                ashville.plot.2015,
                                nrow = 2, ncol = 1,
                                align = "v"
                                )
```

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

```
durham.ashville.combined.1 <- cowplot::plot_grid(durham.ashville.combined,
                                          rel_widths = c(0.8,0.2))

print(durham.ashville.combined.1)
```

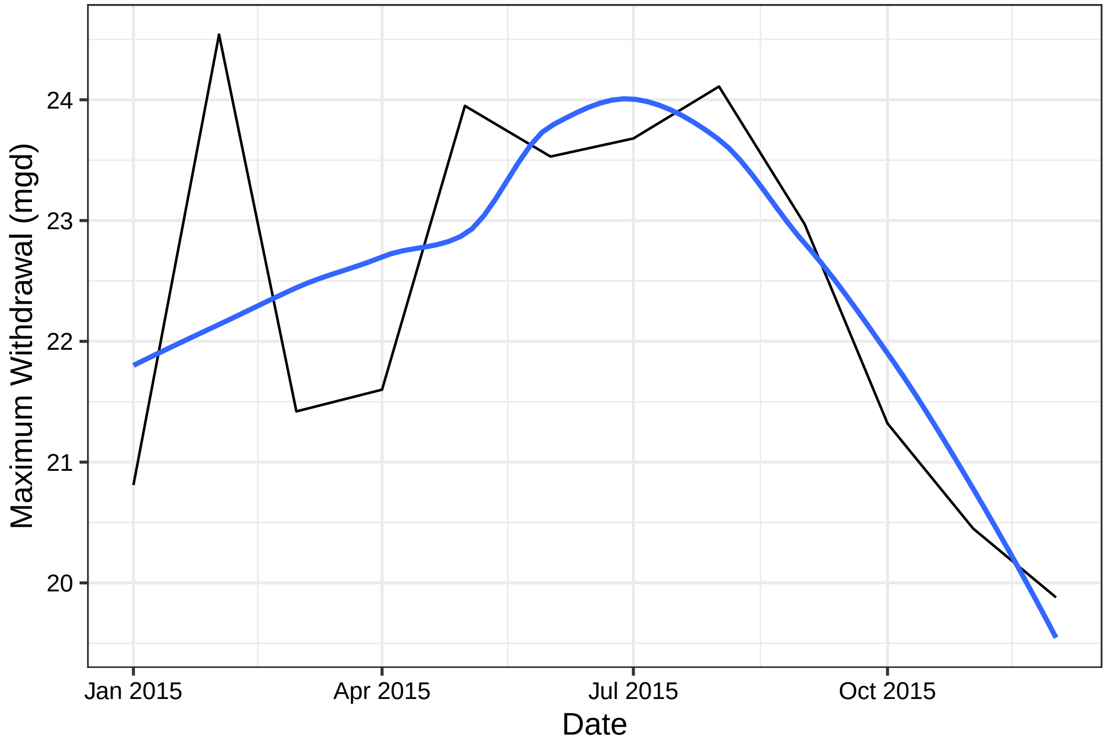
2015 Water usage data for Durham

public water supply ID = 03−32−010



2015 Water usage data for Asheville

public water supply ID = 01−11−010

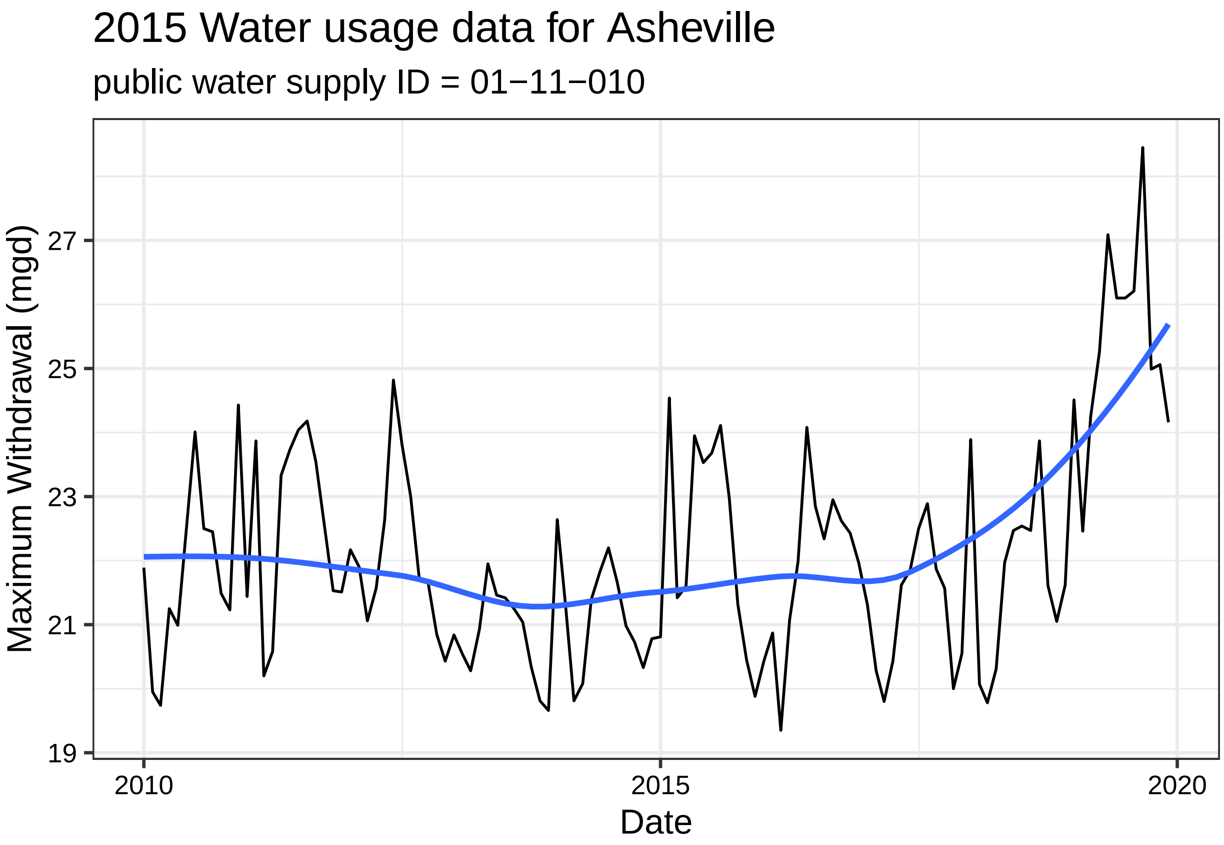


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

   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(2010:2019)
the_pwsid = '01-11-010'

#Use lapply to apply the scrape function
the_dfs <- lapply(X = the_years,
                  FUN = scrape.it,
                  the_pwsid)

#Conflate the returned dataframes into a single dataframe
the_df <- bind_rows(the_dfs)

#Plot, because it's fun and rewarding
ashville.plot.2010.2019 <- ggplot(the_df, aes(x=Date, y=Max.Withdrawal.mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste(the_year, "Water usage data for", the_df$Water.System.Name),
       subtitle = paste("public water supply ID =", the_df$PWSID),
       y="Maximum Withdrawal (mgd)",
       x="Date")

print(ashville.plot.2010.2019)
```

```
## `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? There appears to be an increase in water usage over time in Ashville. However, this trend appears to be nonlinear.