INF550 Section 8.16

Natasha Wesely

2022-11-08

USGS Coding Lab Exercises

Recently NOAA published a press release demonstrating that 2020 was both the hottest and driest summer on record for Arizona. In this lab we will look at USGS NWIS stream gauge and preciptitation data to investigate just how anomalous 2020 data are.

Question 1

Use the readNWISstat function to retrieve the following statewide data for 2015-2020: 1. Precipitation total (inches/week) 2. Streamflow (ft^3/s)

```
# Download precip data
# for some reason the weekly data is messed up
# so I'm downloading daily data
precip <- readNWISdata(stateCd="AZ",
                       service="dv",
                       parameterCd="00045",
                       startDate="2015-01-01",
                       endDate="2020-12-31")

# download steam flow data
flow <- readNWISdata(stateCd="AZ",
                     service="dv",
                     parameterCd="00060",
                     startDate="2015-01-01",
                     endDate="2020-12-31")
```

Question 2

Create two timeseries plots (one for precipitation, one for streamflow), where color is a function of year (e.g. the x axis is month, the y axis is precipitation, legend shows color by year).

```
library(lubridate, quietly = T)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
precip$year = factor(year(precip$dateTime))
precip$month = month(precip$dateTime)
flow$year = factor(year(flow$dateTime))
```

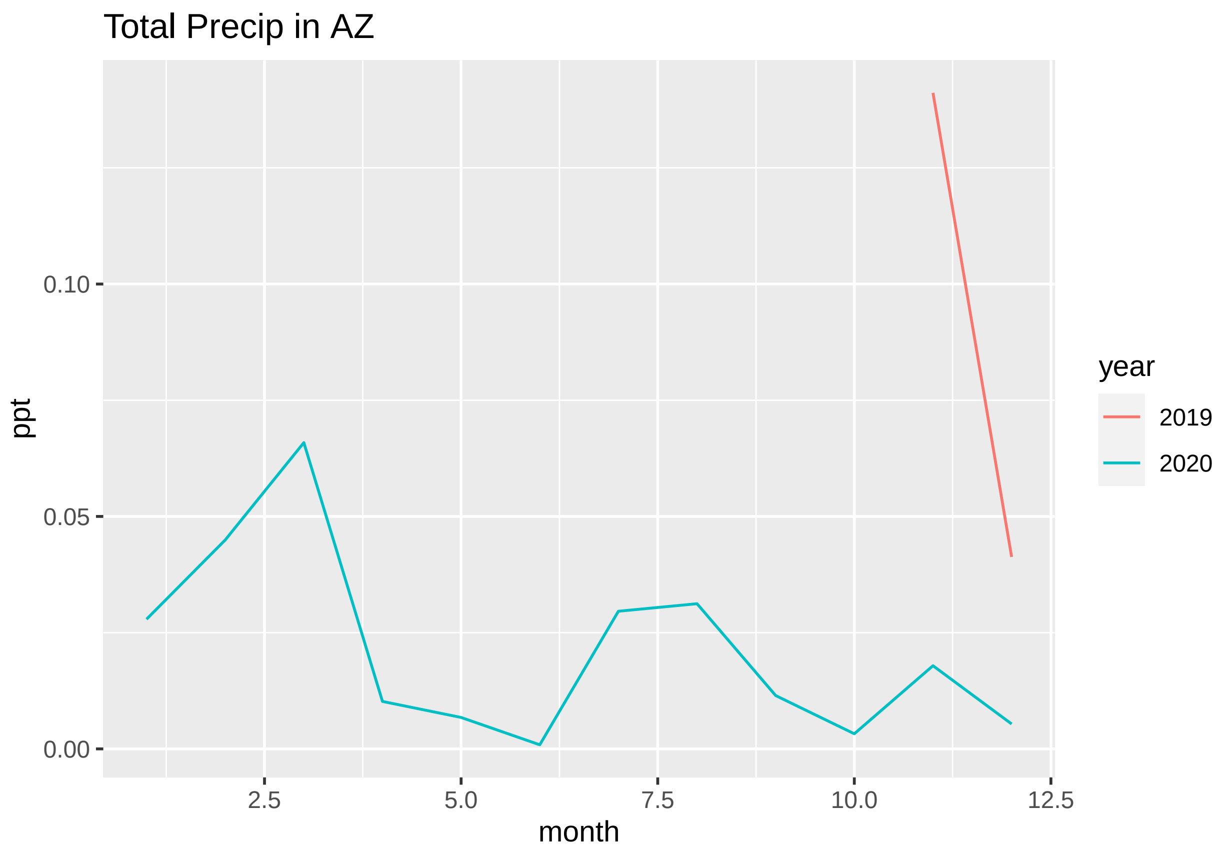
```r
flow$month = month(flow$dateTime)

library(tidyverse, quietly = T)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5 
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1 
## v readr   2.1.3      v forcats 0.5.2 
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x dplyr::id()              masks geoknife::id()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

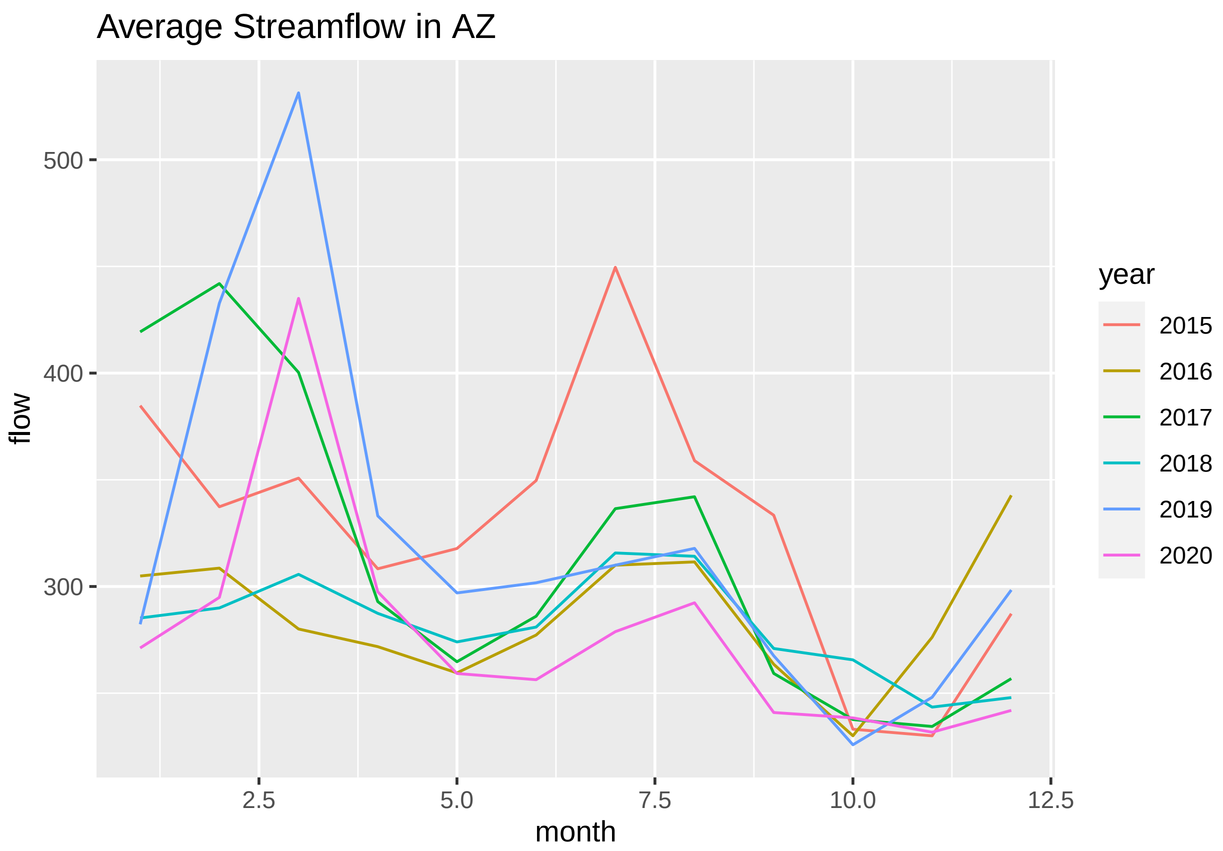
```r
library(ggplot2, quietly = T)
precip %>%
  group_by(year,month) %>%
  summarise(ppt = mean(X_00045_00006, na.rm = T))%>%
  ggplot() +
  geom_line(aes(x = month,
                y = ppt,
                color = year))+
  labs(title = "Total Precip in AZ")
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```



```
flow %>%
  group_by(year,month) %>%
  summarise(flow = mean(X_00060_00003, na.rm = T))%>%
  ggplot() +
  geom_line(aes(x = month,
                y = flow,
                color = year))+
  labs(title = "Average Streamflow in AZ")
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```



Question 3

Calculate the monthly mean precipitation and streamflow from 2015-2019, and use that mean to calculate a 2020 anomaly timeseries. Create two new plots (like #2 above) with the 2015-2019 mean as a thick black line, and 2020 anomaly as a thin red line.

Since the API cannot pull precip data for AZ for most of the date range, I can't make this plot for the precip data. The API is pulling stream flow data for 2015-2020 though, so I can make this plot for the stream flow data.

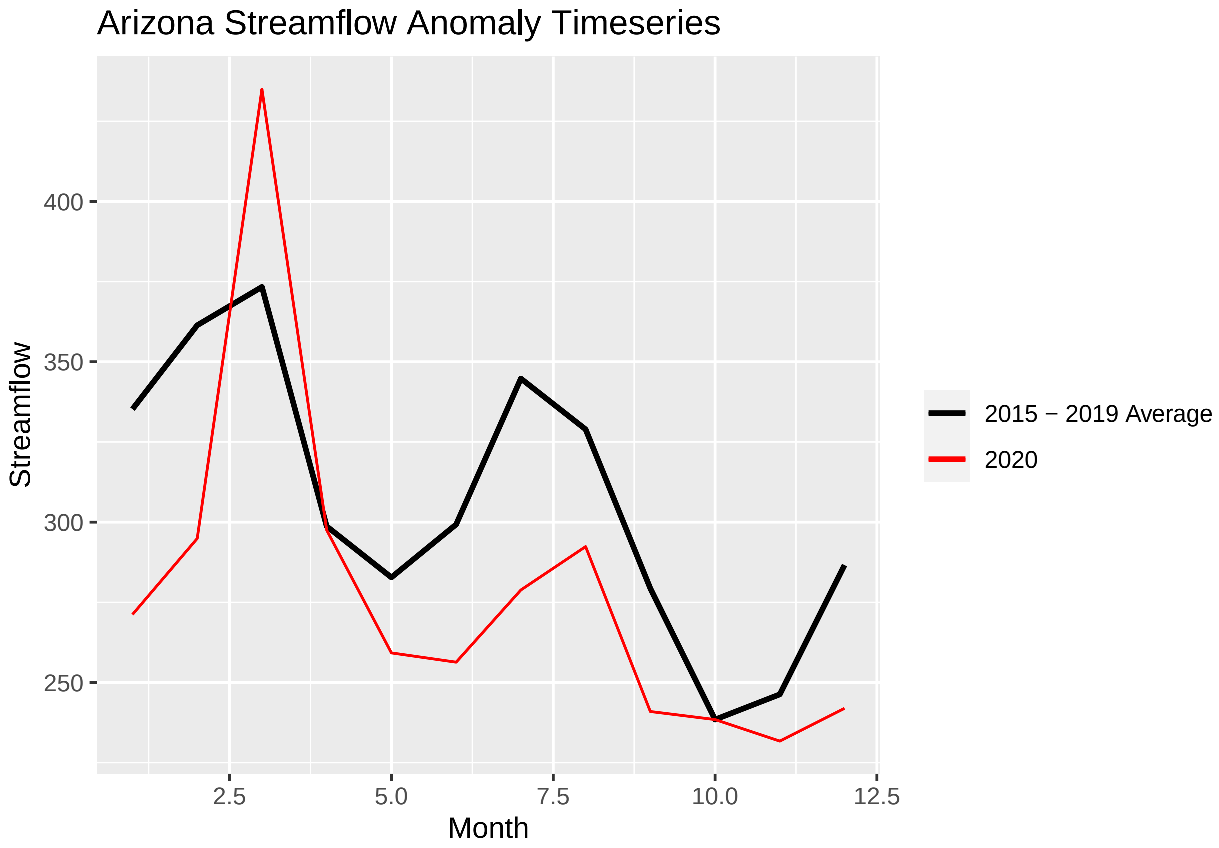
```
flow.monMean =  flow %>%
  filter(year != 2020) %>%
  group_by(month) %>%
  summarise(monthMean = mean(X_00060_00003, na.rm = T))

flow.2020 = flow %>%
  filter(year == 2020) %>%
  group_by(month) %>%
  summarise(monthMean = mean(X_00060_00003, na.rm = T))


ggplot() +
  geom_line(data = flow.monMean,
            aes(x = month, y = monthMean, color = "2015 - 2019 Average"),
            # color = "black",
            size = 1) +
  geom_line(data = flow.2020,
```

```r
            aes(x = month, y = monthMean, color = "2020"),
            # color = "red",
            # size = 1.5
            ) +
  scale_color_manual(values = c("black", "red"))+
  labs(title = "Arizona Streamflow Anomaly Timeseries",
       x = "Month",
       y = "Streamflow",
       color = "")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```


Arizona Streamflow Anomaly Timeseries
Streamflow
Month
2015 − 2019 Average