EDAV Fall 2019 PSet 4

Read *Graphical Data Analysis with R*, Ch. 9, 11

Read *Interactive Data Visualization for the Web*, Ch. 3, 5

The theme of this problem set is freedom. As you will see, you'll have more choices than usual in terms of data and packages.

Remember that you are expected to meet with your partner in person–even if you are working on different parts there is a benefit to having someone to ask questions to or spot the typo in your code. You are also expected to communicate frequently, pull your weight, and not make unilateral decisions. It's all about collaboration and partnership.

Grading is based both on your graphs and verbal explanations. Follow all best practices as discussed in class.

3. Cause of Death

Data: https://wonder.cdc.gov/ucd-icd10.html

a. Create a series of choropleth maps in which only one variable changes, such as level of a factor variable or time.

For inspiration, see these examples:

https://www.nytimes.com/interactive/2017/06/30/upshot/the-best-and-worst-new-york-neighborhoods.html

https://www.nytimes.com/interactive/2017/10/05/upshot/gun-ownership-partisan-divide.html

b. Interpret the graphs you drew in (a).

4. Mosaic plot (SVG / D3)

a. Manually create a 2 x 2 mosaic plot of party affiliation by gender for (House) representatives currently in the U.S. Congress using SVG. Data is available here in `.csv` form: https://github.com/unitedstates/congress-legislators

You may remove any Independents.

The SVG should be 500 x 400 pixels and included in your `.Rmd` file between svg tags (**not** in a code chunk):

```
## set work directiory
setwd("~/workworkwork")
## Read data and save it into df
df = read.csv("legislators-current.csv")
## Remove any independents
df_new = df[df["party"] != "Independent",]

df_new$party <- factor(df_new$party, levels = unique(df_new$party))

df_test = df_new[, c("party", "gender")]
## find 4 values to draw the mosaic plot
f_Dem = sum(df_test$gender == "F" & df_test$party == "Democrat")
f_Rep = sum(df_test$gender == "F" & df_test$party == "Republican")
m_Dem = sum(df_test$gender == "M" & df_test$party == "Democrat")
m_Rep = sum(df_test$gender == "M" & df_test$party == "Republican")
## find the length  in the mosaic plot
f_Dem_ratio = f_Dem / (f_Dem + m_Dem)
m_Dem_ratio = m_Dem / (m_Dem + f_Dem)
f_Rep_ratio = f_Rep / (f_Rep + m_Rep)
m_Rep_ratio = m_Rep / (m_Rep + f_Rep)

f_Dem_len = f_Dem_ratio * 350
m_Dem_len = m_Dem_ratio * 350
f_Rep_len = f_Rep_ratio * 350
m_Rep_len = m_Rep_ratio * 350

f_Dem_len
```

```
## [1] 133.5689
```

```
m_Dem_len
```

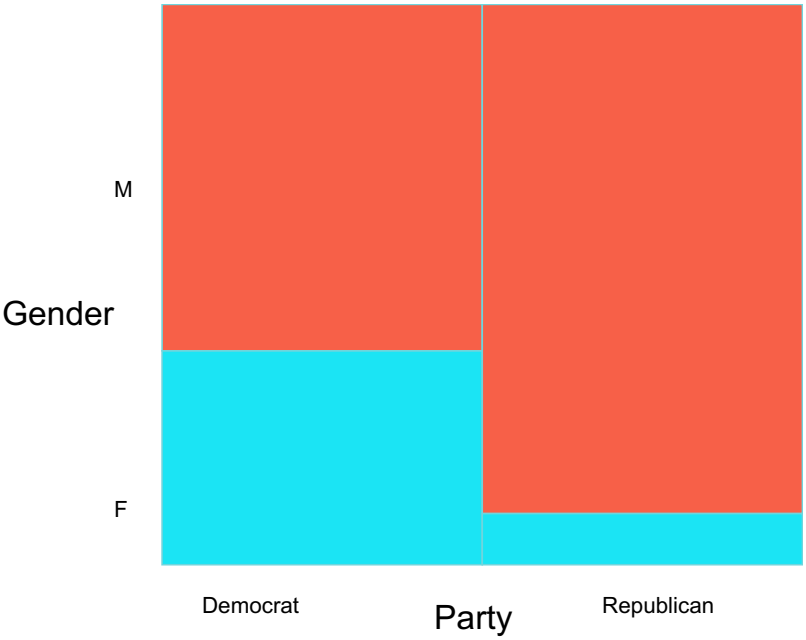
```
## [1] 216.4311
```

```
f_Rep_len
```

```
## [1] 31.94444
```

```
m_Rep_len
```

```
## [1] 318.0556
```



The axes and levels of each variable should be labeled.

(Don't despair, this is the only time you will have to create SVG by hand for this class!)

b. Change your code from (a) so that with the exception of the blank SVG, the mosaic plot is completely created with D3, based on a dataset that you provide that contains the 4 values you calculated for part (a). Your code should work if the values change.

There are two options for including D3 directly in your `.Rmd` file:

i. You can include it as before in the "non code" section of the file using the template below. If you go this route, you will likely prefer to work in a text editor with a browser open on the other half of your screen, or in some other manner which will allow you to view your visualization as you go.

ii. Use the **r2d3** package. The setup here is different. You will create your visualization in a `.js` file and then call it from an R chunk. More information is available on the package site: https://rstudio.github.io/r2d3/ (Note that you no longer need the preview version of RStudio.) If you use **r2d3**, please erase the template code in i.

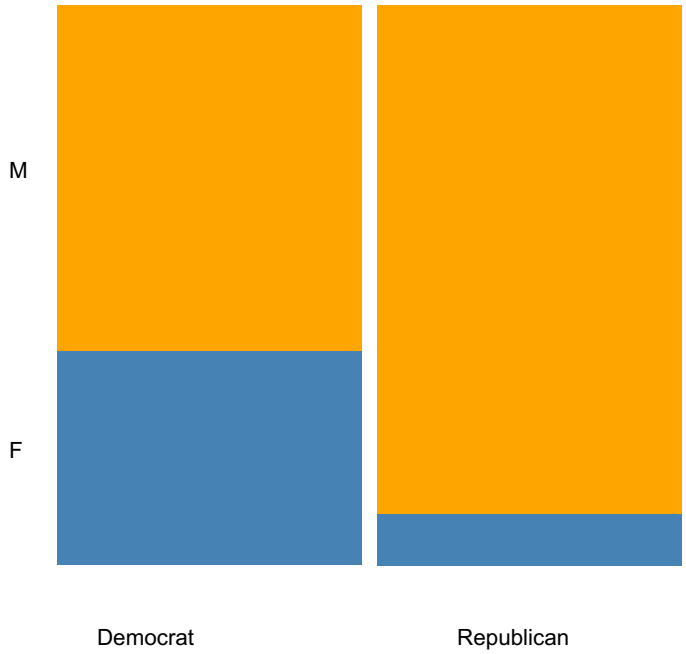
```
library(r2d3)
bars <- df_test

test = c(m_Dem_len, f_Dem_len, m_Rep_len, f_Rep_len)
```

```
var mycolor = ['orange', 'steelblue']
    label = ['M', 'F', 'Democrat', 'Republican']
svg.selectAll('rect')
    .style("500px", "400px")
    .data(data)
    .enter()
      .append('rect')
        .attr('x',  function(d, i) {
          if(i == 0 || i == 1){
            return  100
            }
          else{
            return 300}
          }
        )
        .attr('y',  function(d, i) {
          if(i == 0 || i == 2)
            return 0
          else if (i == 1)
            return data[0]
          else
            return data[2]
        }
        )
        .attr('width', '190px' )
        .attr('height', function(d, i) {
            return d
          }
        )
        .attr('fill', function(d, i){
          if(i == 0 || i == 2)
            return mycolor[0]
          else
            return mycolor[1]
          }
        )

texts = svg.selectAll('.myText')
            .data(data)
            .enter()
            .append('text');

texts.attr('x', function(d, i) {
        if(i == 0 || i == 1){
          return 70
          }
        else if(i == 2){
          return 125
          }
        else
          return 350
            }
          )
     .attr('y', function(d, i) {
        if(i == 0){
          return data[0]/2
        }
        else if(i == 1){
          return data[0] + data[1] / 2
        }
        else
          return 400
        }
      )
     .text(function(d,i) {return label[i]})
```


M
F
Democrat
Republican