GES LAB 7

Justin Johnson

04/2020

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

PART I. Initial Setup

```
#setup
knitr::opts_knit$set(root.dir = "C:/Users/justi/Downloads/School/GES 687/LAB 7/Lab 7 R Code")
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.4
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.5
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'dplyr' was built under R version 4.0.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidycensus)
```

```
## Warning: package 'tidycensus' was built under R version 4.0.4
```

```
library(ggplot2)
#theme_set(theme_bw()) uncomment to use the bw theme in all ggplot maps
library(sf)
```

```
## Warning: package 'sf' was built under R version 4.0.4
```

```
## Linking to GEOS 3.8.0, GDAL 3.0.4, PROJ 6.3.1
```

```
library(sp)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(janitor)
```

```
## Warning: package 'janitor' was built under R version 4.0.4
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(readr)

# installing the Bi_scale package
library(biscale)
```

```
## Warning: package 'biscale' was built under R version 4.0.5
```

```
#installing Cowplot package
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.0.5
```

```
#Loading package for uploading picture
library(png)
```

```
acs_variable_list = load_variables(2019,"acs5", cache= TRUE)

write.csv(acs_variable_list,'acs_variable_list_2018.csv', row.names = FALSE)
```

PART II. Questions

Question 1

Use `tidycensus` to download 1. race/ethnicity (B03002) and 2. median household income for Baltimore City. Store this data in a new object. Choose which race/ethnicity you'd like to relate to income (Non-Hispanic Black and Non-Hispanic White work best). Which census tract has the highest *percentage* of your target race/ethnicity (and what is the percent) and which has the highest median household income (and how much is it?)? (5 points)

```
# Loading In Baltimore ACS Data

Baltimoretractdata = get_acs(geography = "tract", year=2019, state = "MD", survey="acs5", county
= "Baltimore City",
                 variables = c("Total Pop" = "B01003_001",
"Black Pop" = "B01001B_001",
                               "med_income" = "B19013_001"),
                 geometry = TRUE,
                 output = "wide") %>% clean_names()
```

```
## Getting data from the 2015-2019 5-year ACS
```

```
## Downloading feature geometry from the Census website.  To cache shapefiles for use in future
sessions, set `options(tigris_use_cache = TRUE)`.
```

```
##
  |
  |                                                                      |   0%
  |
  |=                                                                     |   2%
  |
  |===                                                                   |   4%
  |
  |=====                                                                 |   6%
  |
  |======                                                                |   8%
  |
  |=======                                                               |  10%
  |
  |=========                                                             |  12%
  |
  |=========                                                             |  13%
  |
  |===========                                                           |  15%
  |
  |===========                                                           |  16%
  |
  |=============                                                         |  18%
  |
  |==============                                                        |  20%
  |
  |================                                                      |  21%
  |
  |================                                                      |  24%
  |
  |==================                                                    |  25%
  |
  |====================                                                  |  27%
  |
  |====================                                                  |  29%
  |
  |======================                                                |  30%
  |
  |======================                                                |  32%
  |
  |========================                                              |  33%
  |
  |========================                                              |  35%
  |
  |==========================                                            |  37%
  |
  |==========================                                            |  38%
  |
  |============================                                          |  41%
  |
  |============================                                          |  43%
  |
  |==============================                                        |  46%
```

```
  |
  |=================================                                     |  48%
  |
  |==================================                                    |  49%
  |
  |====================================                                  |  51%
  |
  |=====================================                                 |  55%
  |
  |======================================                                |  57%
  |
  |======================================                                |  58%
  |
  |=======================================                               |  60%
  |
  |========================================                              |  63%
  |
  |=========================================                             |  66%
  |
  |==========================================                            |  68%
  |
  |===========================================                           |  71%
  |
  |============================================                          |  74%
  |
  |=============================================                         |  78%
  |
  |==============================================                        |  79%
  |
  |================================================                      |  82%
  |
  |=================================================                     |  87%
  |
  |==================================================                    |  88%
  |
  |===================================================                   |  91%
  |
  |=====================================================                 |  93%
  |
  |======================================================                |  96%
  |
  |======================================================================| 99%
  |
  |======================================================================| 100%
```

```
#Measuring Census Tract with highest Black Population
Baltimoretractdata$blackprop = Baltimoretractdata$black_pop_e / Baltimoretractdata$total_pop_e
max(Baltimoretractdata$black_pop_e)
```

```
## [1] 7141
```

```
baltdrop2 = Baltimoretractdata$blackprop[!is.na(Baltimoretractdata$blackprop)]
max(baltdrop2)
```

```
## [1] 0.9931774
```

```
#The highest black population belongs to census tract 1511 with a black population of 7141 peopl
e. The highest proportion of black residents in a census tract belongs to census tract 2007 with
a 99% black population




#Meausring Census tract with highest median income

baltdata1 = Baltimoretractdata %>% filter(!is.na(Baltimoretractdata$med_income_e))

max(baltdata1$med_income_e)
```

```
## [1] 195156
```

```
#The largest income belongs to census tract 2711 with a median income of $209,688
```

Question 2

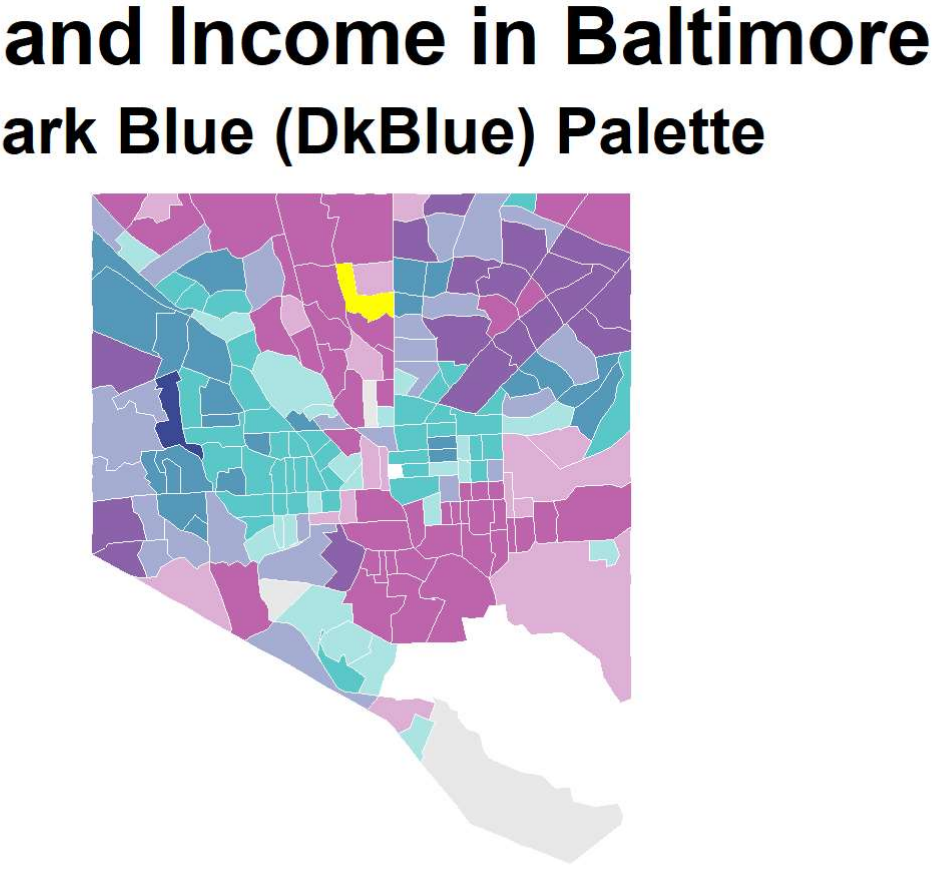
Please reproject this data to Web Mercator. (1 points)

```
#Transforming Baltimore Map to Web Mercator

baltdata1transform = st_transform(baltdata1, 3857)
```

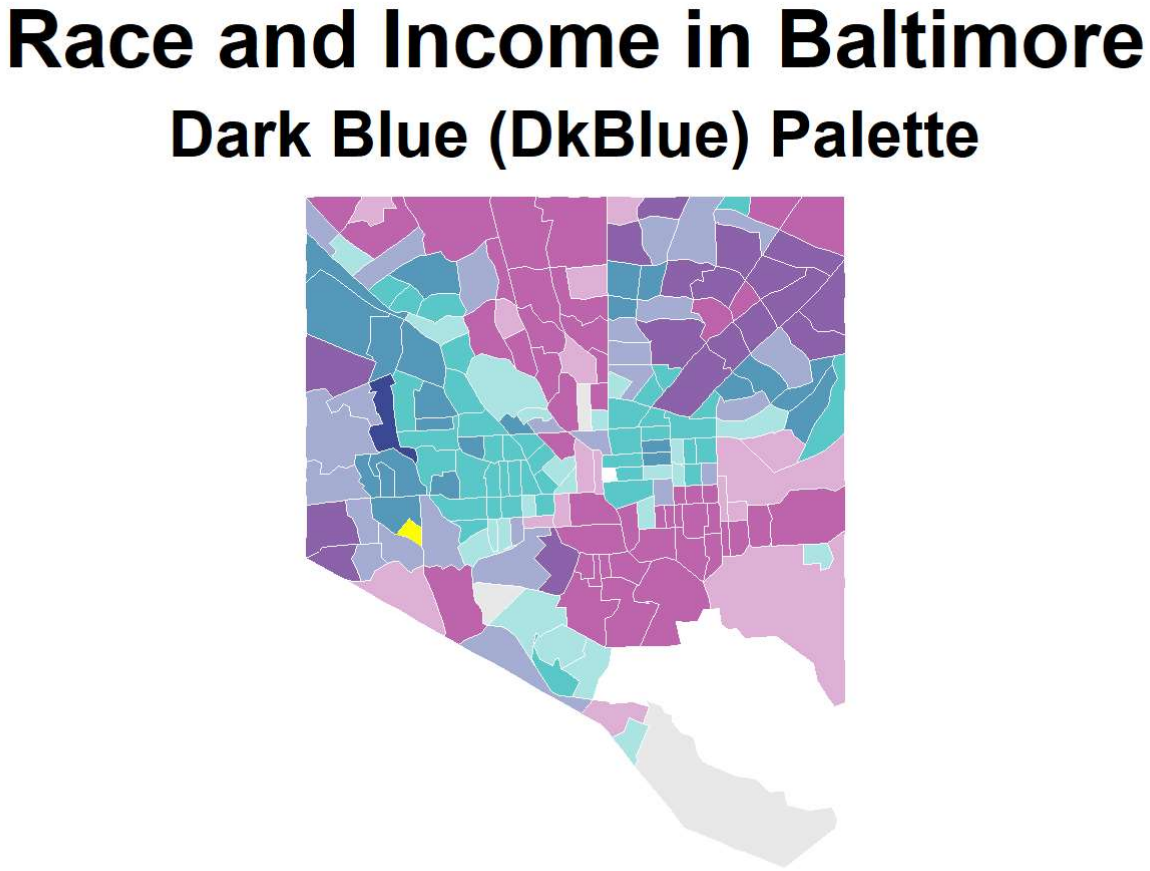
Question 3

Create two plots. In the first plot highlight the tract with the highest concentration of your selected race/eth. In the second plot highlight the tract with the highest median household income? (5 points)

```
# Creating New column
bidata = bi_class(baltdata1, x = blackprop, y = med_income_e, style = "quantile", dim = 3)




highlight1 = baltdata1 %>% filter(med_income_e == 195156)

highlight2 = baltdata1 %>% filter(blackprop > 0.99)



ggplot(bidata) +
  geom_sf(mapping = aes(fill = bi_class), color = "white", size = 0.1, show.legend = FALSE) +
   bi_scale_fill(pal = "DkBlue", dim = 3) +
  geom_sf(data = highlight1, fill = "yellow", color = NA) +
  labs(
    title = "Race and Income in Baltimore",
    subtitle = "Dark Blue (DkBlue) Palette") +
  bi_theme()
```



```
#Census Tract with highest median income is highlighted yellow on the map




ggplot(bidata) +
  geom_sf(mapping = aes(fill = bi_class), color = "white", size = 0.1, show.legend = FALSE) +
   bi_scale_fill(pal = "DkBlue", dim = 3) +
  geom_sf(data = highlight2, fill = "yellow", color = NA) +
  labs(
    title = "Race and Income in Baltimore",
    subtitle = "Dark Blue (DkBlue) Palette") +
  bi_theme()
```




```
  # Census Tract with highest proportion of black residents is highlighted    yellow
```

Question 4.

Create a third column using the bi_class function from the tutorial. (2 points)

```
# Creating New column
bidata = bi_class(baltdata1, x = blackprop, y = med_income_e, style = "quantile", dim = 3)
```

Question 5

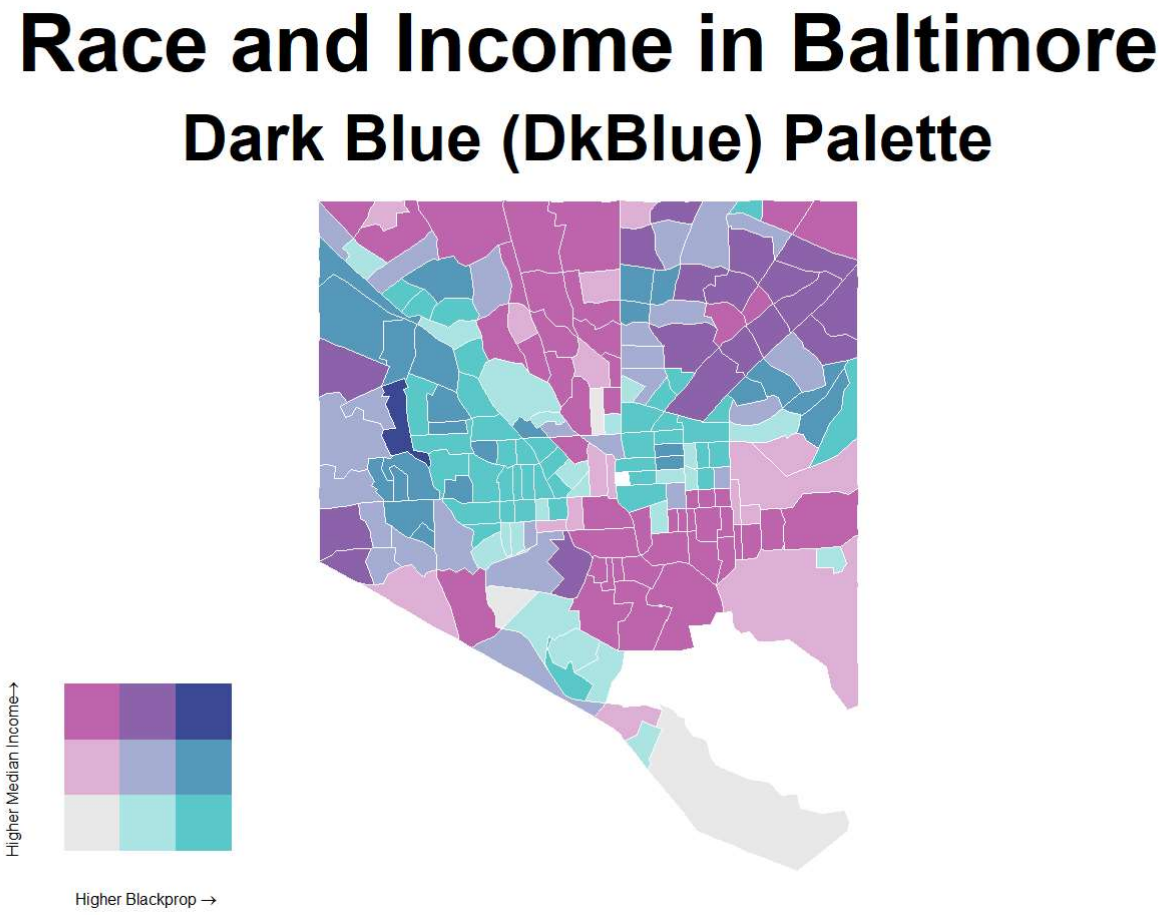
Create a bivariate map with your data. (3 points)

```
bivariatemap =
  ggplot() +
  geom_sf(data = bidata, mapping = aes(fill = bi_class), color = "white", size = 0.1, show.legen
d = FALSE) +
  bi_scale_fill(pal = "DkBlue", dim = 3) +
  labs(
    title = "Race and Income in Baltimore",
    subtitle = "Dark Blue (DkBlue) Palette") +
  bi_theme()
```

Question 6

Use the cowplot package and ggdraw, like in the tutorial to add a legend (2 points).

```
#Creating Legend
legend <- bi_legend(pal = "DkBlue",
                    dim = 3,
                    xlab = "Higher Blackprop",
                    ylab = "Higher Median Income",
                    size = 6)




legendplot_x <- ggdraw() +
  draw_plot(bivariatemap, 0, 0, 1, 1) +
  draw_plot(legend, .01, .01, .3, .3)



legendplot_x
```



Question 7

Rinse and repeat for another county of your choosing, using a *different* color scheme. Be sure to use Psuedo-Mercator (3857). (5 points)

```
Arundeltractdata = get_acs(geography = "tract", year=2019, state = "MD", survey="acs5", county =
"Anne Arundel",
                 variables = c("Total Pop" = "B01003_001",
"Black Pop" = "B01001B_001",
                               "med_income" = "B19013_001"),
                 geometry = TRUE,
                 output = "wide") %>% clean_names()
```

```
## Getting data from the 2015-2019 5-year ACS
```

```
## Downloading feature geometry from the Census website.  To cache shapefiles for use in future
sessions, set `options(tigris_use_cache = TRUE)`.
```

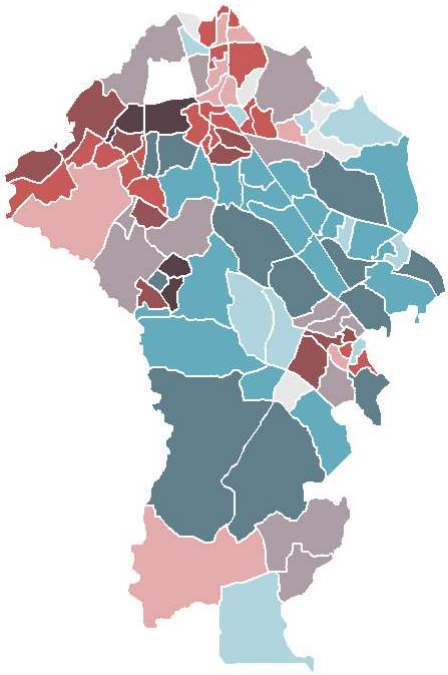
```
#Measuring Census Tract with highest Black Population
Arundeltractdata$blackprop = Arundeltractdata$black_pop_e / Arundeltractdata$total_pop_e
max(Arundeltractdata$black_pop_e)
```

```
## [1] 7025
```

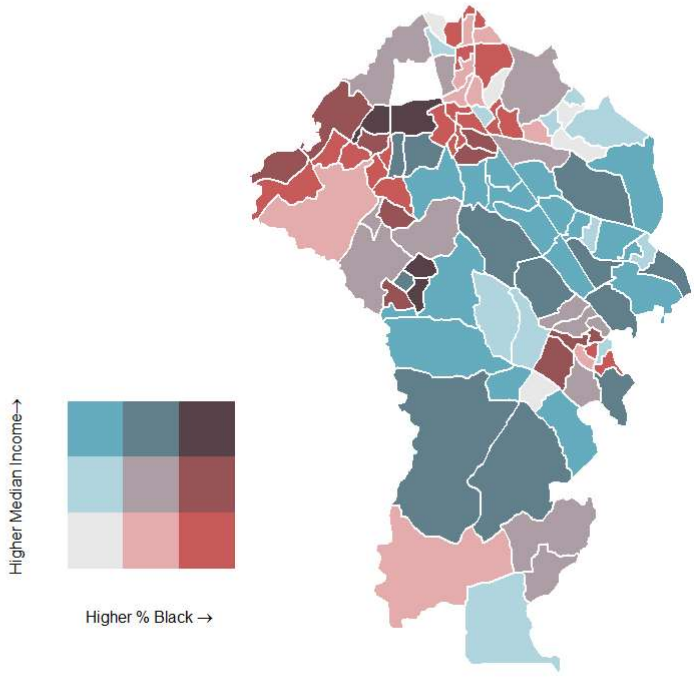
```
arundeldrop = Arundeltractdata$blackprop[!is.na(Arundeltractdata$blackprop)]


#Meausring Census tract with highest median income
arundeldata1 = Arundeltractdata %>% filter(!is.na(Arundeltractdata$med_income_e))


#Creating Bi_class Data for Arundel County
arundelbidata = bi_class(arundeldata1, x = blackprop, y = med_income_e, style = "quantile", dim
 = 3)



#Mapping Bivariate Data for Arundel
arundelbivariate =
  ggplot() +
  geom_sf(data = arundelbidata, mapping = aes(fill = bi_class), color = "white", size = 0.5, sho
w.legend = FALSE) +
  bi_scale_fill(pal = "GrPink", dim = 3) +
  labs(
    title = "Race and Income in Anne Arundel",
    subtitle = "Gray and Pink (GrPink) Palette") +
  bi_theme()

 ggplot() +
  geom_sf(data = arundelbidata, mapping = aes(fill = bi_class), color = "white", size = 0.5, sho
w.legend = FALSE) +
  bi_scale_fill(pal = "GrPink", dim = 3) +
  labs(
    title = "Race and Income in Anne Arundel",
    subtitle = "Gray and Pink (GrPink) Palette") +
  bi_theme()
```

Race and Income in Anne Arundel

Gray and Pink (GrPink) Palette



```
 #Adding Leged to Arundel Map
 legend2 <- bi_legend(pal = "GrPink",
                      dim = 3,
                      xlab = "Higher % Black",
                      ylab = "Higher Median Income",
                      size = 6)



legendplot_x2 <- ggdraw() +
  draw_plot(arundelbivariate, 0, 0, 1, 1) +
  draw_plot(legend2, 0.1, 0.1, .3, .3)

legendplot_x2
```



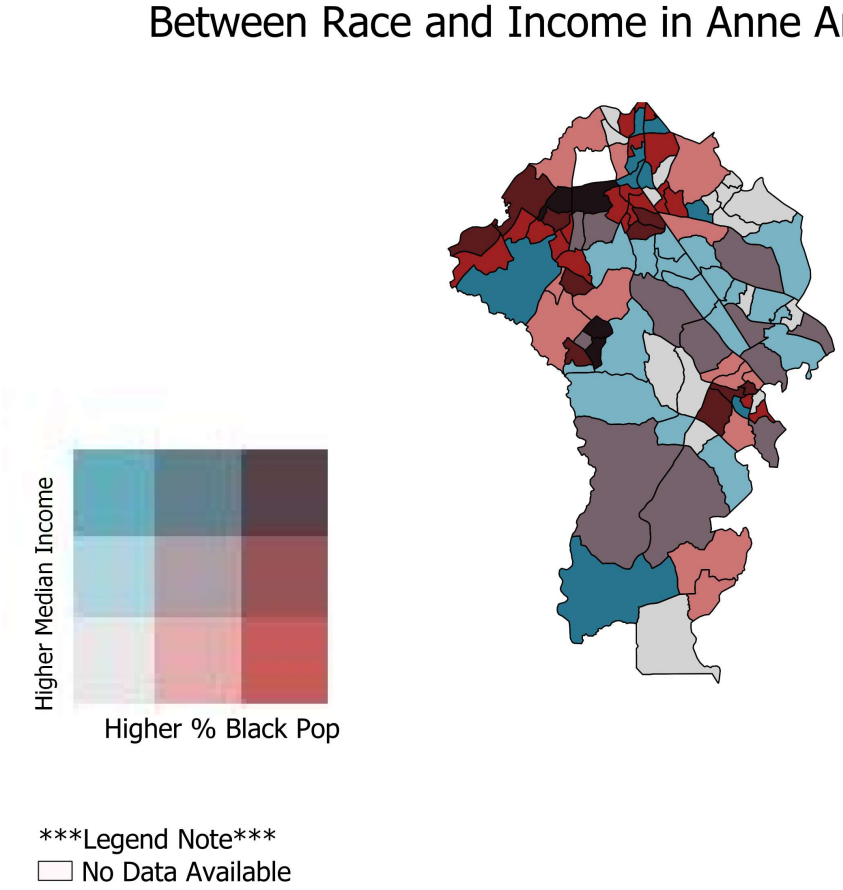
Question 8

Write the bi_class output to a geojson file. (1 points)

```
st_write(arundelbidata, "GESLab8map.geojson")
```

Question 9

Now open your Geojson output and create a QGIS map of your bivariate map. Put an image of that map here. (2 points)



Bivariate QGIS

Question 10

Use qgis2web and put a link here to your github site with the webmap of your bivariate map. (3 points)

```
# You can access my web map by clicking on the link below:

# https://jjustin1.github.io/GES687Lab8/#10/38.9733/-76.5871
```

PART III. Reflection

```
#This assignment has helped with better understanding how to format maps in both R and QGIS. I b
ecame much more comfortable with learning how to edit the specific data within a map, and transf
orming it to drop and generate variables. This assignment would also help a lot with my final pr
oject because I learned how to better edit the colors of my maps and highlight specific areas of
interest.
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.