STA234: HW4

Derin Gezgin

2025-03-06

## Importing the Required Libraries

library(ggplot2)  
library(ggmap)  
library(dplyr)

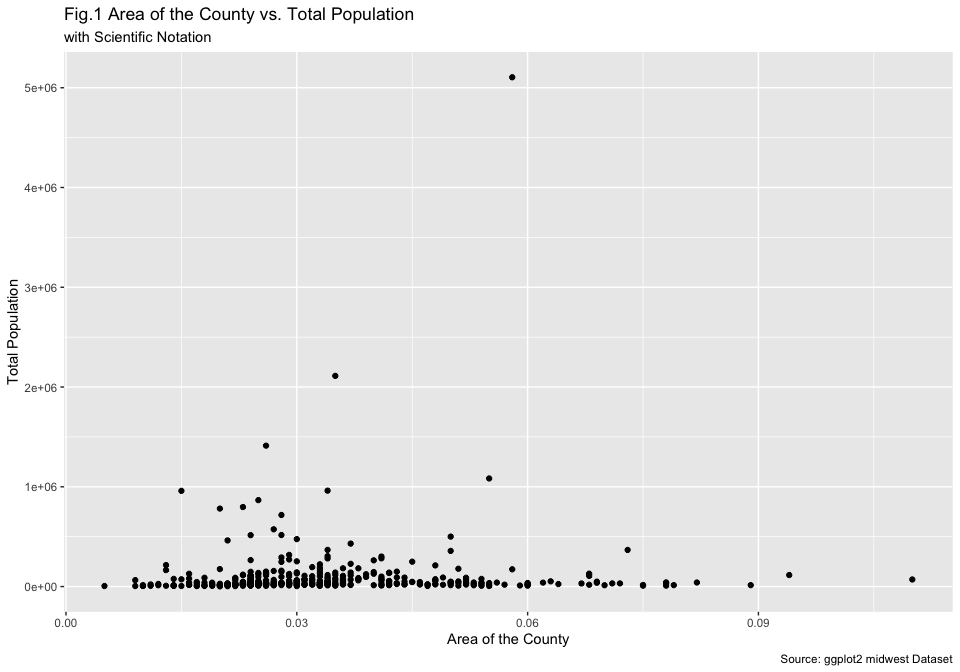
# Problem 1

In this problem we will use the *midwest* data in **ggplot2** package. First read about the data to understand the variables. In steps stated below, we will make a scatterplot between two numerical variables, then we will add more variables to the plot using aesthetics like color and size, and enhance plot in various ways. Do the following parts:

## Part A [5 Points]

Make a scatterplot of area (x-axis) versus total population (y-axis). Label axes and add title. What do you notice?

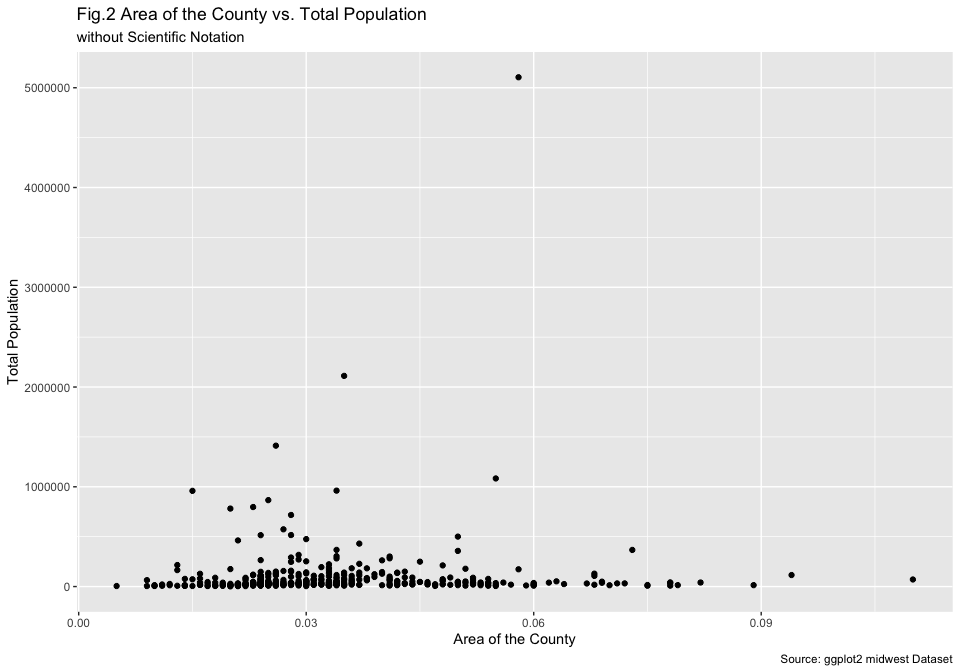
area.population.plot = ggplot(data = midwest,  
 aes(x = area,   
 y = poptotal)) +  
 geom\_point() +   
 xlab("Area of the County") +   
 ylab("Total Population") +  
 labs(title = "Fig.1 Area of the County vs. Total Population",  
 subtitle = "with Scientific Notation",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.plot



In this initial graph the y-axis values are in scientific notation. R automatically simplifies them into scientific notation as the full numbers take significantly more space in the plot. I can also see there are some obvious outliers and most of the data-points have clustered in the bottom-side of the graph.

Use options(scipen=999) to turn off scientific notation like 1e+06 and redo the plot. Add a chart number (Fig.1 etc.) with title and subtitle to your plot.

options(scipen = 999)  
  
area.population.plot = area.population.plot +  
 labs(title = "Fig.2 Area of the County vs. Total Population",  
 subtitle = "without Scientific Notation",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.plot



When I used options(scipen = 999), the y-axis values changed to normal base-10 numbers.

## Part B [5 Points]

Identify all places (state and county) with total population above 1000000 and report them.

We can achieve this by using the base-R and also the dplyr package. I am going to be showing both versions.

filter(midwest[, c("county", "state", "poptotal")], midwest$poptotal > 1000000)

## # A tibble: 4 × 3  
## county state poptotal  
## <chr> <chr> <int>  
## 1 COOK IL 5105067  
## 2 OAKLAND MI 1083592  
## 3 WAYNE MI 2111687  
## 4 CUYAHOGA OH 1412140

midwest[midwest$poptotal > 1000000, c("county", "state", "poptotal")]

## # A tibble: 4 × 3  
## county state poptotal  
## <chr> <chr> <int>  
## 1 COOK IL 5105067  
## 2 OAKLAND MI 1083592  
## 3 WAYNE MI 2111687  
## 4 CUYAHOGA OH 1412140

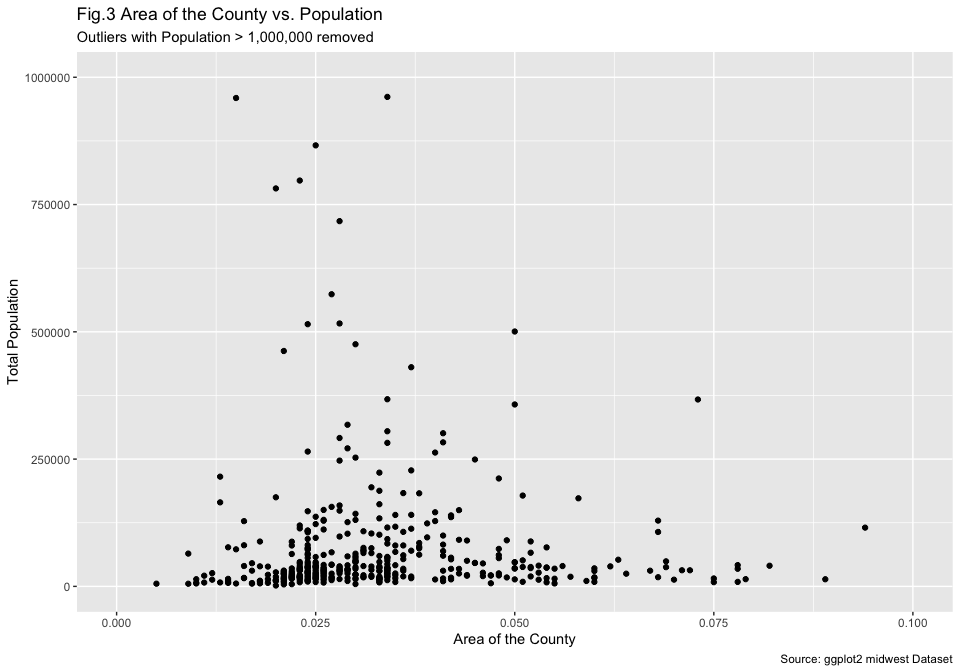
in both cases, we can see that Illinois Cook, Minnesota Oakland & Wayne, and Ohio Cuyahoga has a total population above 1,000,000.

## Part C [7.5 Points]

Delete the above outlier and redo the scatterplot. You can use any way to remove this, here are some options:

* Change x and y axes limits for the plot using two additional settings:
  + xlim(c(0, 0.1))
  + ylim(c(0, 1000000))
* use function coord\_cartesian() with same limits as above.
* create a subset of the Midwest data with population total less than or equal to 1000000 and then redo the scatter plot.

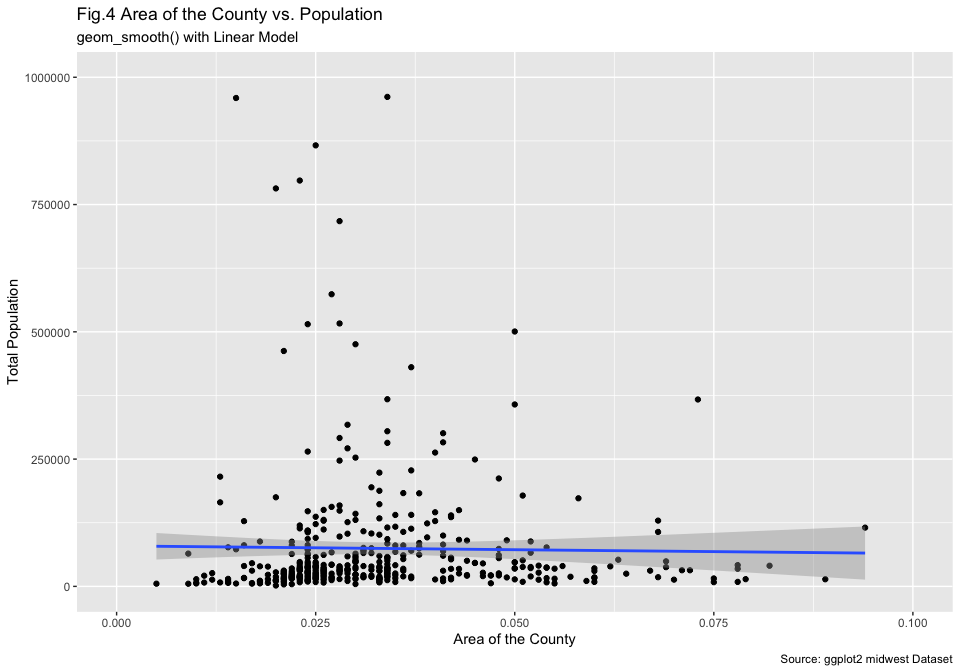
area.population.plot = area.population.plot +   
 xlim(c(0, 0.1)) +  
 ylim(c(0, 1000000)) +   
 labs(title = "Fig.3 Area of the County vs. Population",   
 subtitle = "Outliers with Population > 1,000,000 removed",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.plot



## Part D [5 Points]

Using geom\_smooth() add linear regression model. What can you say about the relation between population and area?

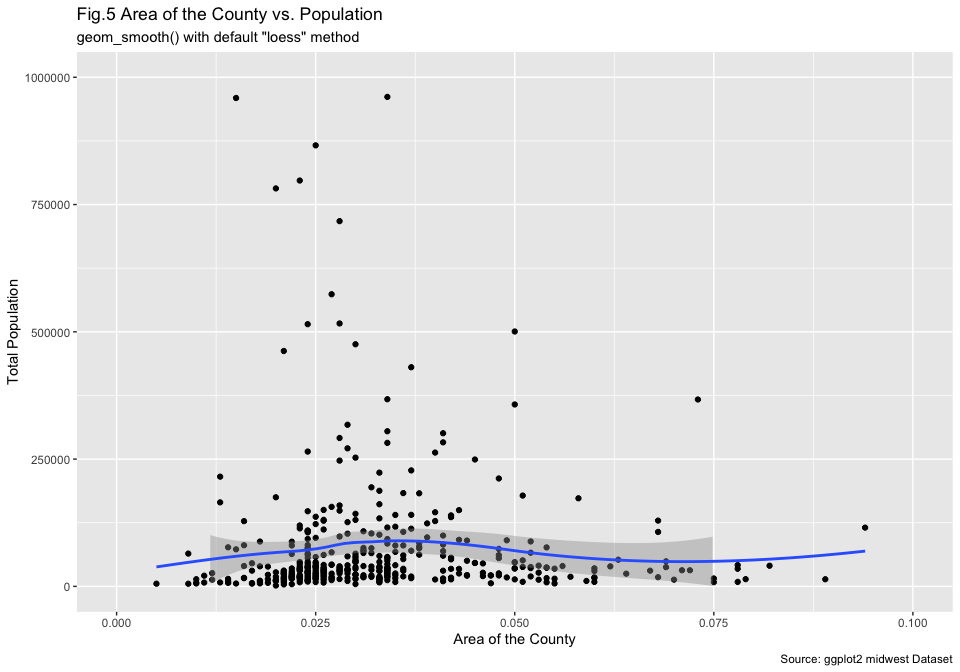
area.population.plot.lm = area.population.plot +   
 geom\_smooth(method = "lm") +   
 labs(title = "Fig.4 Area of the County vs. Population",  
 subtitle = "geom\_smooth() with Linear Model",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.plot.lm



From the linear regression line, we can see that there is weak or nearly no correlation at all between the area of the county and the total population as the best=fit line is nearly flat. At the same time, there is high variability in the relatively small countries as they have both high and low populations. Even though we removed the countries with population larger than 1,000,000, there are still some obvious outliers in the data. Lastly, we can also see that, as the county gets larger, it is tend to have a smaller population, we can see a slight triangle covering the data points which shows us this.

Using geom\_smooth() with R’s default method ”loess” add a fitted non-linear curve instead of lm.

area.population.plot.loess = area.population.plot +   
 geom\_smooth(method = "loess") +   
 labs(title = "Fig.5 Area of the County vs. Population",  
 subtitle = "geom\_smooth() with default \"loess\" method",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.plot.loess



Read help file for the geom\_smooth() to explain this ”loess” and tell what do you observe is different in lm and R’s default.

When I checked the help file and recalling my previous experiences, I can say that geom\_smooth adds a fitted line to the plots to show the relationship between my variables of interest. By default, if nothing is specified in the parameters, geom\_smooth uses stats::loess() as the smoothing method for samples less than 1,000 observations, or mgcv:gam() otherwise.

When I checked the documentation for stats::loess(), I saw that it creates the best-fit-line by performing **local** polynomial regression, determining the fit line using the nearby data points.

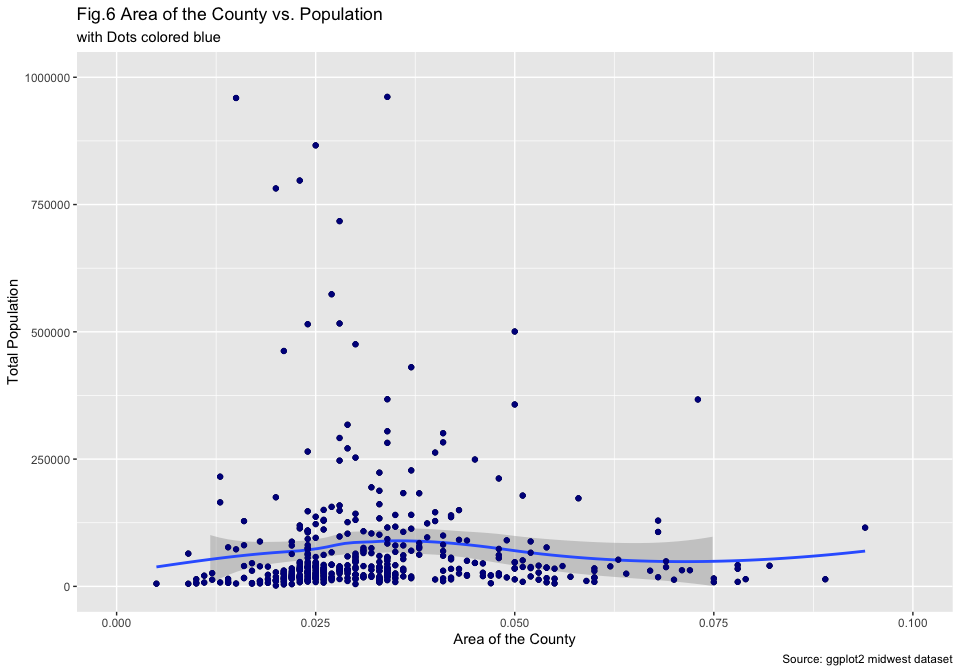
We can say that, in general, stats::loess() or the default method of geom\_smooth() fits a smooth curve that can show the non-linear patterns in the data. On the other hand, method = "lm" fits a linear model in the data which is a straight line, representing the best **linear** fit.

In a case where our data has a linear relationship, both methods may show a similar result. On the other hand, in a non-linear relationship, the stats::loess() can fit a smoother curve which can better show the patterns in the data. Even in our example -which is mostly linear- the curved version has a slight bump in the relatively small counties, taking the outliers into account, compared to the nearly straight line.

## Part E [2.5 Points]

Change color of the dots in the above plot.

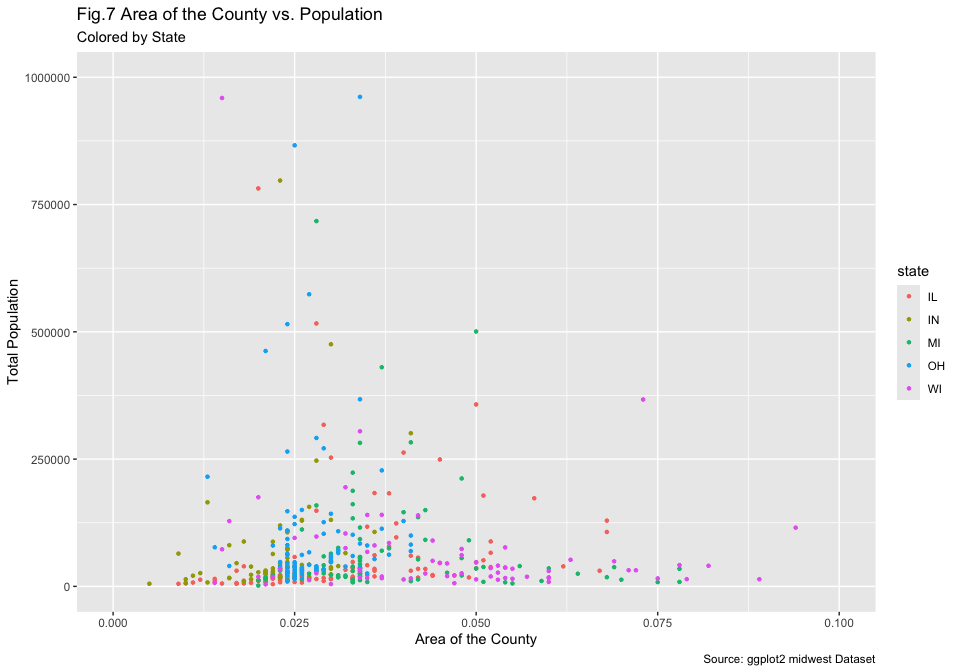
area.population.colored = area.population.plot.loess +   
 geom\_point(color = "darkblue") +  
 labs(title = "Fig.6 Area of the County vs. Population",  
 subtitle = "with Dots colored blue",  
 caption = "Source: ggplot2 midwest dataset")  
  
area.population.colored



## Part F [7.5 Points]

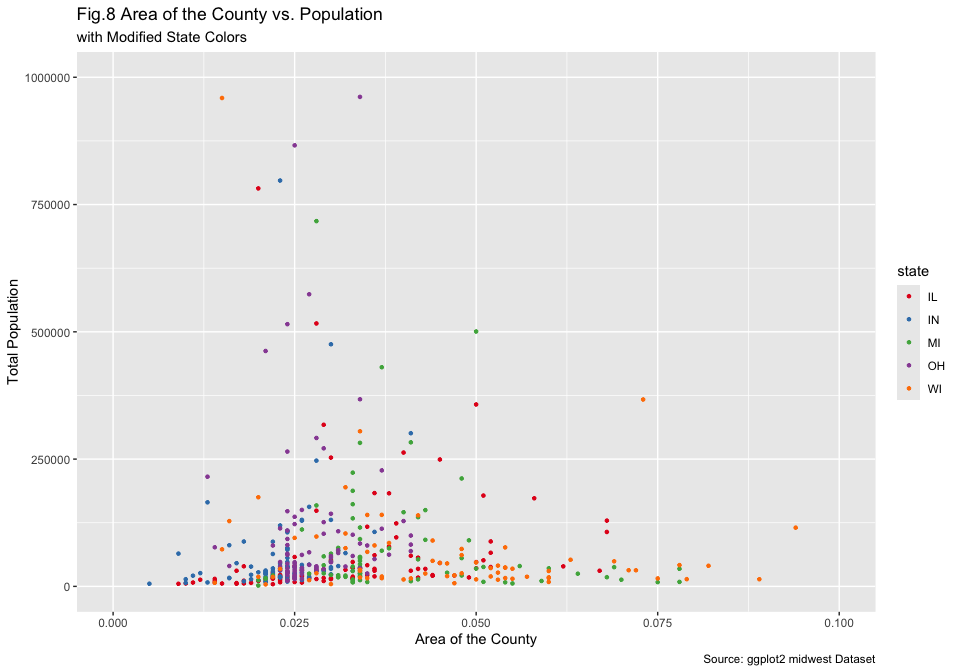
Change color of the dots based on the state (categorical) variable.

area.population.state = ggplot(data = midwest,  
 aes(x = area,  
 y = poptotal,  
 color = state)) +  
 geom\_point(size = 0.9) +  
 xlim(c(0, 0.1)) +  
 ylim(c(0, 1000000)) +   
 xlab("Area of the County") +   
 ylab("Total Population") +   
 labs(title = "Fig.7 Area of the County vs. Population",  
 subtitle = "Colored by State",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.state



Show how to change the color of the state dots from the R’s default choice in above plot.

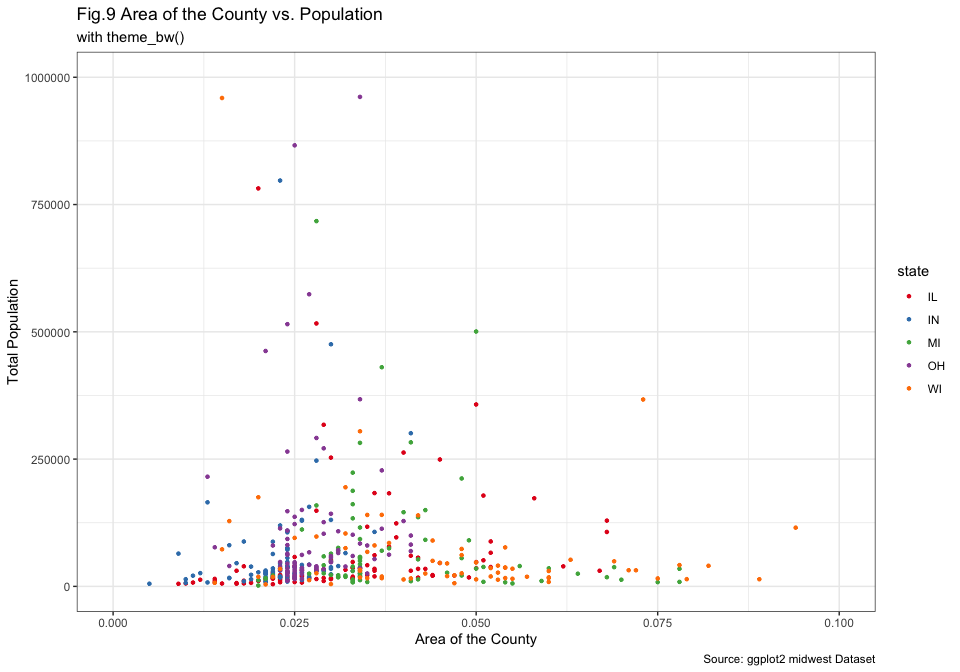
area.population.state.specific = area.population.state +   
 scale\_color\_manual(values = c("#E41A1C", "#377EB8", "#4DAF4A", "#984EA3", "#FF7F00")) +   
 labs(title = "Fig.8 Area of the County vs. Population",  
 subtitle = "with Modified State Colors",  
 caption = "Source: ggplot2 midwest Dataset")  
  
area.population.state.specific



## Part G [5 Points]

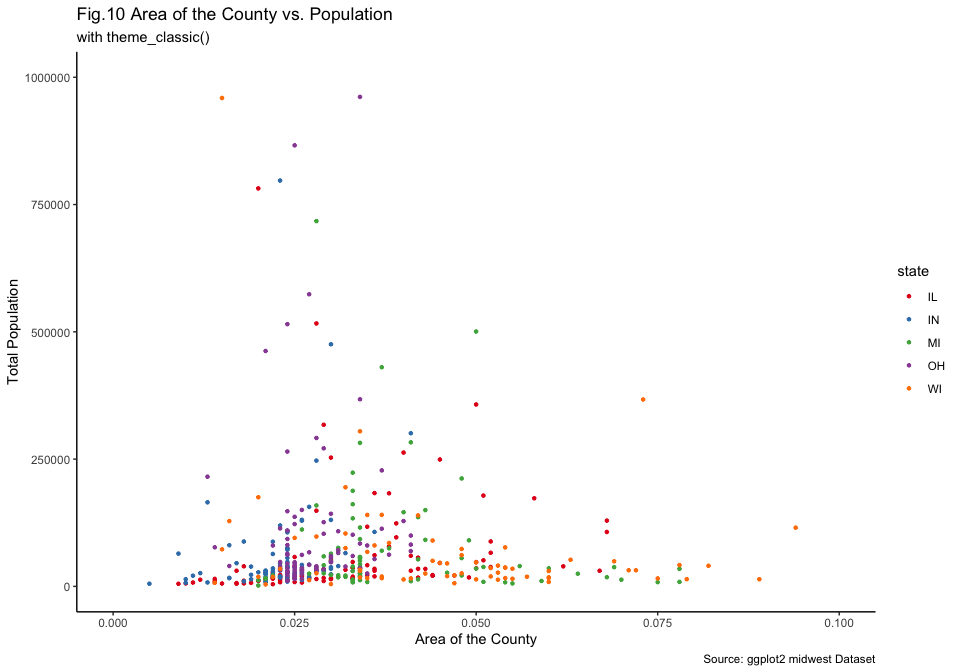
Use theme\_bw() to change the theme of the above (Fig.8) plot.

apss\_bw = area.population.state.specific +   
 theme\_bw() +  
 labs(title = "Fig.9 Area of the County vs. Population",  
 subtitle = "with theme\_bw()",  
 caption = "Source: ggplot2 midwest Dataset")  
  
apss\_bw



Use theme\_classic() to change the theme of the above (Fig.8) plot.

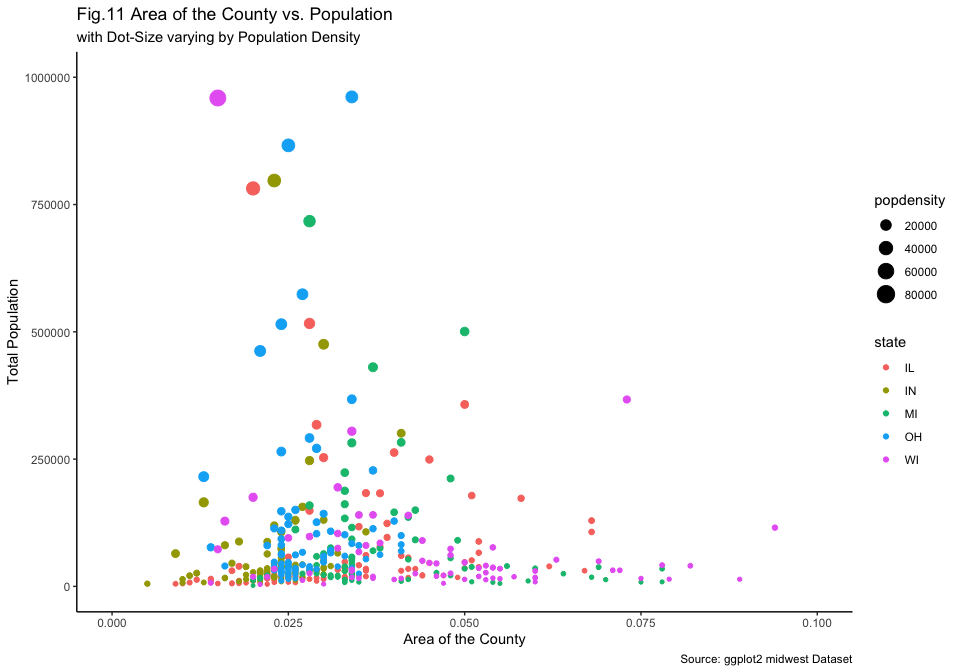
apss\_classic = area.population.state.specific +   
 theme\_classic() +  
 labs(title = "Fig.10 Area of the County vs. Population",  
 subtitle = "with theme\_classic()",  
 caption = "Source: ggplot2 midwest Dataset")  
  
apss\_classic



## Part H [7.5 Points]

Have the dot size vary by popdensity (continuous) variable in above plot. What do you observe?

apss.density = ggplot(data = midwest,   
 aes(x = area,   
 y = poptotal,   
 color = state,   
 size = popdensity)) +  
 geom\_point() +   
 xlim(c(0, 0.1)) +  
 ylim(c(0, 1000000)) +   
 theme\_classic() +   
 xlab("Area of the County") +   
 ylab("Total Population") +   
 labs(title = "Fig.11 Area of the County vs. Population",  
 subtitle = "with Dot-Size varying by Population Density",  
 caption = "Source: ggplot2 midwest Dataset")  
  
apss.density



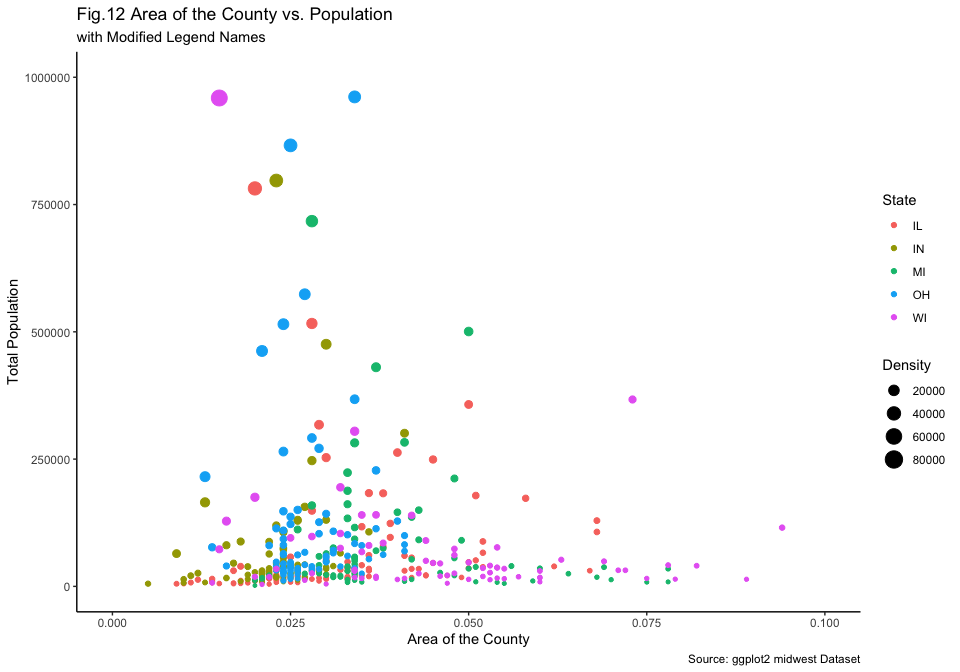
From the graph, we can see that there is a higher population density in larger counties as the large dots are mostly clustered on the relatively small counties. As expected, we can see that, counties with small areas and large populations have higher population densities.

At the same time, we can clearly see that some states have relatively small counties as they do not appear on the right-side of the graph. For example Indiana, Ohio, and -slightly- Illinois. We can see that Wisconsin has a high variability in both county size and population as it spans across both Axis and have varying dot sizes.

## Part I [5 Points]

Modify legend for state and popdensity to States and Density respectively.

apss.density = apss.density +  
 labs(title = "Fig.12 Area of the County vs. Population",  
 subtitle = "with Modified Legend Names",  
 caption = "Source: ggplot2 midwest Dataset",  
 size = "Density",  
 color = "State")  
  
apss.density



*Note: I do not know why but when I set the legend names to State and Density, it flips the order of the legends. I could not figure out the reason for this.*

## Part J [7.5 Points]

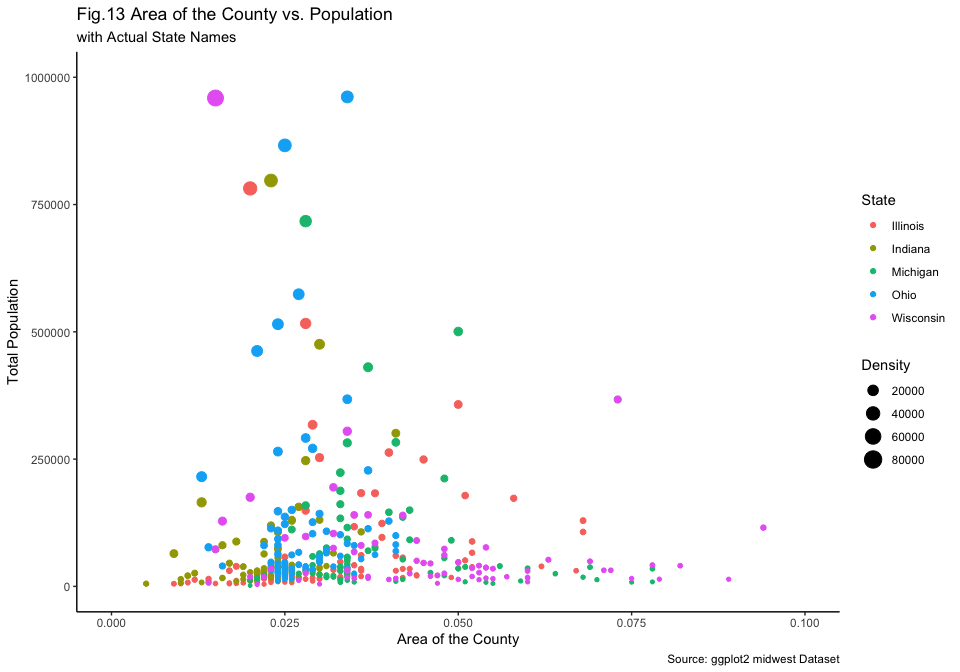
Change state names to actual names of the state, replace IL with Illinois, IN with Indiana and so on.

To do this, we can first convert the state variable into a factor

midwest$state = factor(midwest$state,  
 levels = c("IL", "IN", "MI", "OH", "WI"),  
 labels = c("Illinois", "Indiana", "Michigan", "Ohio", "Wisconsin"))

Following this, we can plot the data as usual.

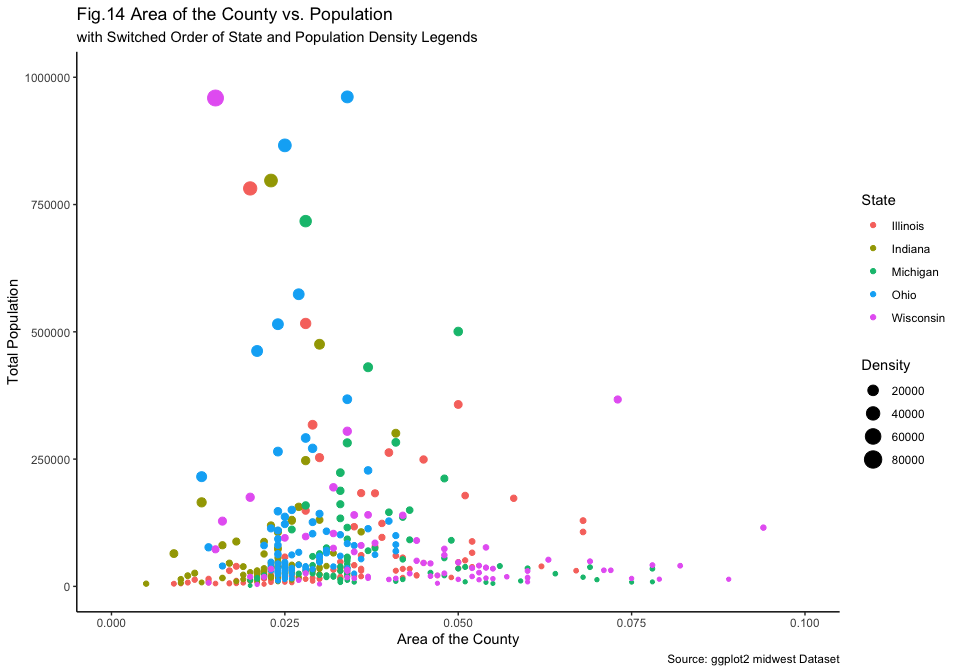
apss.density.named = ggplot(data = midwest,   
 aes(x = area,   
 y = poptotal,   
 color = state,   
 size = popdensity)) +  
 geom\_point() +   
 xlim(c(0, 0.1)) +  
 ylim(c(0, 1000000)) +   
 theme\_classic() +   
 xlab("Area of the County") +   
 ylab("Total Population") +   
 labs(title = "Fig.13 Area of the County vs. Population",  
 subtitle = "with Actual State Names",  
 caption = "Source: ggplot2 midwest Dataset",  
 size = "Density",  
 color = "State")  
  
apss.density.named



## Part K [7.5 Points]

Switch the order of legend for state and popdensity using guides() function.

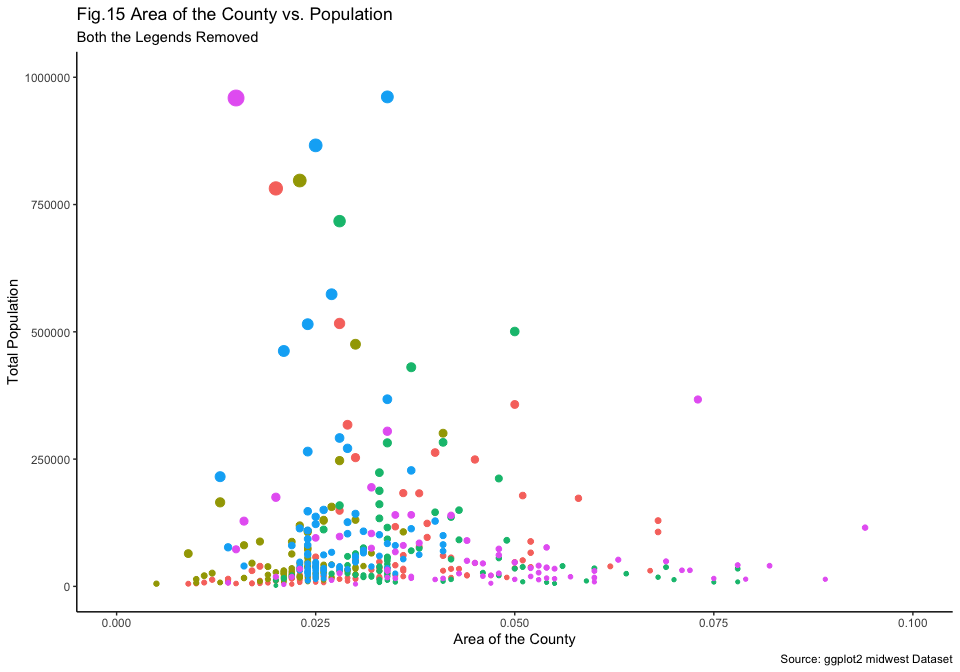
apss.density.ordered = apss.density.named +   
 guides(color = guide\_legend(order = 1),  
 size = guide\_legend(order = 2)) +   
 labs(title = "Fig.14 Area of the County vs. Population",  
 subtitle = "with Switched Order of State and Population Density Legends",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density")  
  
apss.density.ordered



## Part L [5 Points]

Remove legend from the plot.

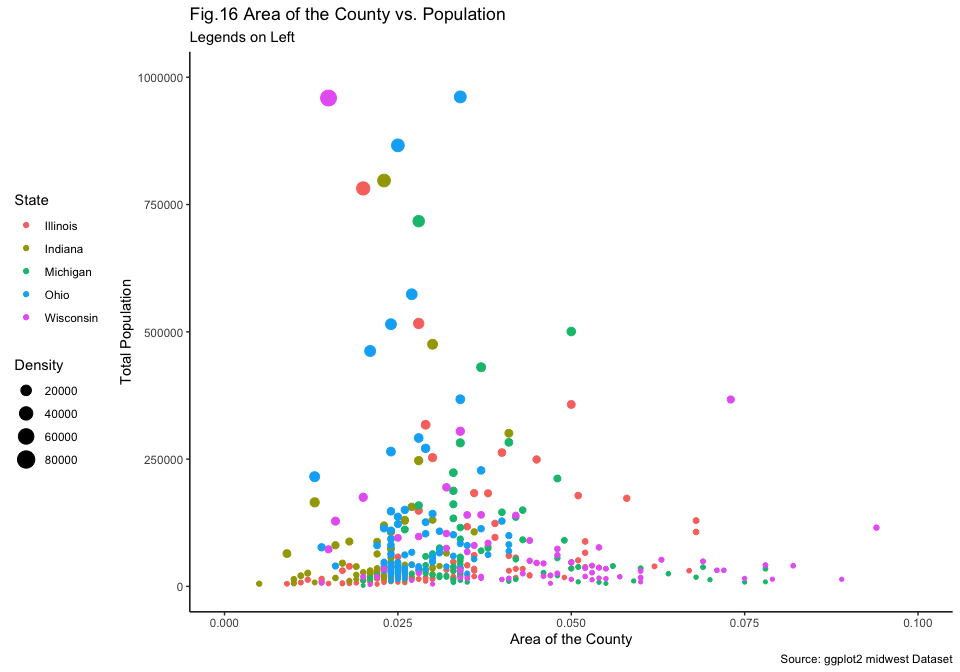
apss.density.removed = apss.density.named +   
 labs(title = "Fig.15 Area of the County vs. Population",  
 subtitle = "Both the Legends Removed",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density") +   
 theme(legend.position = "none")  
  
apss.density.removed



## Part M [5 Points]

Make legend move to the left side.

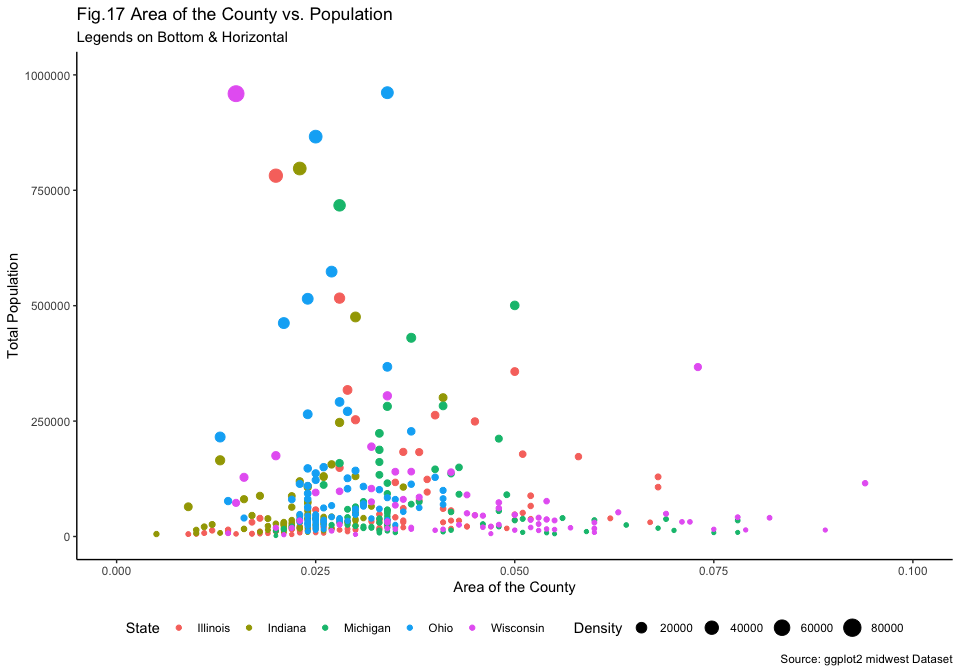
apss.density.left = apss.density.named +   
 labs(title = "Fig.16 Area of the County vs. Population",  
 subtitle = "Legends on Left",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density") +   
 theme(legend.position = "left")  
  
apss.density.left



## Part N [5 Points]

Make legend move to the bottom and horizontal.

apss.density.bottom.h = apss.density.named +   
 theme(legend.position = "bottom",  
 legend.direction = "horizontal") +   
 labs(title = "Fig.17 Area of the County vs. Population",  
 subtitle = "Legends on Bottom & Horizontal",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density")  
  
apss.density.bottom.h



## Part O [7.5 Points]

Filter subset of data with poptotal values > 300,000 and call this dataframe as midwest\_sub. Report how many counties satisfy this criterion.

midwest\_sub = filter(midwest, poptotal > 300000)  
nrow(midwest\_sub)

## [1] 23

There are 23 counties that have a population larger than 300,000.

## Part P [5 Points]

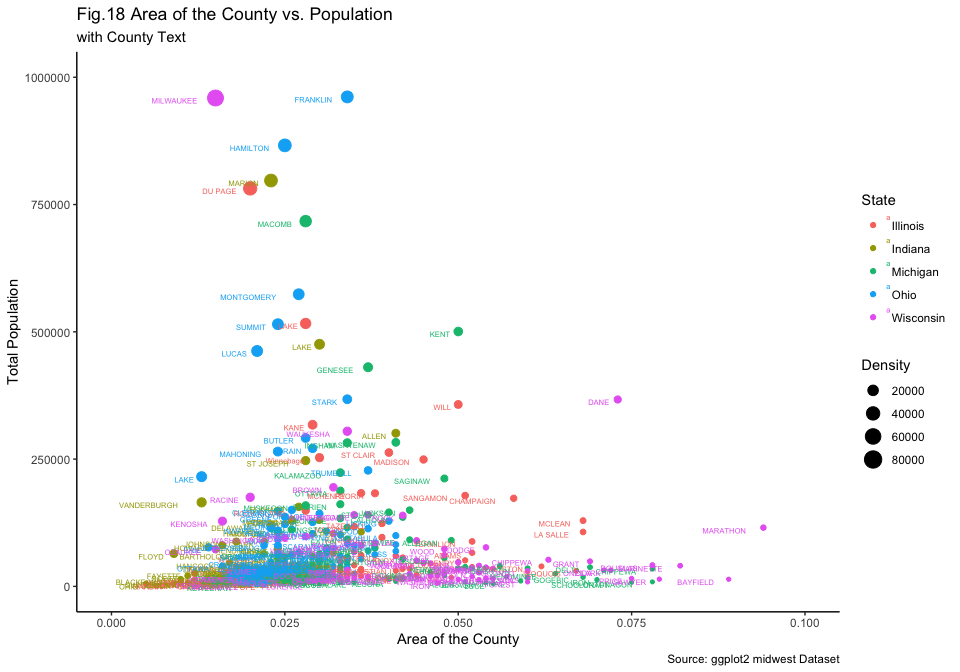
Create a variable in midwest\_sub data for large county which satisfies poptotal>300,000 as follows:

midwest\_sub$large\_county <- ifelse(midwest\_sub$poptotal > 300000, midwest\_sub$county, "")

## Part Q [7.5 Points]

In the scatter plot of area versus population, with dots colored for states and dots varying in size for popdensity, add text to highlight the identifies counties using function geom\_text().

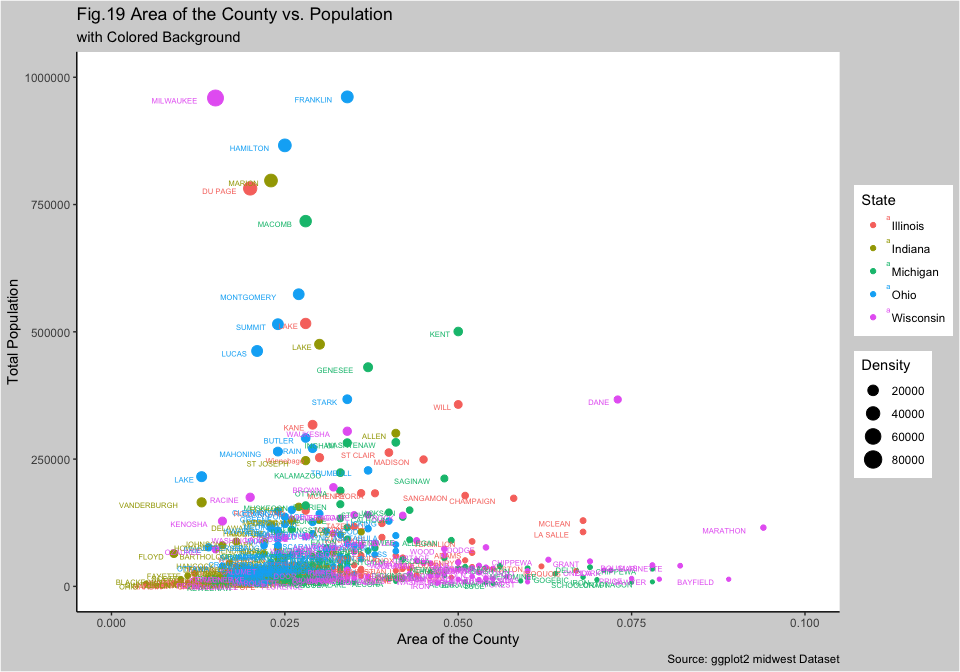
apss.text = apss.density.named +   
 geom\_text(aes(label = county),  
 hjust = 1.4,  
 vjust = 1,  
 size = 2) +   
 labs(title = "Fig.18 Area of the County vs. Population",  
 subtitle = "with County Text",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density")  
  
apss.text



## Part R [5 Points]

Change the background color of your plot.

apss.colored = apss.text +   
 theme(plot.background = element\_rect(fill = "lightgrey")) +   
 labs(title = "Fig.19 Area of the County vs. Population",  
 subtitle = "with Colored Background",  
 caption = "Source: ggplot2 midwest Dataset",  
 color = "State",  
 size = "Density")  
  
apss.colored



# Problem 2 [50 points]

Write a summary of your project, including information about:

* **Introduction:** Tell us what problem of interest. Why is this problem interesting and important? Introduce recent research done in area related to your problem. You can pack all this together to motivate us. Do keep it short, to the point, and interesting.
* **Data:** Tell us about the data resource and explain dimensions of the data, variables in the data, and how does this data relate to your research questions.
* **Initial findings:** Perform exploratory data analysis (EDA) including summaries and data visualizations for one research goal. Show main plot(s) and findings. Make sure to add labels, titles etc. to make your tables and graphs informative.
* share one advanced data visualization with any required information on how this plot is useful for the study and what it tells about your project (interpretation). Show your R skills, creativity, and advanced work in R.

Note: This is an initial report.

*Note: I restarted the figure numbers as this is a separate part.*

# Introduction

Traffic stops are a regular part of our lives, in fact, more than 20 million Americans are stopped each year in the traffic (Pierson et al., 2020). Traffic stops are one of the most common ways of public-police interaction. As police officers conduct these stops, the decision-making process comes down to human judgment, which certainly comes with a certain type of bias. There have been many research projects that focuses on possible bias factors in traffic stops. Most of these studies found that the race of the driver is an important factor influencing the likelihood of being stopped and the outcome of the stop.

According to Pierson et al. (2020) Black and Hispanic drivers are stopped and searched more often than White drivers. However, Black drivers are less likely to be stopped after sunset -compared to the rate of being stopped during the day- when the face of the driver is less visible. It is also pointed out that, the bar to search Black and Hispanic drivers is generally lower than White drivers. Lastly, the study also concludes that the success rates of searches is lower for Hispanic drivers compared to White and Black drivers who has comparable hit rates. Similarly Xu et al. (2024) points out that Black drivers are stopped at higher rates compared to their proportion in the traffic.

In my data-analysis project, I am planning to focus on the demographical analysis of the traffic stops conducted in San Francisco between 2007 and 2016. I already found a research project conducted by *San Francisco Bay Area Planning and Urban Research Association (SPUR)* on San Francisco traffic stop data which only covers the 2019 data. San Francisco Bay Area Planning and Urban Research Association (2023) also has similar findings of Pierson et al. (2020), as it shows Black and Hispanic drivers are stopped more than their share in the population while Black drivers have significantly lower citation rate compared to White and Hispanic drivers. In fact, according to the SPUR study, more than half of the Black drivers who are stopped do not end up with citation at all. At the same time, different than the previous studies I was able to find, I am planning to focus on the locational aspects of the traffic stops and analyze the relationship between the time, outcome, demographics, and the location of the stop.

I can list my possible research questions for my traffic stops data of San Francisco between 2007 and 2016 as the following:

1. What is the relationship between being stopped and the general demographics?
   * How does this relationship change when we split the data into separate times of the day?
2. How does the outcome of the traffic stop (warning, citation, search, arrest) relate to the racial demographics of the driver?
3. How does the amount of drivers stopped vary by time of the day and day of the year?
4. Are certain parts of SF have higher traffic-stop rates?
   * How does this relationship look like if we take race into account as well.
   * How does the outcome of the stop relates to the location

# San Francisco Traffic Stops & Race Distribution Datasets

In my research project I am using the San Francisco police stop data which is a part of the Stanford Open Policing Project. The Open Policing Project is an ongoing project, that collects and organizes law enforcement stop data from different counties across the USA. San Francisco data includes details of 905,070 vehicle stops from January 1st, 2007 to June 30th, 2016. The dataset has 22 different variables on different detail of the stop such as date/time/location of the stop, age/race/sex of the driver that is stopped, and the outcome of the stop.

It is important to note that, I will use data from January 1st, 2007 to December 31st, 2015, inclusive. This is crucial as having 2016 until June can skew my data and lead me to incorrect conclusions. At the same time, some of the variables like the location/district/open address, raw data fields won’t be used anywhere in my research project. The final dataset I will use (without the extra variables and half of 2016) will have 864,722 stops and 17 variables.

The variables in this dataset are helpful for my research questions as they provide important details about the demographic information of each traffic stop. At the same time, I am able to access information on what happened during the stop and how did the stop resulted. In the EDA and further parts of my project, these variables will provide me with much flexibility and potential to explore more.

At the same time, as a helper dataset, I am using a dataset of the proportion of races in the San Francisco population. This dataset is taken from the US Census data and contains the percentage of each racial group in the population for each year from 2007 to 2016. It is a very simple dataset with 3 variables: Year, Subject Race and Population Percentage and 50 rows (10 years x 5 different race groups). This dataset will help me to proportionate my data for different race groups which will allow me to better explore and understand my data.

# Initial Findings

## Data Preparation Steps

### Reading the Data

I am using two datasets. The first dataset, I explained in detail above, is from the Stanford Open Policing Project and contains the stops from 2007 to 2016. On the other hand, the other dataset is from US Census data and contains the proportions of each race from 2007 to 2016. This is crucial to make a better analysis of the data as the races are not equally distributed. I created this dataset manually by copying the values from a [3rd Party Website](https://www.indexmundi.com/facts/united-states/quick-facts/california/county/san-francisco/population#table), which took the values from the US Census Data.

traffic.data = read.csv("ca\_sf\_vehicle\_2007\_2016.csv")  
race.data = read.csv("population\_race\_data.csv")

We have to convert some variables to factors, date/time objects in order to use them proficiently during plotting.

traffic.data$subject\_sex = factor(traffic.data$subject\_sex,  
 levels = c("male", "female"),  
 labels = c("Male", "Female"))  
  
traffic.data$subject\_race = factor(traffic.data$subject\_race,  
 levels = c("asian/pacific islander",  
 "black",  
 "hispanic",  
 "white",  
 "other"),  
 labels = c("Asian/Pacific Islander",  
 "Black",  
 "Hispanic",  
 "White",  
 "Other"))  
  
traffic.data$date = as.Date(traffic.data$date)  
traffic.data$time = strptime(traffic.data$time, format="%H:%M:%S")  
  
traffic.data$outcome[is.na(traffic.data$outcome)] = "No Action"  
  
traffic.data$outcome = factor(traffic.data$outcome,  
 levels = c("citation",  
 "warning",  
 "arrest",  
 "No Action"),  
 labels = c("Citation",  
 "Warning",  
 "Arrest",  
 "No Action"))

### Merging Traffic Data and Race Data

To have the proportion of each race in our data, I will merge two data-frames here using the merge function. The subject\_race and Year columns are common between these two datasets.

traffic.data$Year = format(traffic.data$date, "%Y")  
  
traffic.data = merge(traffic.data,  
 race.data,  
 by = c("subject\_race", "Year"),  
 all.x = TRUE)  
  
colnames(traffic.data)[ncol(traffic.data)] = "proportioned\_value"  
  
traffic.data$Year = NULL

Now we can look at the traffic.data dataframe to see the new version of the dataframe

traffic.data[sample(nrow(traffic.data), 10), c("subject\_race", "proportioned\_value", "date")]

## subject\_race proportioned\_value date  
## 242095 Black 0.060 2012-08-06  
## 664352 White 0.421 2010-10-15  
## 408624 Other 0.033 2007-09-23  
## 577538 White 0.452 2008-09-17  
## 805834 White 0.415 2014-03-28  
## 813653 White 0.415 2014-06-22  
## 323559 Hispanic 0.144 2009-06-25  
## 801236 White 0.415 2014-04-29  
## 318048 Hispanic 0.142 2008-10-23  
## 583034 White 0.452 2008-11-05

We can see that we have a new value called proportioned\_value. They represent the proportion of individuals from that race in that specific year. For example,

traffic.data[99999, c("subject\_race", "proportioned\_value", "date")]

## subject\_race proportioned\_value date  
## 99999 Asian/Pacific Islander 0.346 2012-03-16

Asian/Pacific Islanders represented 34.6% of the population in 2012. As I showed in Figure 2 below, these proportions change over the years so we cannot have a global number for each year.

### Normalizing the Proportions

While these proportions are valid for each year and race, as we duplicate them for each individual traffic stop, they do not have a clear use for our plots. At the same time, all the years do not have th esame race distribution. We have to normalize these values so that we will have the weighted value of each stop.

As we are trying to show a **fair** analysis of data across the races, we have to take the multiplicative inverse of the weights. This ensures that groups with a smaller share in the population is equally represented.

Now the issue is we have bunch of weights but we did not normalize them in any way such that they have a fixed sum. We can ensure this by dividing each weight by the sum of all the weights in the dataset.

traffic.data$generalWeights = (1 / (traffic.data$proportioned\_value)) / sum(1/(traffic.data$proportioned\_value))  
sum(traffic.data$generalWeights)

## [1] 1

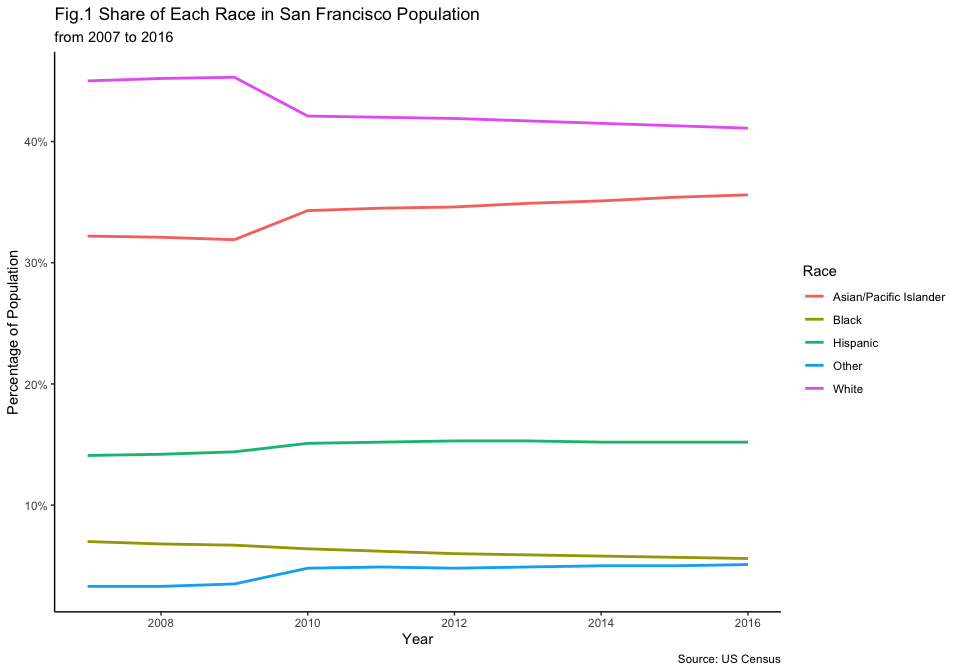
We can see that the weights has a sum of 1 which shows that they have now have a normalized value.

## Data Exploration

### Exploring the Race Distribution of the San Francisco Population from 2007 to 2016

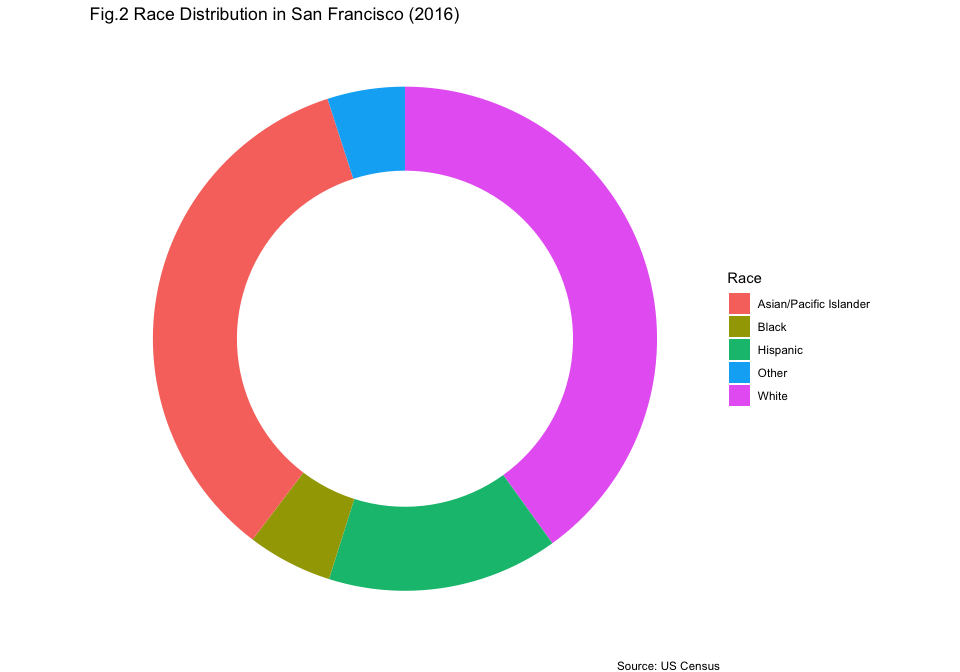
As I mentioned at the beginning of the previous section, the distribution of different races are not equal. To show their progression over time, we can look at their line graph.

race.by.year = aggregate(race.data$Population\_Percentage,  
 by = list(race.data$subject\_race,  
 race.data$Year),  
 FUN = sum)  
  
colnames(race.by.year) = c("subject\_race", "Year", "Population\_Percentage")  
  
race.line.plot = ggplot(data = race.by.year,  
 aes(x = Year,  
 y = Population\_Percentage,  
 color = subject\_race)) +  
 geom\_line(size = 1) +  
 labs(title = "Fig.1 Share of Each Race in San Francisco Population",  
 subtitle = "from 2007 to 2016",  
 x = "Year",  
 y = "Percentage of Population",  
 color = "Race",  
 caption = "Source: US Census") +  
 scale\_y\_continuous(labels = scales::percent) +  
 theme\_classic()  
  
race.line.plot



This line graph shows us that, throughout the time-frame there had been a significant difference in the share of different races in San Francisco. We can also view the 2016 race distribution in a donut plot.

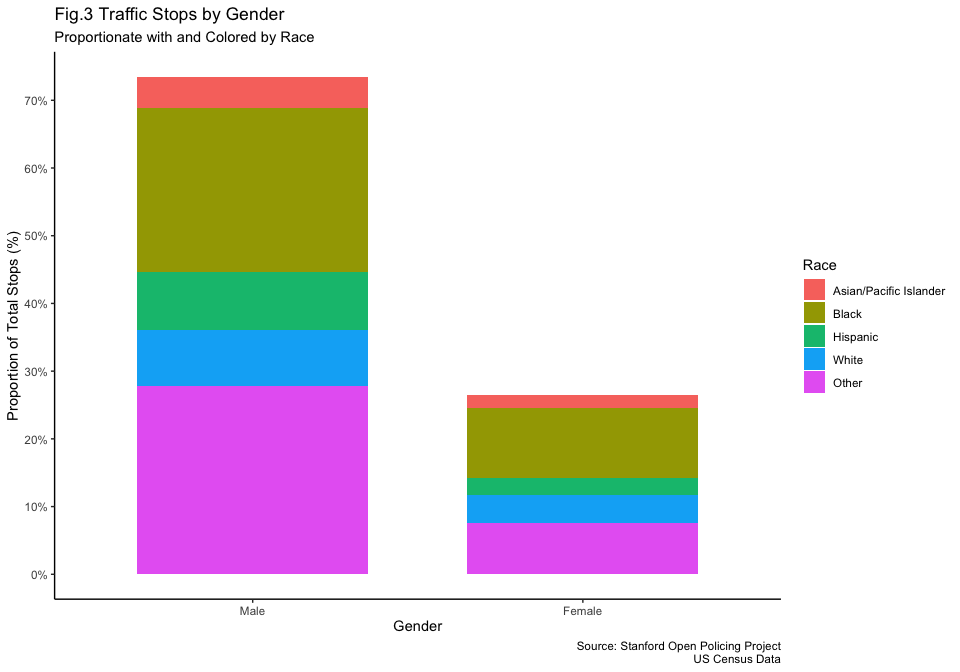
race.2016 = subset(race.data, Year == 2016)  
race.2016$fraction = race.2016$Population\_Percentage / sum(race.2016$Population\_Percentage)  
race.2016$ymax = cumsum(race.2016$fraction)  
race.2016$ymin = c(0, head(race.2016$ymax, n = -1))  
  
race.distribution.2016 = ggplot(race.2016,  
 aes(ymax = ymax,  
 ymin = ymin,  
 xmax = 4,  
 xmin = 3,  
 fill = subject\_race)) +  
 geom\_rect() +  
 coord\_polar(theta = "y") +  
 xlim(c(1, 4)) +  
 theme\_void() +  
 labs(title = "Fig.2 Race Distribution in San Francisco (2016)",  
 fill = "Race",  
 caption = "Source: US Census")   
  
race.distribution.2016



From Figure 2, we can see that the proportion of Black, Hispanic, and Other Race individuals in the San Francisco is significantly low while the large majority of individuals are White and Asian/Pacific Islander. This shows us the need for adjusting the data for race distribution as there is a significant gap between the proportion of races.

### Distribution of the Traffic Stops by Gender

group.race.sex.total = aggregate(traffic.data$generalWeights,  
 by = list(traffic.data$subject\_race,  
 traffic.data$subject\_sex),  
 FUN = sum)  
  
colnames(group.race.sex.total)= c("Race", "Gender", "Count")  
  
stops.sex.race.bplot = ggplot(data = group.race.sex.total, aes(x = Gender,  
 y = Count,  
 fill = Race)) +  
 geom\_bar(stat = "identity", width = 0.7) +  
 scale\_y\_continuous(labels = scales::percent\_format(scale = 100),  
 breaks = seq(0, 1, 0.1)) +  
 xlab("Gender") +  
 ylab("Proportion of Total Stops (%)") +  
 labs(title = "Fig.3 Traffic Stops by Gender",  
 subtitle = "Proportionate with and Colored by Race",  
 caption = "Source: Stanford Open Policing Project\nUS Census Data") +  
 theme\_classic()  
  
stops.sex.race.bplot



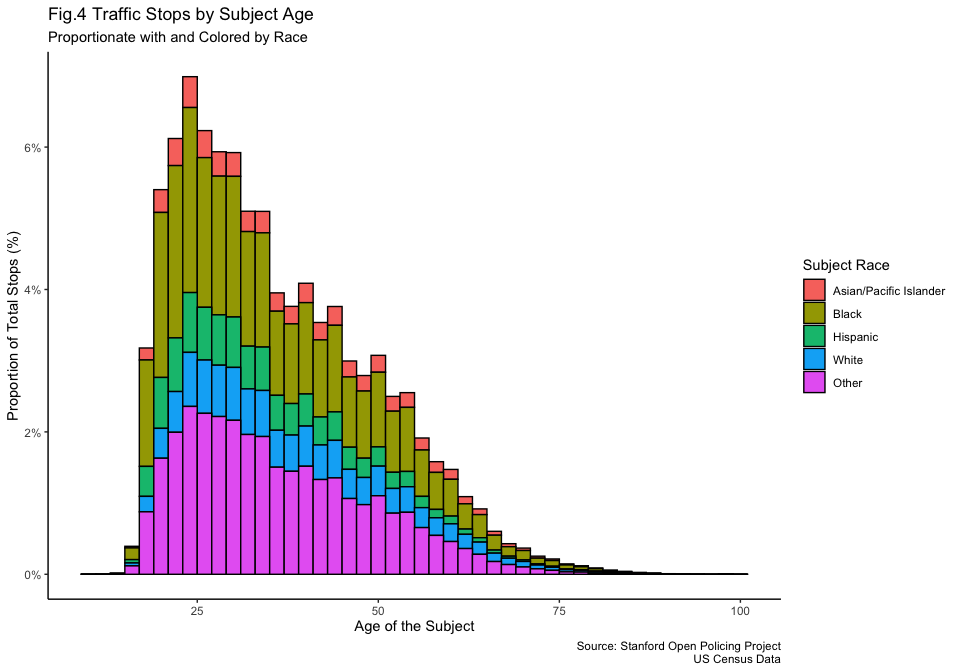
Before my interpretation of this graph, it is important to note that, the Y-Axis values do not represent the number of the stops anymore but they represent the proportion of the stops in all of the traffic stops.

We can still see that more than 70% of the stopped drivers were male compared to ~25% female drivers. As we adjusted our data by race, we can see that black drivers account for around 30% of the stops while covering less than 10% of the population. On the other hand, Asian/Pacific Islander drivers account for less than 10% of the stops while covering more than 30% of the traffic stops.

We can definitely confirm the findings of the previous studies where it was concluded that Black drivers were stopped significantly more than their share in the population.

### Distribution of Traffic Stops by Age

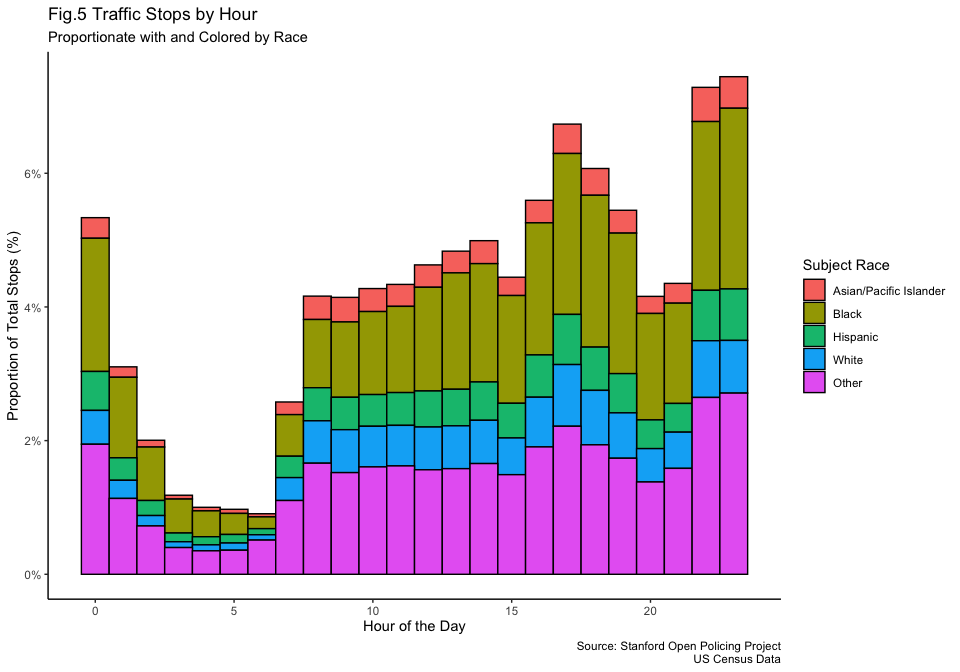
stops.age.hist = ggplot(traffic.data,  
 aes(x = subject\_age,  
 fill = subject\_race,  
 weight = generalWeights)) +  
 geom\_histogram(binwidth = 2, color = "black") +  
 scale\_y\_continuous(labels = scales::percent\_format(scale = 100)) +  
 xlab("Age of the Subject") +  
 ylab("Proportion of Total Stops (%)") +  
 labs(title = "Fig.4 Traffic Stops by Subject Age",  
 subtitle = "Proportionate with and Colored by Race",  
 caption = "Source: Stanford Open Policing Project\nUS Census Data",  
 fill = "Subject Race") +  
 theme\_classic()  
  
stops.age.hist



When we check the age distribution, we can again see the high share of black drivers across all the ages groups that were stopped. It is important to note that percentage of Hispanic drivers that are stopped are very similar to White drivers (while Hispanic drivers have a lower share in the population. Lastly, it can be seen that the age of the stopped driver do not follow a bell-curve distribution, and obviously right-skewed.

### Distribution of Traffic Stops by the Time of the Day

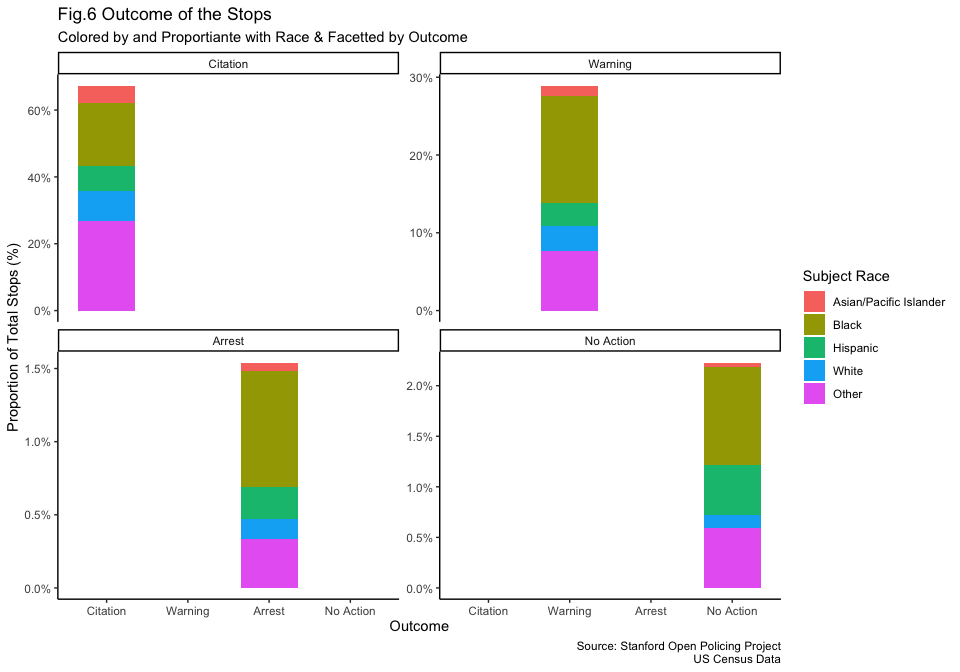
stops.time.hist = ggplot(traffic.data, aes(x = as.numeric(format(time, "%H")),  
 fill = subject\_race,  
 weight = generalWeights)) +  
 geom\_histogram(binwidth = 1, color = "black") +  
 scale\_y\_continuous(labels = scales::percent\_format(scale = 100)) +  
 xlab("Hour of the Day") +  
 ylab("Proportion of Total Stops (%)") +  
 labs(title = "Fig.5 Traffic Stops by Hour",  
 subtitle = "Proportionate with and Colored by Race",  
 caption = "Source: Stanford Open Policing Project\nUS Census Data",  
 fill = "Subject Race") +   
 theme\_classic()  
   
stops.time.hist



In Figure 5, it can be seen that after midnight, the amount of stops significantly decline and start to rise again at morning hours around 7AM. Interestingly, we cannot see that the proportion of Black drivers stopped declining after sunset which was claimed by Pierson et al. (2020). I will investigate this further in the upcoming parts of the project.

### Distribution of Traffic Stops by Outcome

stop.race.total = aggregate(traffic.data$generalWeights,  
 by = list(traffic.data$subject\_race,  
 traffic.data$outcome),  
 FUN = sum)  
  
colnames(stop.race.total)= c("Race", "Outcome", "Count")  
  
stops.outcome.bplot = ggplot(data = stop.race.total,  
 aes(x = Outcome,  
 y = Count,  
 fill = Race)) +  
 geom\_bar(stat = "identity", width = 0.7) +  
 scale\_y\_continuous(labels = scales::percent\_format(scale = 100)) +  
 xlab("Outcome") +  
 ylab("Proportion of Total Stops (%)") +  
 labs(title = "Fig.6 Outcome of the Stops",  
 subtitle = "Colored by and Proportiante with Race & Facetted by Outcome",  
 caption = "Source: Stanford Open Policing Project\nUS Census Data",  
 fill = "Subject Race") +  
 facet\_wrap(~ Outcome, scales = "free\_y") +   
 theme\_classic()  
  
stops.outcome.bplot



From the facets in Figure 6, we can see that most of the arrests resulted with citation followed by warning, no action and arrest. While citation has a race distribution relatively equal, the arrests were dominated by Black drivers. At the same time, while there are nearly no White drivers whose stop results with No Action, 1% of the total stops, which are all Black drivers, result in No action, which raises the concern about the legitimacy of the stops of Black drivers. In the upcoming parts of the project, I can examine the reason for stop variable and see the correlation between the driver race, reason and outcome of the traffic stops.

### Spatial Distribution of Traffic Stops in San Francisco

sf\_map = get\_stadiamap(bbox = c(left = -122.52,  
 bottom = 37.70,  
 right = -122.35,  
 top = 37.85),  
 zoom = 12,  
 maptype = "alidade\_smooth")  
  
stop.map = ggmap(sf\_map) +   
 geom\_density2d\_filled(data = traffic.data,  
 aes(x = lng, y = lat),  
 alpha = 0.6) +  
 scale\_fill\_viridis\_d(option = "rocket") +  
 labs(title = "Fig.7 Police Stops in San Francisco",  
 fill = "Stop Count",  
 caption = "Source: Stanford Open Policing Project") +  
 xlab("Longitude") +  
 ylab("Latitude") +   
 theme\_classic()  
  
stop.map



We can see that most of the traffic stops accumulated around the same part of the city and following some paths which can point to major roads or generally used paths, etc.

#### Examining Spatial Distribution by Race

stop.map.race = ggmap(sf\_map) +  
 geom\_density2d\_filled(data = traffic.data,  
 aes(x = lng, y = lat),  
 alpha = 0.6) +  
 scale\_fill\_viridis\_d(option = "rocket") +  
 labs(title = "Fig.8 Police Stops in San Francisco",  
 subtitle = "Facetted by Race",  
 fill = "Stop Count",  
 caption = "Source: Stanford Open Policing Project") +  
 xlab("Longitude") +  
 ylab("Latitude") +  
 facet\_wrap(~ subject\_race, nrow = 2) +  
 theme\_classic() +  
 theme(panel.spacing.x = unit(2, "lines"))  
  
stop.map.race



As an addition to the figure 7, I split it to facets of different races. Interestingly, we can see that traffic stops of Black drivers have accumulated in specific places in the city while Hispanic drivers’ traffic stops have followed a path (which can point to a major road). In the future parts of the project, I will examine the reasoning of this relationship as White, Asian and Hispanic drivers do not show this peak pattern and have more linear distribution around the map.

## Conclusion

In our analysis of the San Francisco traffic stops data from 2007 to 2016, we saw a clear racial bias. Black drivers were stopped for more than their share in the population while Asian/Pacific Islander drivers were stopped the least. Black drivers faced significantly high arrest rates, while significant amount of their stops resulted in no action which question the legitimacy of these stops.

Moving forward, I plan to examine the stop reasons, location-based biases and the general flow of stop (maybe with an alluvial plot). At the same time, I will look at how the stop rates change under different conditions like time of the year.

## Resources

* How to plot a line graph: [R-Graph Gallery](https://r-graph-gallery.com/line-plot.html)
* How to add spacing between facets: [StackOverflow Question](https://stackoverflow.com/questions/3681647/ggplot-how-to-increase-spacing-between-faceted-plots)
* Modifying tick frequency in axis: [Source](https://www.sthda.com/english/wiki/ggplot2-axis-ticks-a-guide-to-customize-tick-marks-and-labels)

## References

Pierson, E., Simoiu, C., Overgoor, J., Corbett-Davies, S., Jenson, D., Shoemaker, A., Ramachandran, V., Barghouty, P., Phillips, C., Shroff, R., & Goel, S. (2020). A large-scale analysis of racial disparities in police stops across the United States. *Nature Human Behaviour*, *4*(7), 736–745. <https://doi.org/10.1038/s41562-020-0858-1>

San Francisco Bay Area Planning and Urban Research Association. (2023). *Putting an end to biased traffic stops in san francisco*. <https://www.spur.org/news/2023-02-21/putting-end-biased-traffic-stops-san-francisco>

Xu, W., Smart, M., Tilahun, N., Askari, S., Dennis, Z., Li, H., & Levinson, D. (2024). The racial composition of road users, traffic citations, and police stops. *Proceedings of the National Academy of Sciences*, *121*(24). <https://doi.org/10.1073/pnas.2402547121>