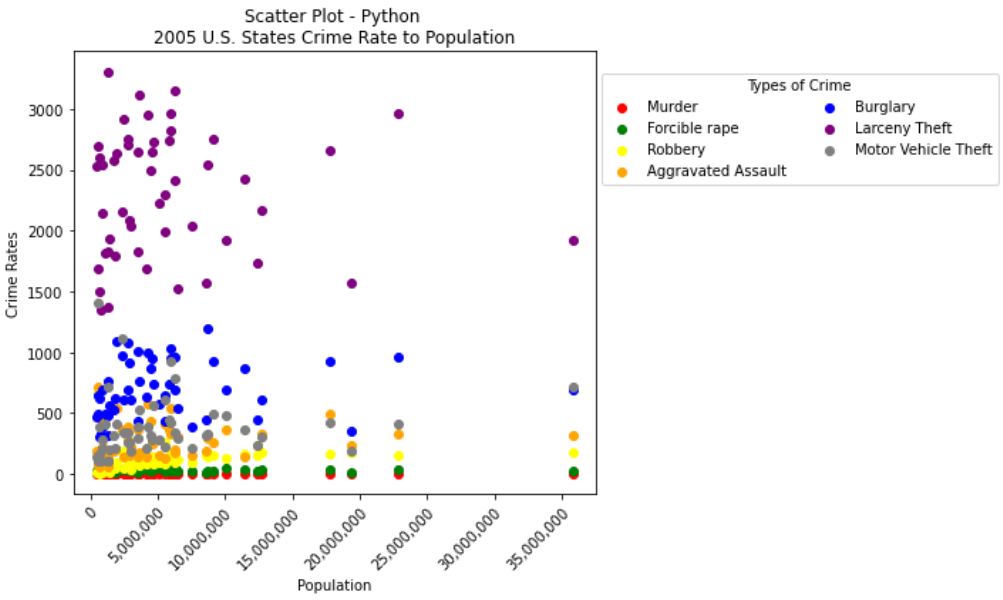
DSC 640: Weeks 7 – 8
Author: Kimberly Cable
Date: Oct 22, 2022
Exercise 4.2 Scatterplots, Bubble Charts, & Density Plots/Maps

Scatterplot

Python

```
states_crime = crime_rate_df[crime_rate_df['state'] != 'United States']
states_crime.head()
```

```
fig, ax = plt.subplots(figsize = (10, 6))

plt.scatter(states_crime['population'], states_crime['murder'], c = 'red', label = 'Murder')
plt.scatter(states_crime['population'], states_crime['forcible_rape'], c = 'green', label = 'Forcible rape')
plt.scatter(states_crime['population'], states_crime['robbery'], c = 'yellow', label = 'Robbery')
plt.scatter(states_crime['population'], states_crime['aggravated_assault'], c = 'orange', label = 'Aggravated Assault')
plt.scatter(states_crime['population'], states_crime['burglary'], c = 'blue', label = 'Burglary')
plt.scatter(states_crime['population'], states_crime['larceny_theft'], c = 'purple', label = 'Larceny Theft')
plt.scatter(states_crime['population'], states_crime['motor_vehicle_theft'], c = 'gray', label = 'Motor Vehicle Theft')

ax.get_xaxis().set_major_formatter( tick.FuncFormatter(lambda x, p: format(int(x), ',')))

plt.xlabel("Population")
plt.xticks(rotation = 45, ha = 'right', rotation_mode = 'anchor')
plt.ylabel("Crime Rates")
plt.title("Scatter Plot - Python \n2005 U.S. States Crime Rate to Population")

ax.legend(loc=(1.01, 0.7), ncol = 2, title = 'Types of Crime')
plt.tight_layout()

plt.show()

# Save figure
ax.get_figure().savefig('images/scatter-plot-python.png', bbox_inches = 'tight', transparent = True)
```



R

```{r}
#| label: statescrime

states_crime <- crimerate_df[crimerate_df$state != 'United States', ]
states_crime
```

```{r}
#| label: mutatestates

crimes_df <- melt(states_crime, id.vars = c('state', 'population'), variable.name = 'crimes')

crimes_df
```

```{r}
#| label: scatterplot
#| echo: false

fig <- crimes_df %>%
  ggplot(aes(x = population, y = value, color = crimes)) +
  geom_point() +
  ggtitle("Scatter Plot - R \n2005 U.S. States Crime Rate to Population") +
  guides(size = 'none') +
  xlab("Population") +
  ylab("Crime Rates") +
  scale_x_continuous(labels = scales::comma) +
  guides(color = guide_legend(title = "Types of Crime", override.aes = list(size = 8))) +
  scale_color_manual(values = c("murder" = "red",
                        "forcible_rape" = "green",
                        "robbery" ="yellow",
                        "aggravated_assault" = "orange",
                        "burglary" = "blue",
                        "larceny_theft" = "purple",
                        "motor_vehicle_theft" = "gray"),
                     labels = c("Murder", "Forcible Rape", "Robbery", "Aggravated Assault",
                          "Burglary", "Larceny Theft", "Motor Vehicle Theft"))

ggsave("images/scatter-plot-r.png")
```

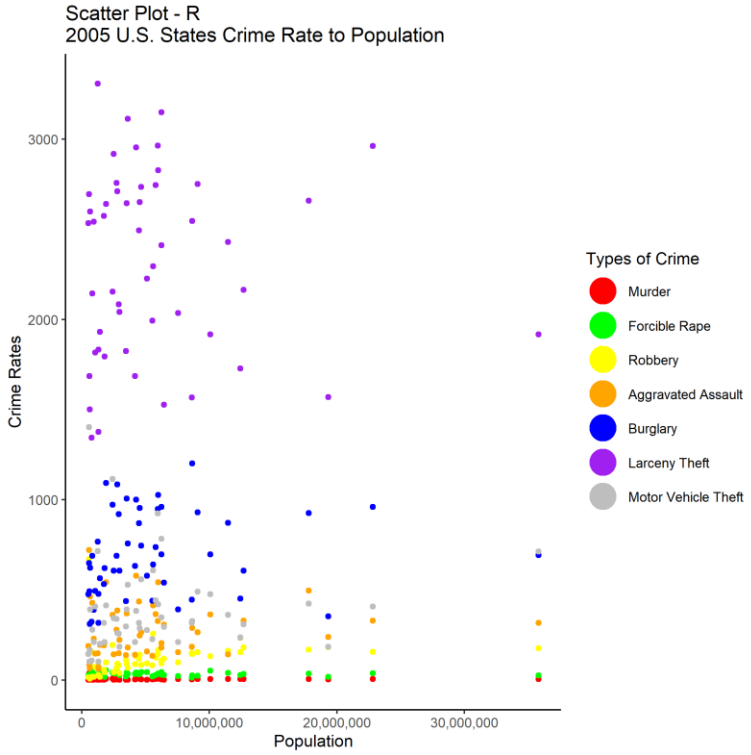
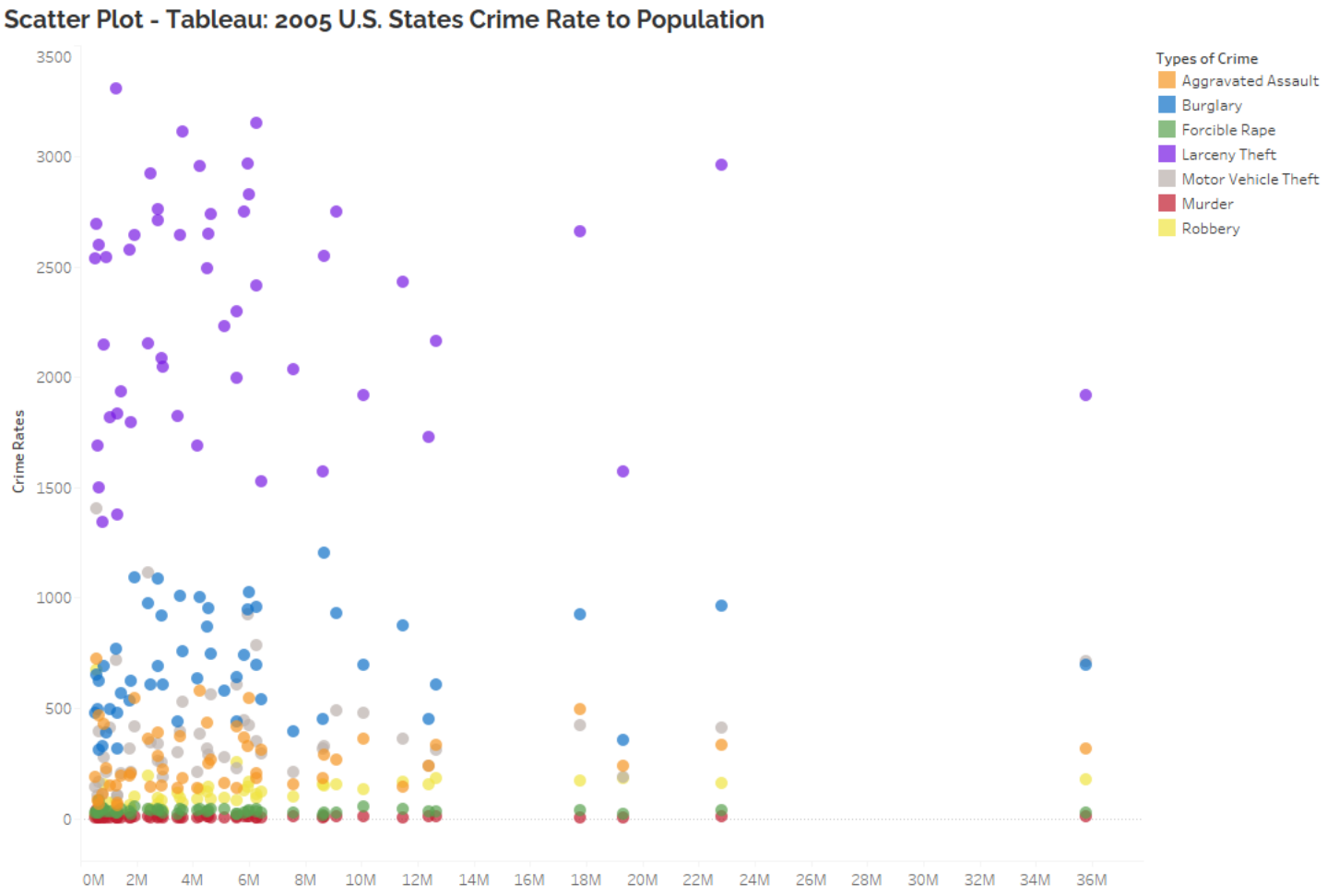


Tableau
(see Weeks7_8_Tableau.twb for code)



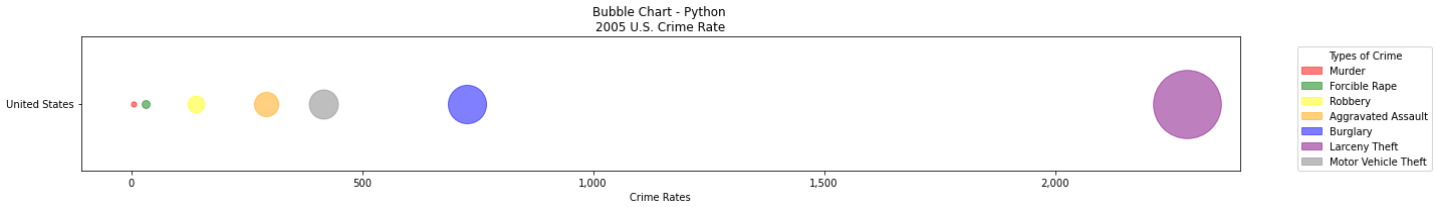
Bubbleplot

Python

```
us_crime = crime_rate_df[crime_rate_df['state'] == 'United States']
us_crime.head()
```

```
fig, ax = plt.subplots(figsize = (20, 3))
```

```
plt.scatter(us_crime['murder'], us_crime['state'], c = 'red', label = 'Murder', s = us_crime['murder'] * 5, alpha = 0.5)
plt.scatter(us_crime['forcible_rape'], us_crime['state'], c = 'green', label = 'Forcible rape', s = us_crime['forcible_rape'] * 2,  alpha = 0.5)
plt.scatter(us_crime['robbery'], us_crime['state'], c = 'yellow', label = 'Robbery', s = us_crime['robbery'] * 2, alpha = 0.5)
plt.scatter(us_crime['aggravated_assault'], us_crime['state'], c = 'orange', label = 'Aggravated Assault', s = us_crime['aggravated_assault'] * 2,
        alpha = 0.5)
plt.scatter(us_crime['burglary'], us_crime['state'], c = 'blue', label = 'Burglary', s = us_crime['burglary'] * 2, alpha = 0.5)
plt.scatter(us_crime['larceny_theft'], us_crime['state'], c = 'purple', label = 'Larceny Theft', s = us_crime['larceny_theft'] * 2, alpha = 0.5)
plt.scatter(us_crime['motor_vehicle_theft'], us_crime['state'], c = 'gray', label = 'Motor Vehicle Theft', s = us_crime['motor_vehicle_theft'] * 2,
        alpha = 0.5)

ax.get_xaxis().set_major_formatter(
   tick.FuncFormatter(lambda x, p: format(int(x), ',')))

plt.xlabel("Crime Rates")
plt.title("Bubble Chart - Python \n2005 U.S. Crime Rate")

crimes_list = ["Murder", "Forcible Rape", "Robbery", "Aggravated Assault", "Burglary",  "Larceny Theft", "Motor Vehicle Theft"]
color_list = ["red", "green", "yellow", "orange", "blue", "purple", "gray"]
legend_list = []
for i in range(0, len(crimes_list)):
   legend_list.append(mpatches.Patch(color = color_list[i], alpha = 0.5, label = crimes_list[i]))
ax.legend(handles = legend_list, loc = (1.05, 0), title = 'Types of Crime')

plt.tight_layout()

plt.show()

# Save figure
ax.get_figure().savefig('images/bubble-chart-python.png',
        bbox_inches = 'tight',
        transparent = True)
```



R

```{r}
#| label: uscrime

us_crime <- crimerate_df[crimerate_df$state == 'United States', ]
us_crime
```

```{r}
#| label: mutateus

uscrimes_df <- melt(us_crime, id.vars = c('state', 'population'), variable.name = 'crimes')

uscrimes_df
```

```{r}
#| label: bubblechart
#| echo: false

fig <- uscrimes_df %>%
  ggplot(aes(x = value, y = state)) +
  geom_point(aes(color = crimes, size = value), alpha = 0.5) +
  scale_size(range = c(0.8, 15), guide = 'none') +
  guides(color = guide_legend(title = "Types of Crime", override.aes = list(size = 8))) +
  scale_color_manual(values = c("murder" = "red",
                          "forcible_rape" = "green",
                          "robbery" ="yellow",
                          "aggravated_assault" = "orange",
                          "burglary" = "blue",
                          "larceny_theft" = "purple",
                          "motor_vehicle_theft" = "gray"),
                labels = c("Murder", "Forcible Rape", "Robbery", "Aggravated Assault",
                          "Burglary", "Larceny Theft", "Motor Vehicle Theft")) +
  ggtitle("Bubble Chart - R \n2005 U.S. Crime Rate") +
  xlab("Crime Rates") +
  ylab(NULL) +
  theme(legend.position="bottom")

ggsave("images/bubble-chart-r.png")
```

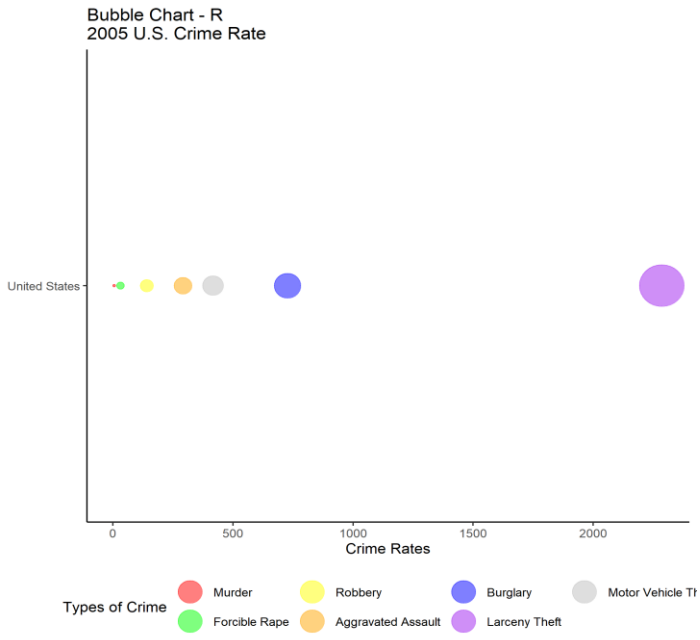
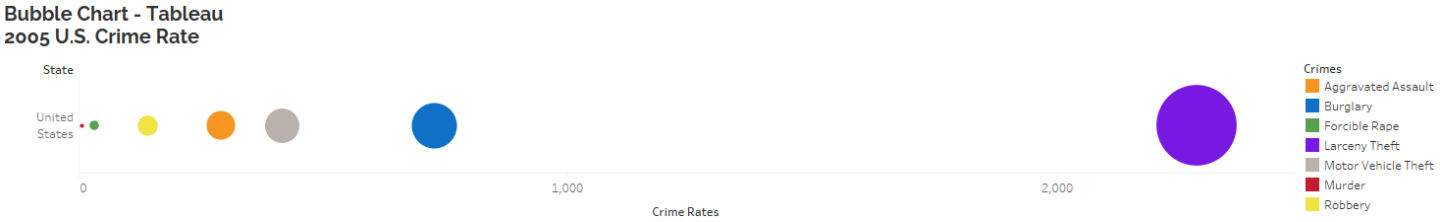


Tableau
(see Weeks7_8_Tableau.twb for code)



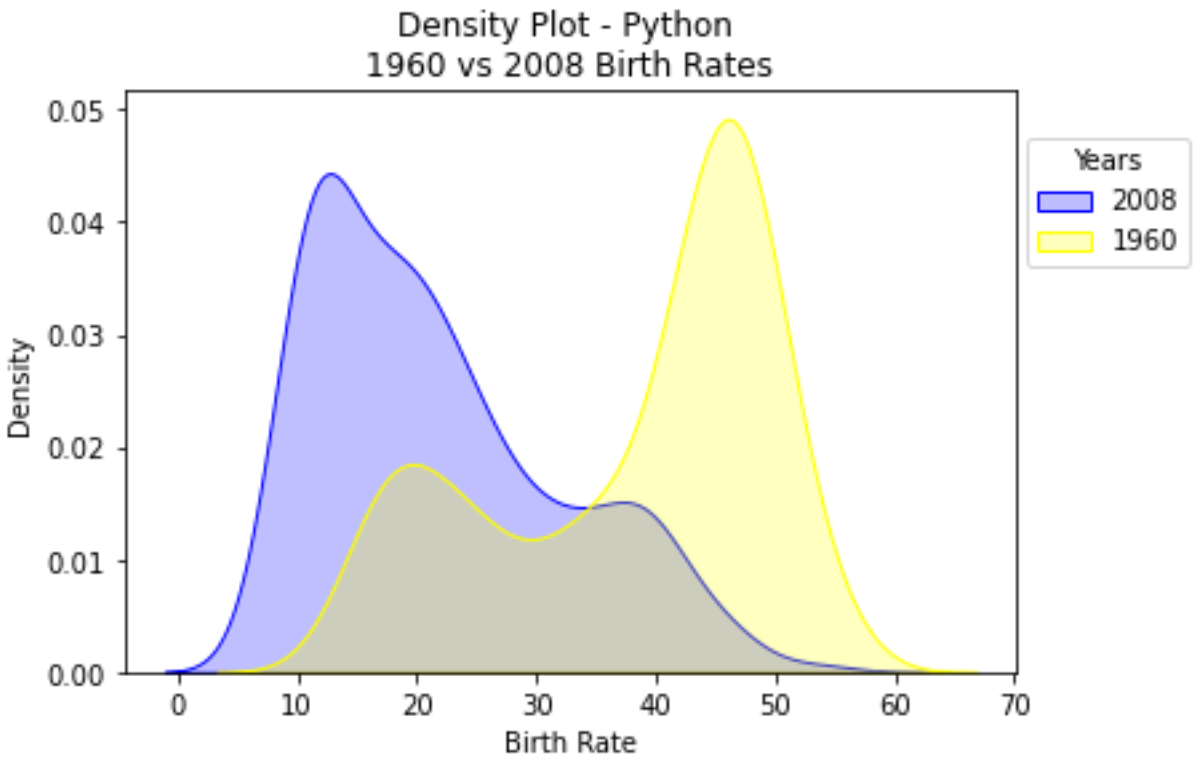
Density Plot/Map

Python

```
birth_2008 = birth_rate_df[birth_rate_df['year'] == 2008]
birth_2008.head()
```

```
birth_1960 = birth_rate_df[birth_rate_df['year'] == 1960]
birth_1960.head()
```

```
density = gaussian_kde(birth_2008['rate'])

fig, ax = plt.subplots()

sb.kdeplot(birth_2008['rate'] , bw_method = 0.3 , fill = True, color = 'blue', label = '2008')
sb.kdeplot(birth_1960['rate'] , bw_method = 0.3 , fill = True, color = 'yellow', label = '1960')

plt.title('Density Plot - Python \n1960 vs 2008 Birth Rates')
plt.xlabel('Birth Rate')

ax.legend(loc=(1.01, 0.7), ncol = 1, title = 'Years')

plt.show()

# Save figure
ax.get_figure().savefig('images/density-plot-python.png',
        bbox_inches = 'tight',
        transparent = True)
```



R

```{r}
#| label: birthyears

births_df <- birthrate_df %>%
  filter(year == 2008 | year == 1960)
births_df <- births_df[order(births_df$year), ]
births_df$year <- as.character(births_df$year)
births_df
```

```{r}
#| label: densityplot
#| echo: false

fig <- births_df %>%
  ggplot(aes(x = rate, fill = year)) +
  geom_density(alpha = 0.3) +
  ggtitle("Density Plot - R\n1960 vs 2008 Birth Rates") +
  labs(fill = 'Year') +
  scale_x_continuous(name="Birth Rates", limits=c(0, 70)) +
  scale_fill_manual(values = c("blue", "yellow"))

ggsave("images/density-plot-r.png")
```

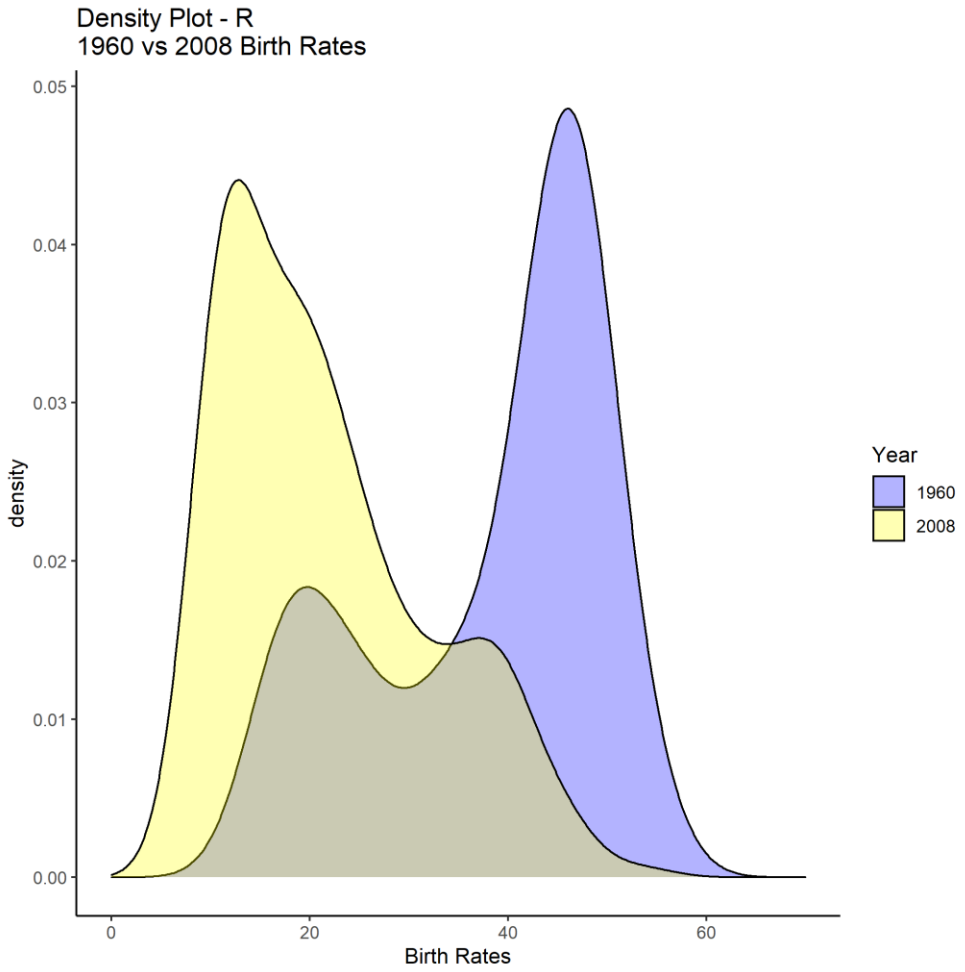


Tableau
(see Weeks7_8_Tableau.twb for code)