Bret Young

DSC 640

Assignment 4.2

25 October 2020

- scatter plot
- bubble plot
- density plot

In [1]:
```
# Import required packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
# Load dataset
url = '~/Desktop/DSC 640/ex4-2/crimerates-by-state-2005.csv'
data = pd.read_csv(url, sep = ',')
```

In [3]:
```
data.head()
```

Out[3]:

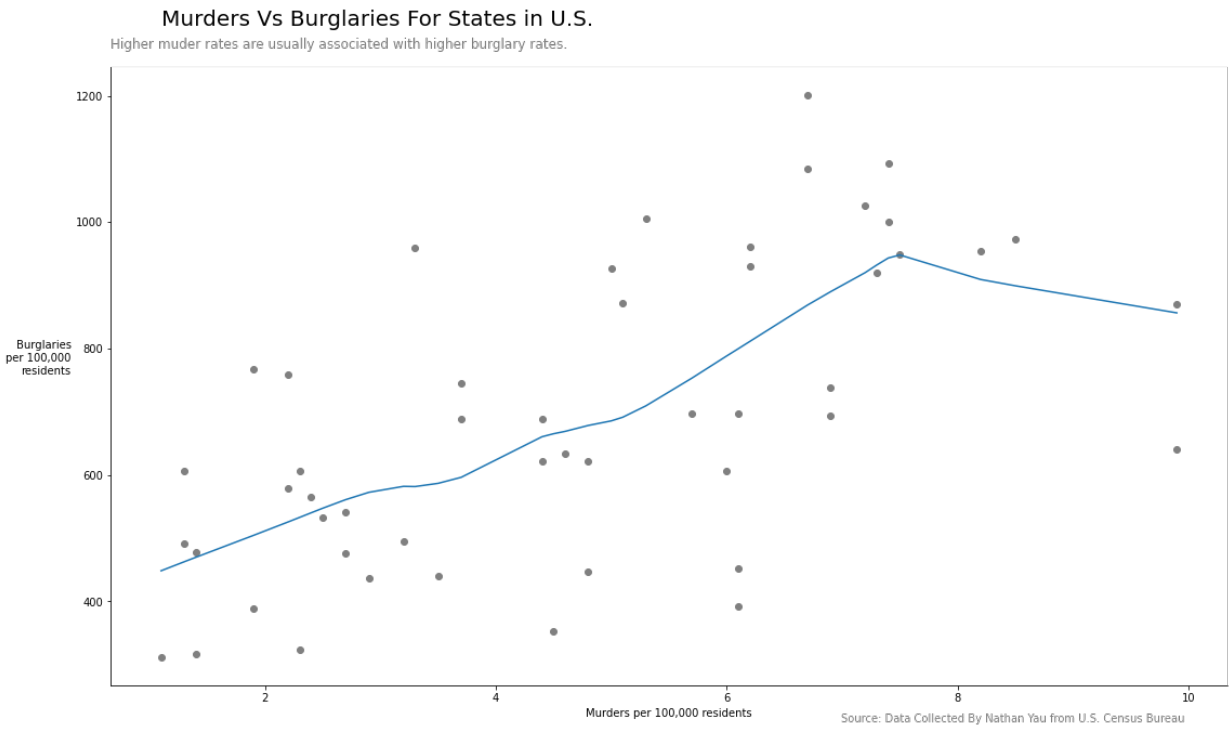
	state	murder	forcible_rape	robbery	aggravated_assault	burglary	larceny_theft	motor
0	United States	5.6	31.7	140.7	291.1	726.7	2286.3	
1	Alabama	8.2	34.3	141.4	247.8	953.8	2650.0	
2	Alaska	4.8	81.1	80.9	465.1	622.5	2599.1	
3	Arizona	7.5	33.8	144.4	327.4	948.4	2965.2	
4	Arkansas	6.7	42.9	91.1	386.8	1084.6	2711.2	

In [4]:
```
# Remove District of Columbia & United States
data_filter = data[(data['state'] != 'United States') & (data['state']
!= 'District of Columbia')]
```

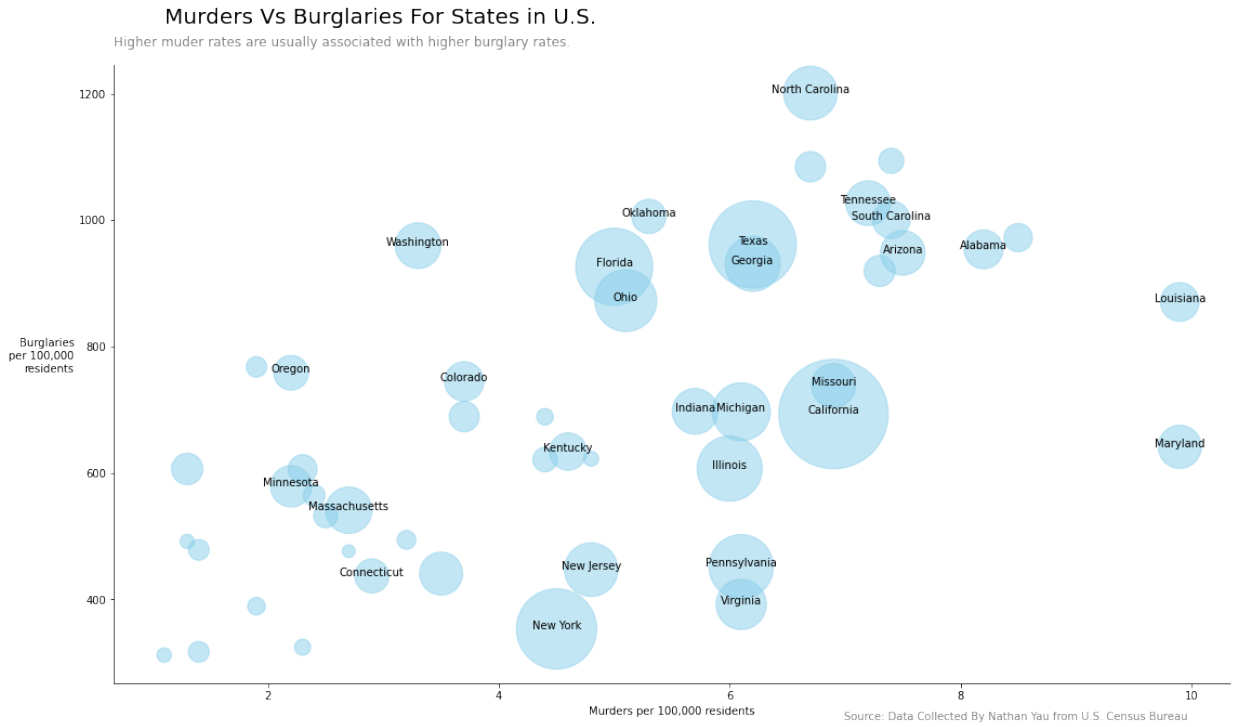
In [5]:
```
# loading required libraries
import scipy.stats as stats
import statsmodels.api as sm

# creating LOESS curve
lowess = sm.nonparametric.lowess
w = lowess(data_filter['burglary'], data_filter['murder'], frac = 0.5)
```

```
/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.ut
il.testing is deprecated. Use the functions in the public API at pan
das.testing instead.
  import pandas.util.testing as tm
```

In [6]:
```
# Create axes and figure
fig = plt.figure()
ax1 = fig.add_subplot(111)

# Set figure size
fig.set_size_inches(18.5, 10.5)

# Add plot to figure
ax1.scatter(data_filter['murder'], data_filter['burglary'], color = 'g
ray')
ax1.plot(w[:,0], w[:,1])


# Set titles, caption and axis labels
fig.suptitle("Murders Vs Burglaries For States in U.S.", x = 0.31, y =
0.95, fontsize=20)
fig.text(.87, .08, 'Source: Data Collected By Nathan Yau from U.S. Cen
sus Bureau', ha = 'right', color = 'gray')
ax1.set_title("Higher muder rates are usually associated with higher b
urglary rates.", y = 1.02, loc='left', color = 'gray')
ax1.set_xlabel("Murders per 100,000 residents")
ax1.set_ylabel("Burglaries\nper 100,000\nresidents", rotation = 0, ha
= 'right')

# Remove frame
ax1.spines['right'].set_visible(False)
ax1.spines['top'].set_visible(False)

# Show plot
plt.show

# save file
fig.savefig("python_scatter.png")
```


Murders Vs Burglaries For States in U.S.
Higher muder rates are usually associated with higher burglary rates.
Burglaries per 100,000 residents
1200
1000
800
600
400
2
4
6
8
10
Murders per 100,000 residents
Source: Data Collected By Nathan Yau from U.S. Census Bureau

In [7]:
```
# Create axes and figure
fig = plt.figure()
ax1 = fig.add_subplot(111)

# Set figure size
fig.set_size_inches(18.5, 10.5)

# Add plot to figure
ax1.scatter(data_filter['murder'], data_filter['burglary'], color = 's
kyblue', s = data_filter['population']/3500, alpha = 0.5)
# add a label inside the bubbles
for line in range(0, data_filter.shape[0]):
    try:
        if data_filter['population'][line] >= 3000000:
            ax1.text(data_filter['murder'][line], data_filter['burglar
y'][line], data_filter['state'][line], horizontalalignment='center', s
ize='medium', color='black')
        else:
            continue
    except KeyError:
        continue

# Set titles, caption and axis labels
fig.suptitle("Murders Vs Burglaries For States in U.S.", x = 0.31, y =
0.95, fontsize=20)
fig.text(.87, .08, 'Source: Data Collected By Nathan Yau from U.S. Cen
sus Bureau', ha = 'right', color = 'gray')
ax1.set_title("Higher muder rates are usually associated with higher b
urglary rates.", y = 1.02, loc='left', color = 'gray')
ax1.set_xlabel("Murders per 100,000 residents")
ax1.set_ylabel("Burglaries\nper 100,000\nresidents", rotation = 0, ha
= 'right')

# Remove frame
ax1.spines['right'].set_visible(False)
ax1.spines['top'].set_visible(False)

# Show plot
plt.show

# save file
fig.savefig("python_bubble.png")
```



In [8]:
```
# Load dataset
url_2 = '~/Desktop/DSC 640/ex4-2/birth-rate.csv'
data_2 = pd.read_csv(url_2, sep = ',')
```

In [9]:
```
data_2.head()
```

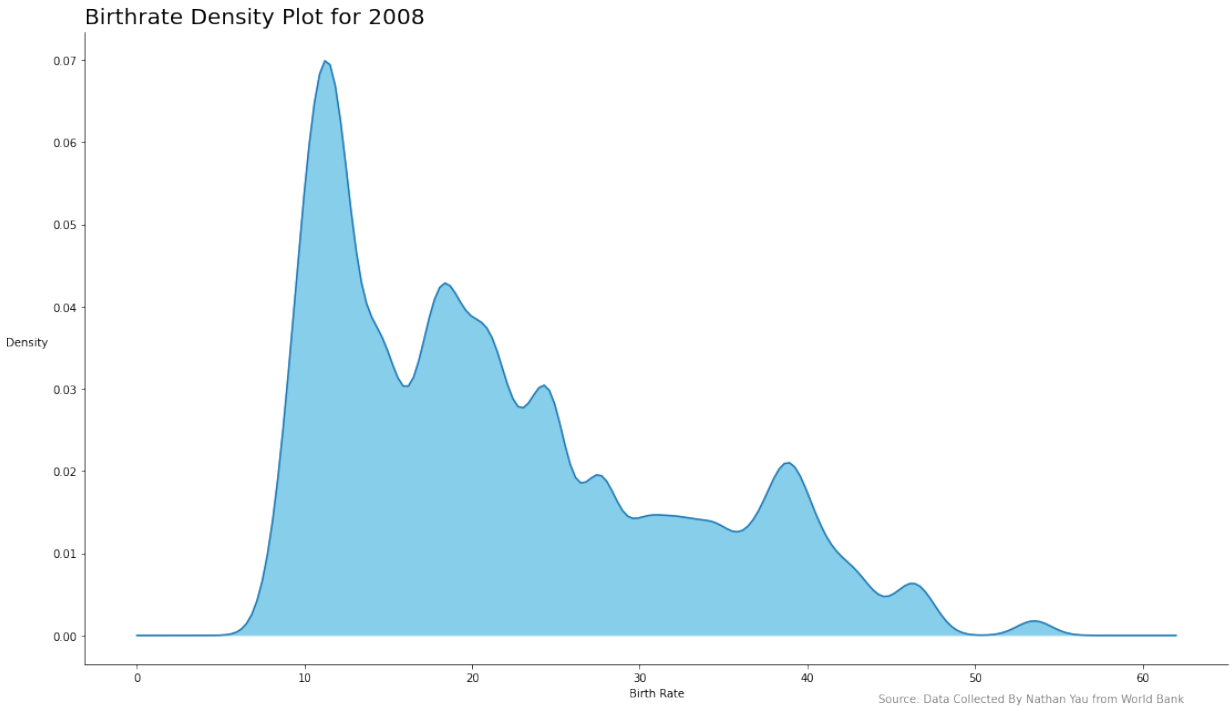
Out[9]:

	Country	1960	1961	1962	1963	1964	1965	1966	1967	1968	...	19
0	Aruba	36.400	35.179	33.863	32.459	30.994	29.513	28.069	26.721	25.518	...	15.0
1	Afghanistan	52.201	52.206	52.208	52.204	52.192	52.168	52.130	52.076	52.006	...	51.2
2	Angola	54.432	54.394	54.317	54.199	54.040	53.836	53.585	53.296	52.984	...	48.6
3	Albania	40.886	40.312	39.604	38.792	37.913	37.008	36.112	35.245	34.421	...	17.7
4	Netherlands Antilles	32.321	30.987	29.618	28.229	26.849	25.518	24.280	23.173	22.230	...	15.8

5 rows × 50 columns

In [10]:
```
# load package needed to create density plot
from scipy.stats import gaussian_kde

# generate density curve
density = gaussian_kde(data_2['2008'].dropna(how='all'))
density.covariance_factor = lambda : .1
xs = np.linspace(0,62,200)
density._compute_covariance()
```

In [11]:
```
# Create axes and figure
fig = plt.figure()
ax1 = fig.add_subplot(111)

# Set figure size
fig.set_size_inches(18.5, 10.5)

# Add plot to figure
ax1.plot(xs,density(xs))
ax1.fill(xs, density(xs), color = 'skyblue')


# Set titles, caption and axis labels
fig.suptitle("Birthrate Density Plot for 2008", x = 0.24, y = 0.91, fo
ntsize=20)
fig.text(.87, .08, 'Source: Data Collected By Nathan Yau from World Ba
nk', ha = 'right', color = 'gray')
ax1.set_xlabel("Birth Rate")
ax1.set_ylabel("Density", rotation = 0, ha = 'right')

# Remove frame
ax1.spines['right'].set_visible(False)
ax1.spines['top'].set_visible(False)

# Show plot
plt.show

# save file
fig.savefig("python_density.png")
```


Birthrate Density Plot for 2008
Density
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
10
20
30
40
50
60
Birth Rate
Source: Data Collected By Nathan Yau from World Bank

In [ ]: