

```
In [1]: import requests
from bs4 import BeautifulSoup
import pandas as pd
from selenium import webdriver

url = "https://en.m.wikipedia.org/wiki/2016_United_States_presidential_election"

driver = webdriver.Chrome("/Users/gavin/downloads/chromedriver_win32/chromedriver")
driver.get(url)
content = driver.page_source

soup = BeautifulSoup(content, "html.parser")
table = soup.find("div", {"style": "overflow:auto"}).find("table", {"class": "wikitable sortable jquery-tablesorter"})

table_data = pd.read_html(str(table))
data_list = table_data[0].values.tolist()
data_list = data_list[1:]
selected_columns = [row[0:3] + row[4:6] + row[7:9] for row in data_list]
results_2016 = pd.DataFrame(selected_columns, columns=['State', 'Democratic Votes', 'Dem_%', 'Republican Votes', 'Rep_%', 'Libertarian', 'Lib_%'])
print(results_2016.head(10))

driver.quit()

State Democratic Votes Dem_% Republican Votes Rep_% Libertarian Lib_%
0 Ariz. 1261167 44.58% 1252401 48.88% 186327 4.08%
1 Ark. 380494 33.02% 684872 60.57% 29049 2.64%
2 Calif. 8753788 61.73% 4483818 31.62% 478560 3.37%
3 Colo. 1338778 48.16% 1292484 43.25% 144121 5.18%
4 Conn. 897572 54.57% 673215 40.93% 48676 2.98%
5 Del. 235683 53.89% 185127 41.72% 14757 3.32%
6 D.C. 282830 90.80% 12723 4.89% 4906 1.57%
7 Fla. 4584975 47.82% 4617886 49.82% 287043 2.28%
8 Ga. 1894352 55.46% 1364897 41.90% 52469 3.85%
9 Hawaii 266891 62.22% 128847 30.83% 15954 3.72%

In [2]: url = "https://en.wikipedia.org/wiki/2020_United_States_presidential_election"

driver = webdriver.Chrome("/Users/gavin/downloads/chromedriver_win32/chromedriver")
driver.get(url)
content = driver.page_source

soup = BeautifulSoup(content, "html.parser")

table = soup.find("div", {"style": "overflow:auto"}).find("table", {"class": "wikitable sortable jquery-tablesorter"})

table_data = pd.read_html(str(table))
data_list = table_data[0].values.tolist()
data_list = data_list[2:]
selected_columns = [row[0:3] + row[4:6] + row[7:9] for row in data_list]
results_2020 = pd.DataFrame(selected_columns, columns=['State', 'Democratic Votes', 'Dem_%', 'Republican Votes', 'Rep_%', 'Libertarian', 'Lib_%'])
print(results_2020.head(5))

driver.quit()

State Democratic Votes Dem_% Republican Votes Rep_% Libertarian Lib_%
0 Arizona 1672143 49.38% 1661886 49.86% 51465 1.52%
1 Ark. 423932 34.78% 760647 62.40% 13133 1.08%
2 Calif. 1119639 63.48% 6806518 34.32% 187910 1.07%
3 Colo. 1694352 55.46% 1364897 41.90% 52469 3.13%
4 Conn. 1898831 59.26% 714717 39.19% 20230 1.11%

Lib_%
0 1.52%
1 1.08%
2 1.07%
3 3.13%
4 1.11%

In [3]: results_2016['winner'] = results_2016.apply(lambda row: 'Democrats' if row['Dem_%'] > row['Rep_%'] else 'Republican', axis=1)

print(results_2016.head(10))

State Democratic Votes Dem_% Republican Votes Rep_% Libertarian Lib_%
0 Ariz. 1261167 44.58% 1252401 48.88% 186327 4.08%
1 Ark. 380494 33.02% 684872 60.57% 29049 2.64%
2 Calif. 8753788 61.73% 4483818 31.62% 478560 3.37%
3 Colo. 1338778 48.16% 1292484 43.25% 144121 5.18%
4 Conn. 897572 54.57% 673215 40.93% 48676 2.98%
5 Del. 235683 53.89% 185127 41.72% 14757 3.32%
6 D.C. 282830 90.80% 12723 4.89% 4906 1.57%
7 Fla. 4584975 47.82% 4617886 49.82% 287043 2.28%
8 Ga. 1894352 55.46% 1364897 41.90% 52469 3.85%
9 Hawaii 266891 62.22% 128847 30.83% 15954 3.72%

Lib_%
0 4.08%
1 2.64%
2 3.37%
3 5.18%
4 2.98%
5 3.32%
6 1.57%
7 2.28%
8 3.85%
9 3.72%

In [4]: results_2020['winner'] = results_2020.apply(lambda row: 'Democrats' if row['Dem_%'] > row['Rep_%'] else 'Republican', axis=1)

print(results_2020.head(5))

State Democratic Votes Dem_% Republican Votes Rep_% Libertarian Lib_%
0 Arizona 1672143 49.38% 1661886 49.86% 51465 1.52%
1 Ark. 423932 34.78% 760647 62.40% 13133 1.08%
2 Calif. 1119639 63.48% 6806518 34.32% 187910 1.07%
3 Colo. 1694352 55.46% 1364897 41.90% 52469 3.13%
4 Conn. 1898831 59.26% 714717 39.19% 20230 1.11%

Lib_%
0 1.52%
1 1.08%
2 1.07%
3 3.13%
4 1.11%

In [5]: import matplotlib.pyplot as plt

dem_percentage = float(results_2016.at[54, 'Dem_%'].strip('%'))
rep_percentage = float(results_2016.at[54, 'Rep_%'].strip('%'))
lib_percentage = float(results_2016.at[54, 'Lib_%'].strip('%'))

percentages = [dem_percentage, rep_percentage, lib_percentage]
parties = ['Democrats', 'Republican', 'Libertarian']
colors = ['blue', 'red', 'yellow']

plt.pie(percentages, labels=parties, colors=colors, autopct='%1.1f%%')
plt.axis('equal')
plt.title('Presidential Election Results - 2016')
plt.show()

dem_percentage = float(results_2020.at[54, 'Dem_%'].strip('%'))
rep_percentage = float(results_2020.at[54, 'Rep_%'].strip('%'))
lib_percentage = float(results_2020.at[54, 'Lib_%'].strip('%'))

percentages = [dem_percentage, rep_percentage, lib_percentage]
parties = ['Democrats', 'Republican', 'Libertarian']
colors = ['blue', 'red', 'yellow']

plt.pie(percentages, labels=parties, colors=colors, autopct='%1.1f%%')
plt.axis('equal')
plt.title('Presidential Election Results - 2020')
plt.show()

Presidential Election Results - 2016
Democrats
49.4%
Republican
47.2%
Libertarian
3.4%

Presidential Election Results - 2020
Democrats
51.7%
Republican
47.2%
Libertarian
1.2%

In [6]: import geopandas as gpd
import matplotlib.pyplot as plt

data = pd.read_excel("Python project State Data.xlsx")
gdf = gpd.read_file("ne_10m_admin_1_states_provinces.shp")
merged_gdf = gdf.merge(data, left_on="name", right_on="State ", how="left")

fig, ax = plt.subplots(figsize=(12, 8))
merged_gdf.plot(ax=ax, column="2016_Result", cmap="magma_r", legend=True)
plt.title('Political Affiliation of U.S. States (2016)')
plt.show()

fig, ax = plt.subplots(figsize=(12, 8))
merged_gdf.plot(ax=ax, column="2020_Result ", cmap="magma_r", legend=True)
plt.title('Political Affiliation of U.S. States (2020)')
plt.show()

Political Affiliation of U.S. States (2016)
70
60
50
40
30
20
-160 -140 -120 -100 -80
Democrats
Republican

Political Affiliation of U.S. States (2020)
70
60
50
40
30
20
-160 -140 -120 -100 -80
Democrats
Republican

In [7]: excel_file = "C:/Users/gavin/OneDrive/Documents/Python project State Data.xlsx"

compiled_data = pd.read_excel(excel_file)

print(compiled_data)
pd.set_option('display.max_columns', 6)
pd.set_option('display.max_row', 6)
pd.set_option('display.expand_frame_repr', False)
pd.set_option('display.width', None)

State 2016_Result 2020_Result 2016_Voter_Turnout \
0 Alabama Republican Republican 58.3
1 Alaska Republican Republican 57.4
2 Arizona Republican Democrats 48.9
3 Arkansas Republican Republican 49.4
4 California Democrats Democrats 47.0
5 Colorado Democrats Democrats 64.6
6 Connecticut Democrats Democrats 58.3
7 Delaware Democrats Democrats 59.2
8 Florida Republican Republican 56.9
9 Georgia Republican Democrats 52.6
10 Hawaii Democrats Democrats 61.1
11 Idaho Republican Republican 55.0
12 Illinois Democrats Democrats 56.1
13 Indiana Republican Republican 54.0
14 Iowa Republican Republican 65.1
15 Kansas Republican Republican 55.4
16 Kentucky Republican Republican 56.1
17 Louisiana Republican Republican 56.8
18 Maine Democrats Democrats 69.4
19 Maryland Democrats Democrats 59.5
20 Massachusetts Democrats Democrats 62.0
21 Michigan Republican Democrats 58.3
22 Minnesota Democrats Democrats 69.4
23 Mississippi Republican Republican 53.3
24 Missouri Republican Republican 59.6
25 Montana Republican Republican 69.8
26 Nebraska Republican Republican 58.8
27 Nevada Democrats Democrats 49.4
28 New Hampshire Democrats Democrats 68.1
29 New Jersey Democrats Democrats 55.7
30 New Mexico Democrats Democrats 59.2
31 New York Democrats Democrats 49.6
32 North Carolina Republican Republican 69.2
33 North Dakota Republican Republican 59.3
34 Ohio Republican Republican 61.0
35 Oklahoma Republican Republican 49.0
36 Oregon Democrats Democrats 61.7
37 Pennsylvania Republican Democrats 61.0
38 Rhode Island Democrats Democrats 54.7
39 South Carolina Republican Republican 54.2
40 South Dakota Republican Republican 52.6
41 Tennessee Republican Republican 48.6
42 Texas Republican Republican 43.4
43 Utah Republican Republican 52.8
44 Vermont Democrats Democrats 62.8
45 Virginia Democrats Democrats 68.3
46 Washington Democrats Democrats 58.3
47 West Virginia Republican Republican 49.2
48 Wisconsin Republican Democrats 66.2
49 Wyoming Republican Republican 57.3

2020_Voter_Turnout 2016_Black_Demographic 2020_Black_Demographic \
0 63.1 0.27 0.25
1 68.8 0.63 0.628
2 65.9 0.94 0.841
3 56.1 0.15 0.144
4 68.5 0.85 0.851
5 76.4 0.84 0.838
6 71.5 0.10 0.097
7 70.7 0.21 0.203
8 71.7 0.15 0.144
9 67.7 0.31 0.303
10 57.5 0.01 0.013
11 67.7 0.07 0.067
12 67.0 0.14 0.132
13 61.4 0.09 0.088
14 73.2 0.63 0.634
15 65.9 0.06 0.047
16 64.9 0.06 0.068
17 64.6 0.32 0.306
18 76.3 0.82 0.814
19 70.7 0.29 0.286
20 72.1 0.07 0.061
21 73.9 0.14 0.131
22 66.3 0.06 0.067
23 66.2 0.38 0.357
24 66.3 0.38 0.379
25 69.9 0.11 0.084
26 65.4 0.08 0.086
27 65.4 0.08 0.086
28 75.5 0.93 0.812
29 37.1 0.38 0.354
30 38.6 0.90 0.804
31 38.6 0.90 0.804
32 38.5 0.90 0.804
33 71.5 0.21 0.197
34 64.5 0.63 0.629
35 55.0 0.07 0.065
36 75.5 0.12 0.118
37 71.0 0.10 0.097
38 65.0 0.06 0.043
39 66.8 0.27 0.257
40 36.8 0.91 0.828
41 36.9 0.88 0.847
42 34.3 0.16 0.147
43 69.2 0.01 0.010
44 74.2 0.61 0.611
45 73.0 0.19 0.177
46 75.7 0.83 0.837
47 57.6 0.39 0.025
48 75.8 0.06 0.058
49 64.6 0.61 0.059

2016_Median_Age 2020_Median_Age 2016_Average_Income \
0 38.6 39.2 46257
1 33.8 34.6 76440
2 37.4 37.9 53658
3 37.7 38.3 44334
4 36.7 36.7 67739
5 36.9 36.9 65685
6 40.5 41.1 73433
7 39.4 41.0 61557
8 41.6 42.2 58868
9 36.5 36.9 53575
10 38.8 39.4 74511
11 35.8 36.6 51807
12 37.0 38.3 60868
13 37.5 37.8 52314
14 38.1 38.3 56247
15 36.3 36.9 54935
16 38.7 39.0 67806
17 36.4 37.2 45146
18 44.1 44.8 53079
19 38.3 38.8 78945
20 39.4 39.6 75297
21 39.8 39.8 52492
22 37.9 38.1 65559
23 36.7 37.7 41754
24 38.5 38.7 51746
25 39.8 40.1 58927
26 36.4 36.6 56927
27 37.9 38.2 59380
28 42.7 43.0 78936
29 39.0 40.0 76126
30 37.1 38.1 46748
31 38.6 39.0 62909
32 38.5 38.9 58964
33 35.8 36.2 68656
34 39.2 39.5 52334
35 36.3 36.7 49176
36 39.0 39.5 57532
37 40.6 40.9 45146
38 39.7 40.0 60596
39 39.1 39.7 49561
40 36.8 37.2 54667
41 36.9 38.8 48547
42 34.3 34.8 52597
43 30.8 31.1 65977
44 42.8 42.8 57877
45 38.4 38.4 65144
46 37.9 37.8 67106
47 42.4 42.7 42365
48 39.3 39.6 56811
49 37.6 38.0 59882

2020_Average_Income 2016_Educated_Population 2020_Educated_Population \
0 51254 0.2467 32.79%
1 75463 0.3834 32.78%
2 62655 0.2932 32.43%
3 48952 0.2238 25.27%
4 80440 0.3365 36.19%
5 72327 0.4091 44.42%
6 78833 0.3932 42.13%
7 70176 0.3274 35.62%
8 59227 0.2988 33.18%
9 61980 0.3097 34.63%
10 83282 0.298 35.98%
11 60999 0.2724 36.72%
12 69187 0.3496 37.14%
13 57683 0.2579 28.88%
14 61691 0.2715 36.54%
15 62887 0.3259 35.48%
16 52295 0.2374 26.88%
17 51073 0.2392 26.45%
18 58843 0.3884 36.62%
19 80738 0.3857 42.55%
20 85045 0.2776 45.82%
21 59584 0.2843 31.67%
22 74983 0.3535 38.08%
23 45782 0.2272 24.78%
24 57489 0.2866 31.72%
25 57155 0.3154 34.68%
26 63229 0.3165 34.45%
27 63276 0.2496 27.57%
28 77933 0.3664 40.25%
29 85761 0.3927 43.18%
30 51845 0.2778 30.95%
31 72188 0.3866 39.82%
32 57341 0.3883 34.91%
33 64577 0.2870 31.74%
34 58642 0.2769 30.72%
35 54449 0.2954 36.33%
36 67858 0.3200 36.23%
37 63463 0.3091 34.54%
38 51953 0.2879 36.60%
39 56227 0.2799 31.53%
40 6968 0.2676 31.67%
41 56071 0.2878 30.46%
42 64634 0.2952 33.32%
43 75980 0.3368 36.61%
44 63801 0.3888 44.44%
45 74586 0.3759 38.87%
46 78897 0.3855 38.87%
47 48560 0.2861 29.24%
48 64168 0.2965 32.54%
49 65083 0.2662 24.12%

Violent_Crime_2016
0 884.6
1 532.4
2 471.0
3 554.3
4 444.8
5 344.1
6 227.7
7 510.0
8 420.4
9 598.4
10 241.6
11 330.7
12 436.7
13 399.7
14 292.9
15 481.2
16 335.6
17 565.0
18 124.0
19 469.9
20 241.9
21 590.8
22 241.9
23 281.7
24 520.9
25 374.1
26 296.5
27 677.9
28 199.4
29 243.5
30 699.8
31 371.8
32 387.7
33 252.1
34 391.7
35 455.3
36 260.8
37 315.9
38 239.1
39 596.8
40 422.0
41 630.5
42 436.5
43 439.5
44 219.8
45 283.5
46 382.7
47 386.9
48 244.7
49 386.9

In [8]: compiled_data.columns = compiled_data.columns.str.strip()

combined_average_turnout = compiled_data[['2016_Voter_Turnout', '2020_Voter_Turnout']].mean().mean()

above_average_turnout = compiled_data[(compiled_data['2016_Voter_Turnout'] > combined_average_turnout) &
(compiled_data['2020_Voter_Turnout'] > combined_average_turnout)]

correlation_2016 = above_average_turnout['2016_Result'].value_counts(normalize=True)
correlation_2020 = above_average_turnout['2020_Result'].value_counts(normalize=True)

print("Combined Average Turnout:", combined_average_turnout)
print("Correlation of 2016 Result:")
print(correlation_2016)
print("Correlation of 2020 Result:")
print(correlation_2020)

Combined Average Turnout: 62.26

Correlation of 2016 Result:
Democrats 0.714286
Republican 0.285714
Name: 2016_Result, dtype: float64

Correlation of 2020 Result:
Democrats 0.857143
Republican 0.142857
Name: 2020_Result, dtype: float64

In [9]: compiled_data['2016_Median_Age'] = pd.to_numeric(compiled_data['2016_Median_Age'], errors='coerce')
compiled_data['2020_Median_Age'] = pd.to_numeric(compiled_data['2020_Median_Age'], errors='coerce')

average_age_2016 = compiled_data['2016_Median_Age'].mean()
average_age_2020 = compiled_data['2020_Median_Age'].mean()
average_age_combined = (average_age_2016 + average_age_2020) / 2

print("Average Age (2016):", average_age_2016)
print("Average Age (2020):", average_age_2020)
print("Average Age (Combined):", average_age_combined)

Average Age (2016): 38.242
Average Age (2020): 38.647999999999996
Average Age (Combined): 38.449999999999999

In [10]: compiled_data['2016_Result'] = compiled_data['2016_Result'].map({'Republican': 1, 'Democrats': -1})

correlation = compiled_data['2016_Median_Age'].corr(compiled_data['2016_Result'])

print("Correlation between older than average population and voting result in 2016: (correlation)")

Correlation between older than average population and voting result in 2016: -0.32801857789538174

In [11]: compiled_data['Income_Increase'] = compiled_data['2020_Average_Income'] - compiled_data['2016_Average_Income']
average_increase = compiled_data['Income_Increase'].mean()

print("The average increase in income between 2016 and 2020 is: (average_increase)")

The average increase in income between 2016 and 2020 is: 7180.76

In [12]: above_average_states = compiled_data[compiled_data['Income_Increase'] > average_increase]

party_percentage = above_average_states['2020_Result'].value_counts(normalize=True) * 100

print("Percentage of states with an above-average income increase for each party:")
Percentage of states with an above-average income increase for each party:
Democrats 78.26687
Republicans 21.73313
Name: 2020_Result, dtype: float64

In [13]: excel_file = "C:/Users/gavin/OneDrive/Desktop/Wisconsin_Pop_Data.xlsx"

Wisconsin_Data = pd.read_excel(excel_file)

print(Wisconsin_Data)
pd.set_option('display.max_columns', None)

County Pop_Density Voting_Outcome
0 Adams 29.9 Republican
1 Ashland 7.0 Democrats
2 Barron 51.4 Republican
3 Beaver Dam 1.0 Democrats
4 Dodge 1.0 Democrats
5 Janesville 291.1 Republican
6 Madison 291.1 Republican
7 Monona 91.7 Republican
8 Wood 91.7 Republican

[72 rows x 3 columns]

In [14]: from sklearn.cluster import KMeans

k = (Wisconsin_Data)

x = df[['Pop_Density']]
kmeans = KMeans(n_clusters=2, random_state=0)
kmeans.fit(x)
df['cluster'] = kmeans.labels_

cluster_outcomes = df.groupby('cluster')['Voting_Outcome'].value_counts()

print(cluster_outcomes)

C:\Users\gavin\New folder\New folder\libsite-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default value of 'n_init' will change from 10 to 'auto' in 1.4. Set the value of 'n_init' explicitly to suppress the warning
C:\Users\gavin\New folder\New folder\libsite-packages\sklearn\cluster\_kmeans.py:1382: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are 16 or more clusters. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
warnings.warn(

cluster_outcomes
cluster Voting_Outcome
0 Republican 7
1 Democrats 4
2 Republican 51
3 Democrats 10
Name: Voting_Outcome, dtype: int64

In [15]: print(results_2020.iloc[52])

State Wis.
Democratic Votes 1638866
Dem_% 49.45%
Libertarian ...
Lib_% 38491
Lib_% 1.17%
winner Democrats
Name: 52, Length: 8, dtype: object

In [ ]:
```