Introduction

When you think about the most prestigious universities in the United States, which schools immediately come to mind? Harvard? Yale? Princeton? MIT? UMBC? If you were to mark those schools on a map of the United States, you'd find that they're all clustered together in the top right-hand corner. In a nation which spans 3,500 miles, and is home to over 300 million people, it seems like many (if not most) of its highly rated colleges happen to be located in very close proximity to each other. Is this really the case, or is it an optical illusion? If it is indeed the case, why is this so? Which regional attributes can help explain this disparity? These are the questions that we set out to answer with this analysis.

Our research strategy was as follows:

1. Cluster the United States into regions of states which are geographically and culturally similar to each other.
2. Determine whether there is a significant difference in the average ranking of the universities within each of these regions.
3. If such a difference does exist, investigate which specific attributes of a region correlate with a difference in university rankings.

Our overarching goal did not change at all as we conducted our research. However, what did change were the specific regional attributes we were investigating. Some turned out to have no impact on average university ranking, and were therefore discarded. Others turned out to be significant, and were therefore researched further.

Dataset Descriptions

income.csv

income	Integer value indicating average household income in U.S. dollars
state	Two-character state abbreviation

density.csv

density	Float value indicating the population density of a state in people/sq-mile
state	Two-character state abbreviation

hsrankings.csv

rank	Integer value indicating a given state's ranking in comparison to the other states' average high school scores.
state	Two-character state abbreviation
score	Float value indicating the score given to the state's high schools.
quality	Integer value indicating a given state's ranking in comparison to other states based solely on quality.
safety	Integer value indicating a given state's ranking in comparison to other states based solely on safety.

universities.csv

nonResponder	Boolean value indicating whether the school responded to the survey
act_avg	Integer value indicating average ACT score of the students
sat_avg	Integer value indicating average SAT score of the students
enrollment	Integer value indicating the number of students

city	String value indicating the city the college is located in
sortName	String value indicating the name of the college
zip	Integer value indicating the zip code of the college's location
acceptance_rate	Integer value indicating the acceptance rate
rankingDisplayScore	Integer value indicating the score of the university, used to rank them.
percent_receiving_aid	Integer value indicating the percentage of students receiving financial aid
cost_after_aid	Integer value indicating the average cost of attendance after receiving financial aid
state	Two-character abbreviation of the state the university is located in
hs_gpa_avg	Float value indicating average high school GPA of the students
rankingIsTied	Boolean value indicating whether the school is tied with another school in rank
isPublic	Boolean value indicating whether the university is public
businessRepScore	Float value indicating the business reputation
tuition	Integer value indicating the tuition cost of the university
engineeringRepScore	Float value indicating the engineering reputation
institutionalControl	String value indicating whether the school is private or public
univ_rank	Integer value indicating the rank of the university

us_companies.csv

company_name_id	String value indicating the name of a company
state	Two-character abbreviation of the state the companyis located in

uscities.csv

city	String indicating the name of a city
state	String indicating the state the city is located in
population	Integer indicating total population of the city
density	Integer indicating population density of the city

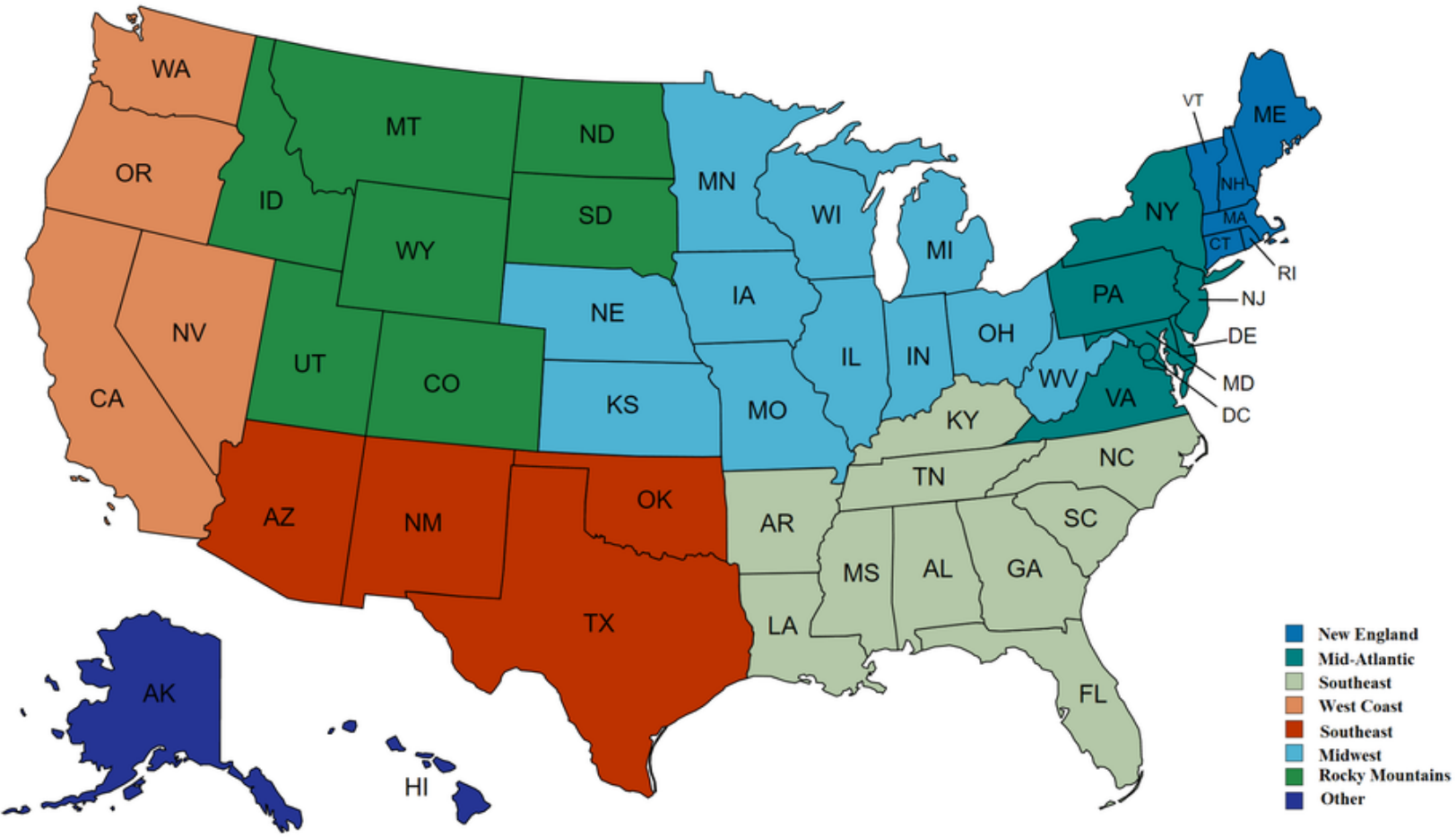
Exploratory Data Analysis

We first grouped all of the states (including the District of Columbia) into eight regions (see Figure 1). These regions were designed both with physical location and cultural similarity in mind.

Figure 1

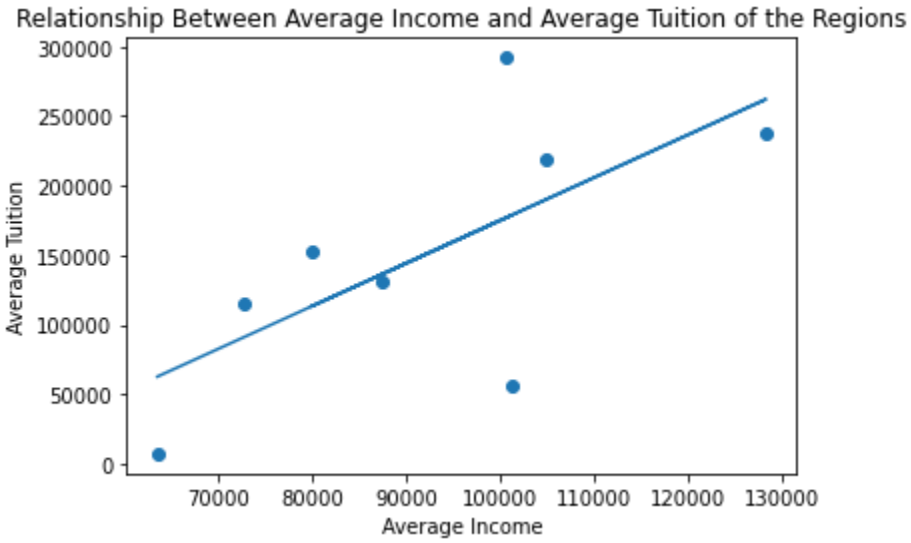
New England	Mid-Atlantic	Midwest	Southeast	West Coast	Rocky Mountains	Southwest
Maine	New York	West Virginia	Arkansas	Washington	Utah	Arizona
Massachusetts	New Jersey	Ohio	Louisiana	Oregon	Colorado	New Mexico
Connecticut	Pennsylvania	Wisconsin	Mississippi	California	Wyoming	Texas
New Hampshire	Maryland	Michigan	Alabama	Nevada	Idaho	Oklahoma
Rhode Island	Delaware	Minnesota	Tennessee		Montana	
Vermont	Virginia	Illinois	Georgia		North Dakota	
	Washington, D.C.	Indiana	North Carolina		South Dakota	
		Iowa	South Carolina			
		Nebraska	Florida			
		Kansas	Kentucky			

Figure 1 (continued)



Next, we imported the provided *Universities* dataset into our Python script. We removed every university which did not have a numeric ranking, as these are wholly unhelpful to us. We then added the attribute “region” to each college, and assigned them a value based on which region from Figure 1 they are located in. We then began our exploratory data analysis. In this stage, we simply examined the dataset we were given and tried to grasp its meaning by exploring the relationships between its various variables.

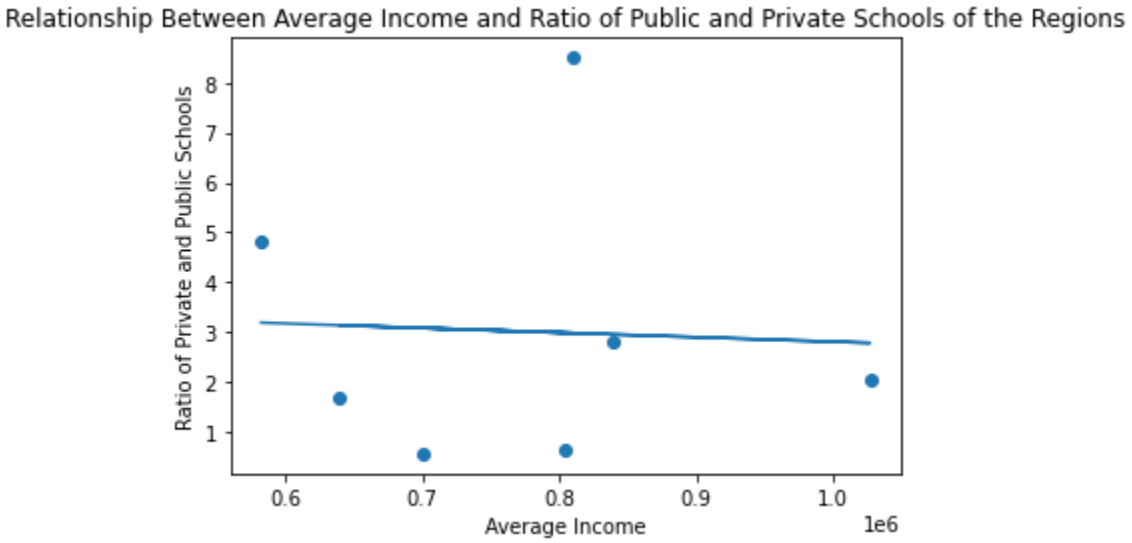
Figure 2



At first, we tried to find out if there is any correlation between average income and average tuition of the regions. In Fig.2. we can see the relationship between average income and average tuition of the regions. Even though it seems that there is a positive correlation between average income and average tuition of the regions, the correlation coefficient (0.136) is not high enough to be conclusive. The two outliers are the Mid-Atlantic and the Rocky Mountains region. For the Mid-Atlantic region, the average tuition is much higher than the other regions and for the Rocky Mountains region, the

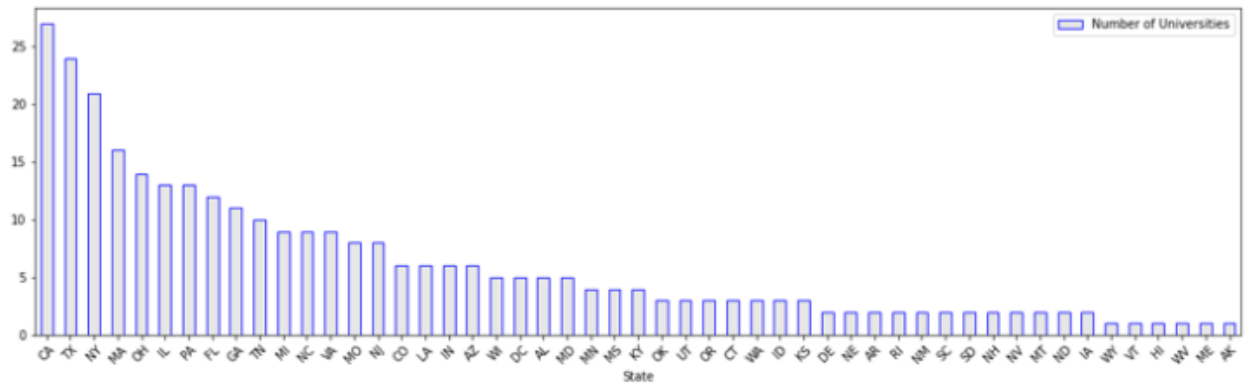
average tuition is much less compared to regions with lower average income. The biggest difference between these two regions is the population density. The Mid-Atlantic region has much higher population density compared to the Rocky Mountains region. This could be a reason why these two regions are outliers but further analysis of socio-economic datasets are required to reach a conclusion.

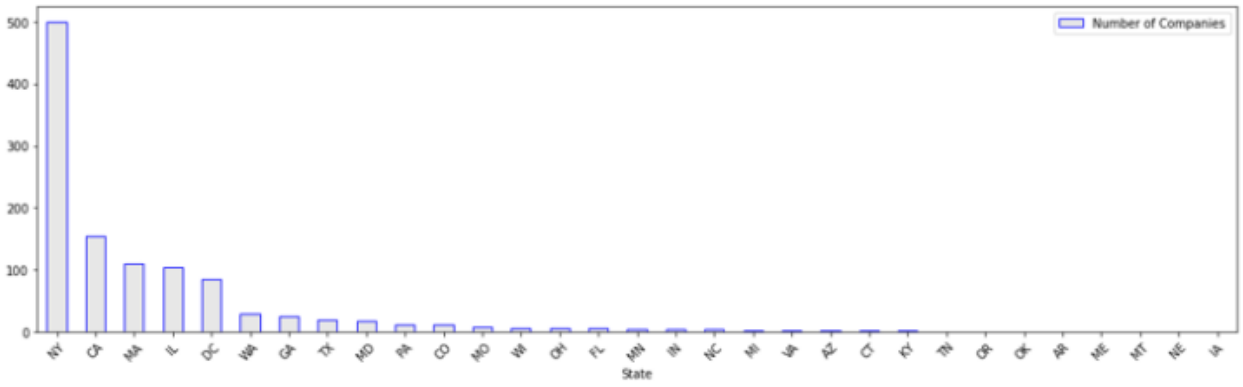
Figure 3



Next, we tried to see if there is any correlation between average income and ratio of public and private universities of the regions. The relationship is plotted in Fig. 3. As we can see from the figure, we couldn't find any correlation between the ratio of public and private universities and average income of the regions.

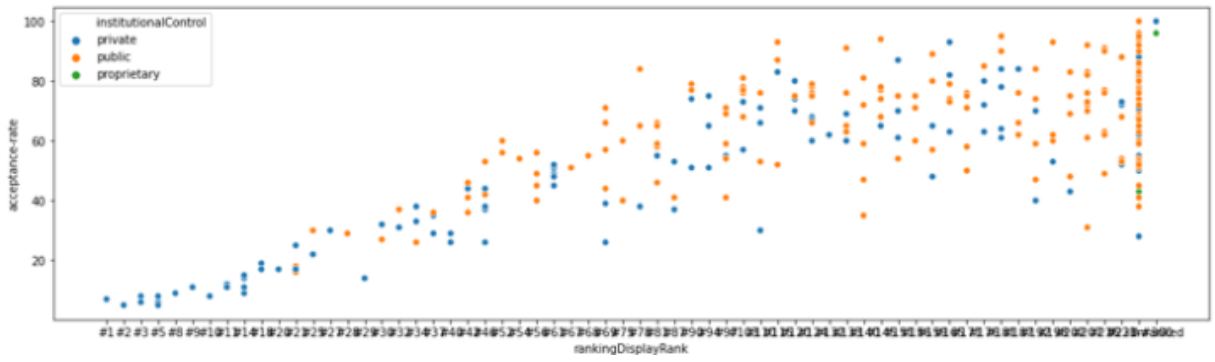
Figure 4





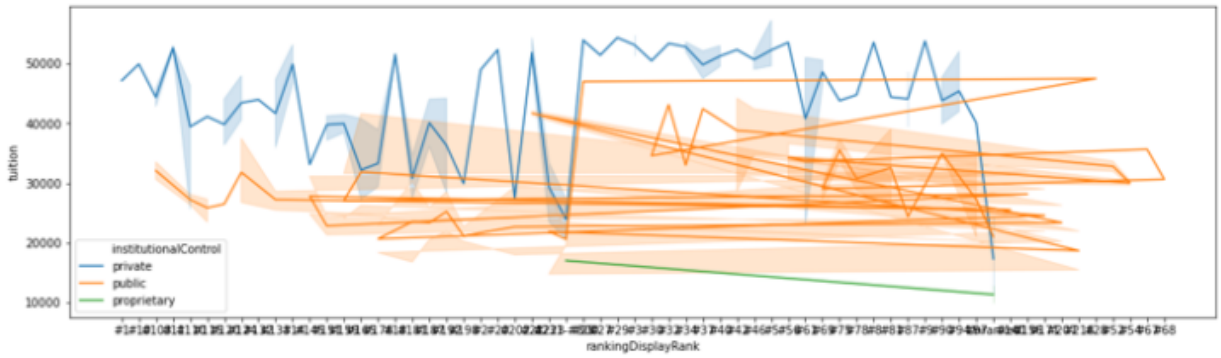
We then decided to investigate whether there was a relationship between the number of universities in a state and the number of major companies in a state; these graphs can be seen in Figure 4. It appears that they're related to some degree, as some of the states (NY, MA especially) that have the highest number of universities also have the highest number of companies. However, the relationship doesn't look especially strong (TX, for example, has the second-highest number of universities, but the eighth-highest number of companies) and as such, we moved on to investigate other factors (additionally: whatever relationship does exist is easily explained by the fact that states with larger populations probably have the most universities and the most companies).

Figure 5



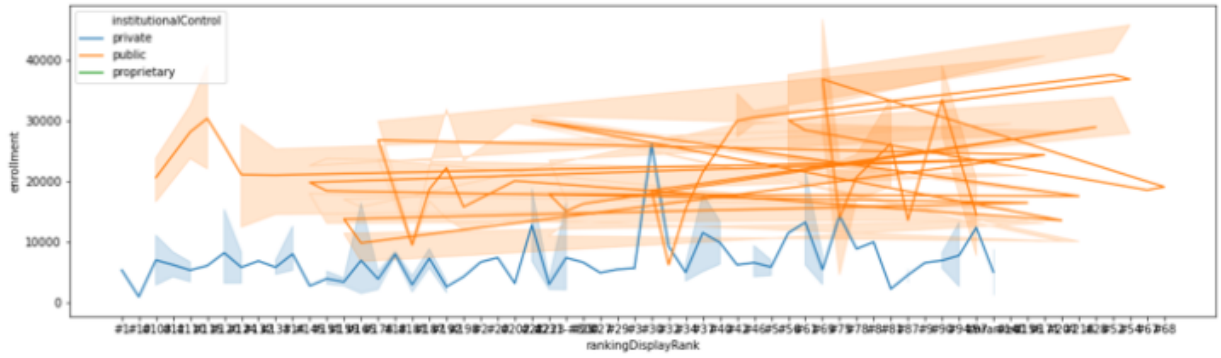
We then evaluated the relationship between a university's ranking and both its acceptance rate, and its status as a private or public university. The results of this analysis are shown in Figure 5. It appears that there is a strong correlation between having a higher ranking and lower acceptance rate; additionally, it seems that public universities largely have higher acceptance rates, and lower rankings.

Figure 6



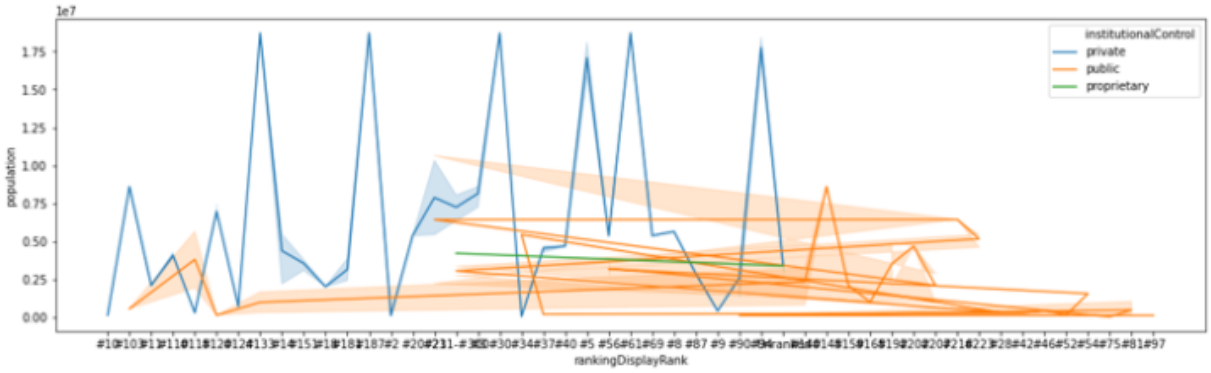
We then investigated whether a university's tuition is related to its ranking. It appears that private universities, on average, have higher tuitions than public universities, and that they are also, on average, higher-ranked than public universities. This finding suggested that the income of a region could be related to the quality of its universities, and encouraged us to explore that link further later.

Figure 7



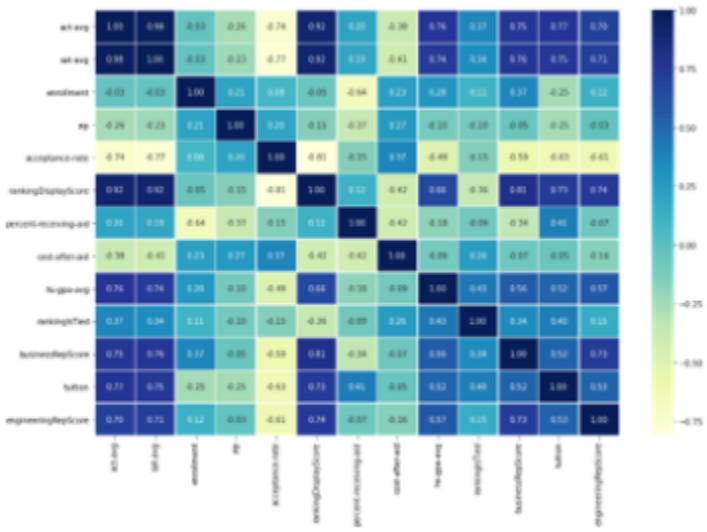
We then investigated whether the total enrollment of a university is related to its ranking. In Figure 7, we can clearly see that public universities have larger enrollments than private universities. However, it doesn't seem like there is any relationship between the enrollment and ranking within either of those groups.

Figure 8



We then investigated whether there was a relationship between the population of a university's area and the ranking of that university. Our biggest takeaway from this analysis was that, on average, public universities are located in much less densely populated areas than private universities.

Figure 9

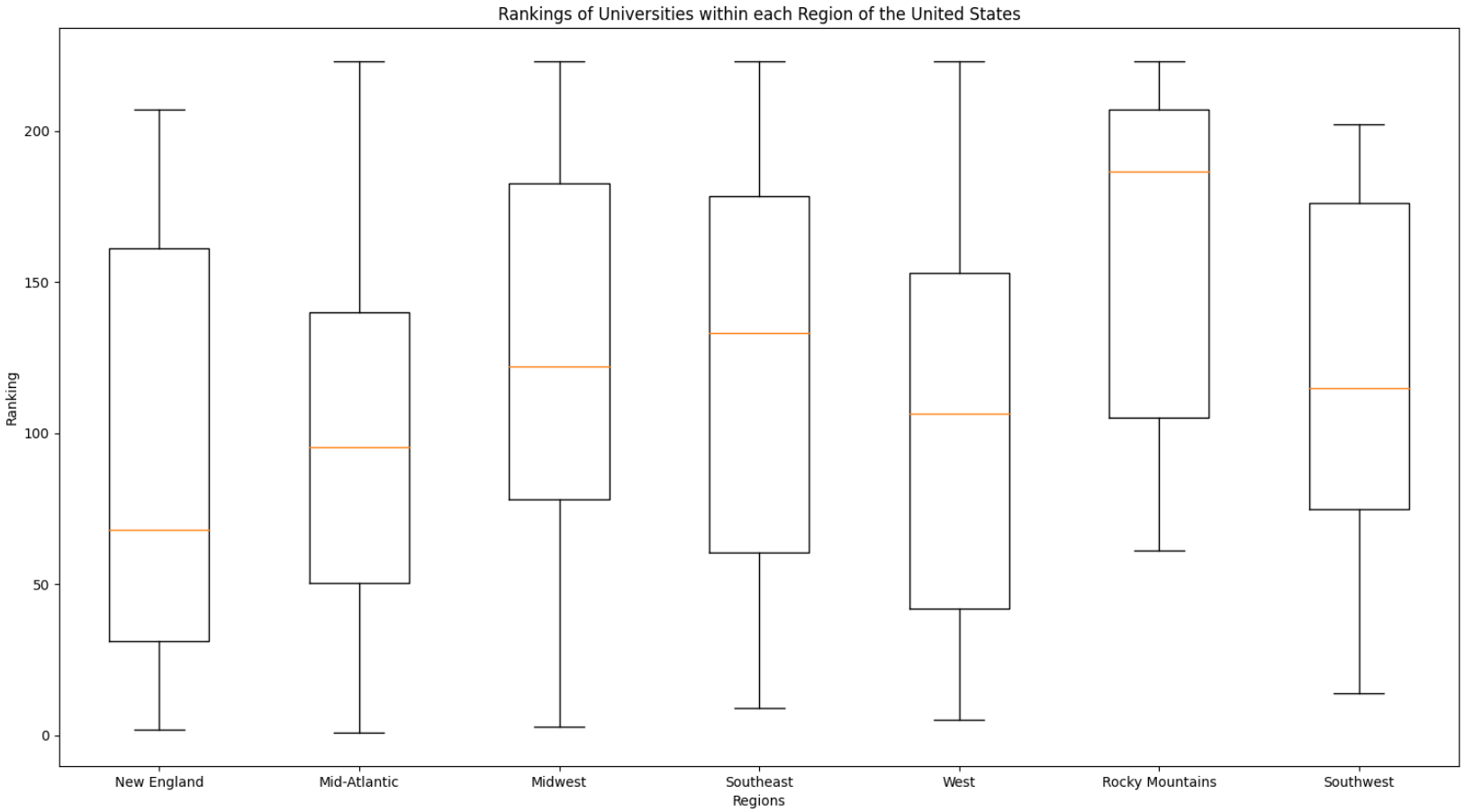


Finally, we created a correlation matrix of all of the variables in the *Universities* dataset. This was to help us see if there are any other relationships worth potentially exploring as we began our focused analysis of the relationship between region and ranking.

Results

Now that we had explored the data a bit, we began our investigation. To help us determine whether there was a relationship between region and average university ranking, we created a box plot (Figure 10) from the rankings of every university in a given region.

Figure 10



Visually, the above box plots seem to suggest that there is a relationship between region and university ranking. There is significant variation in the mean ranking of each region, and the Rocky Mountain region has a mean which falls outside of the IQR of

every other region. However, a hunch based on a visual observation is not sufficient to conclude that such a relationship exists. Thus, we investigated these data further.

If region and university ranking are *truly* independent of each other, then we should be able to treat each region's group of universities as a random sample of the total population of universities. From each of these clusters, we constructed a 90% confidence interval regarding the population mean (Figure 11). We know that the mean ranking of the entire dataset is 112.864. Therefore, if region and ranking are independent, we would expect this value to fall within the confidence interval for each region's rankings. However, this value actually falls outside of the confidence intervals for the New England, Mid-Atlantic, and Rocky Mountain regions. This strongly suggests that region and university ranking are *not* independent.

Figure 11

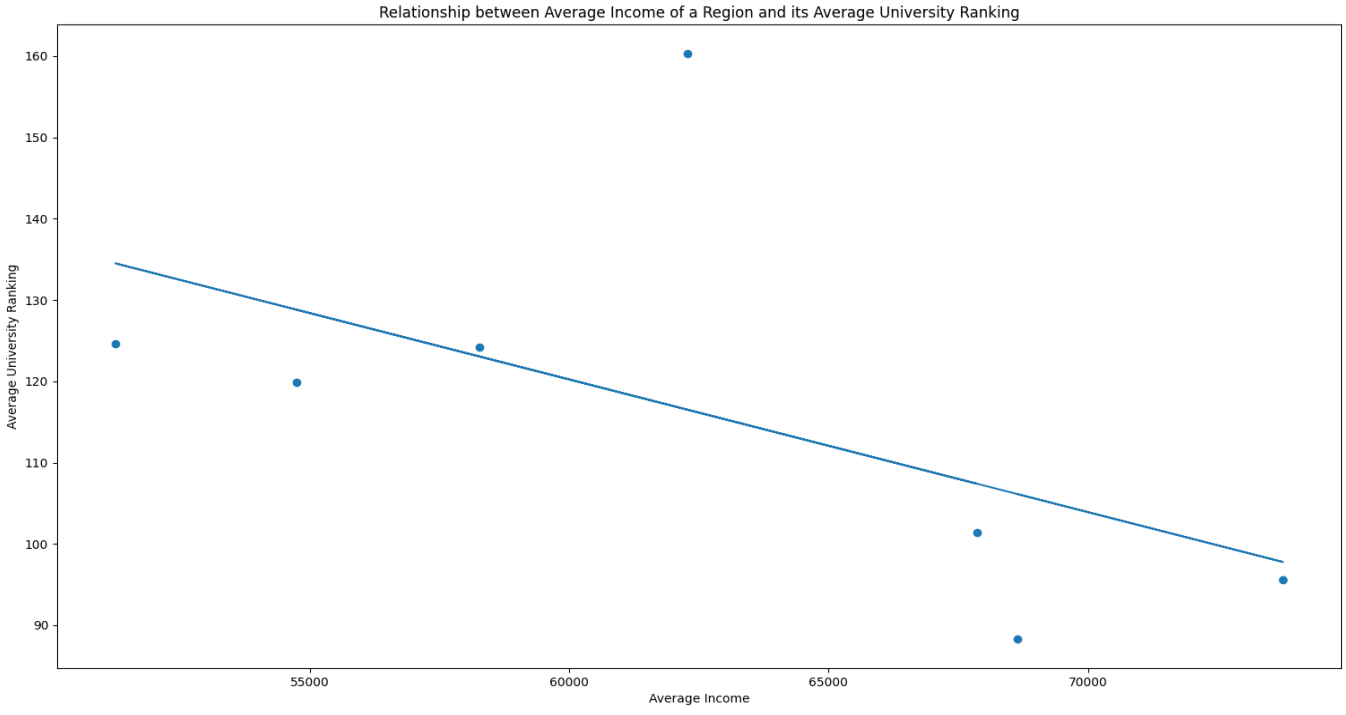
Region	Sample Mean	Lower Bound	Upper Bound
New England	88.333	64.109	112.557
Mid-Atlantic	95.571	83.033	108.110
Midwest	124.188	108.835	139.540
Southeast	124.615	106.476	142.755
West Coast	101.357	80.805	121.909
Rocky Mountains	160.313	136.347	184.278
Southwest	119.824	96.414	143.233

Specifically, New England, the Mid-Atlantic, and (to a lesser extent) the West Coast seem to have better-than-average universities, and the Rocky Mountains seem to have subpar universities.

Now that we had established that there was a relationship between geographical location and university rankings, we could begin to investigate which variables could potentially explain this finding.

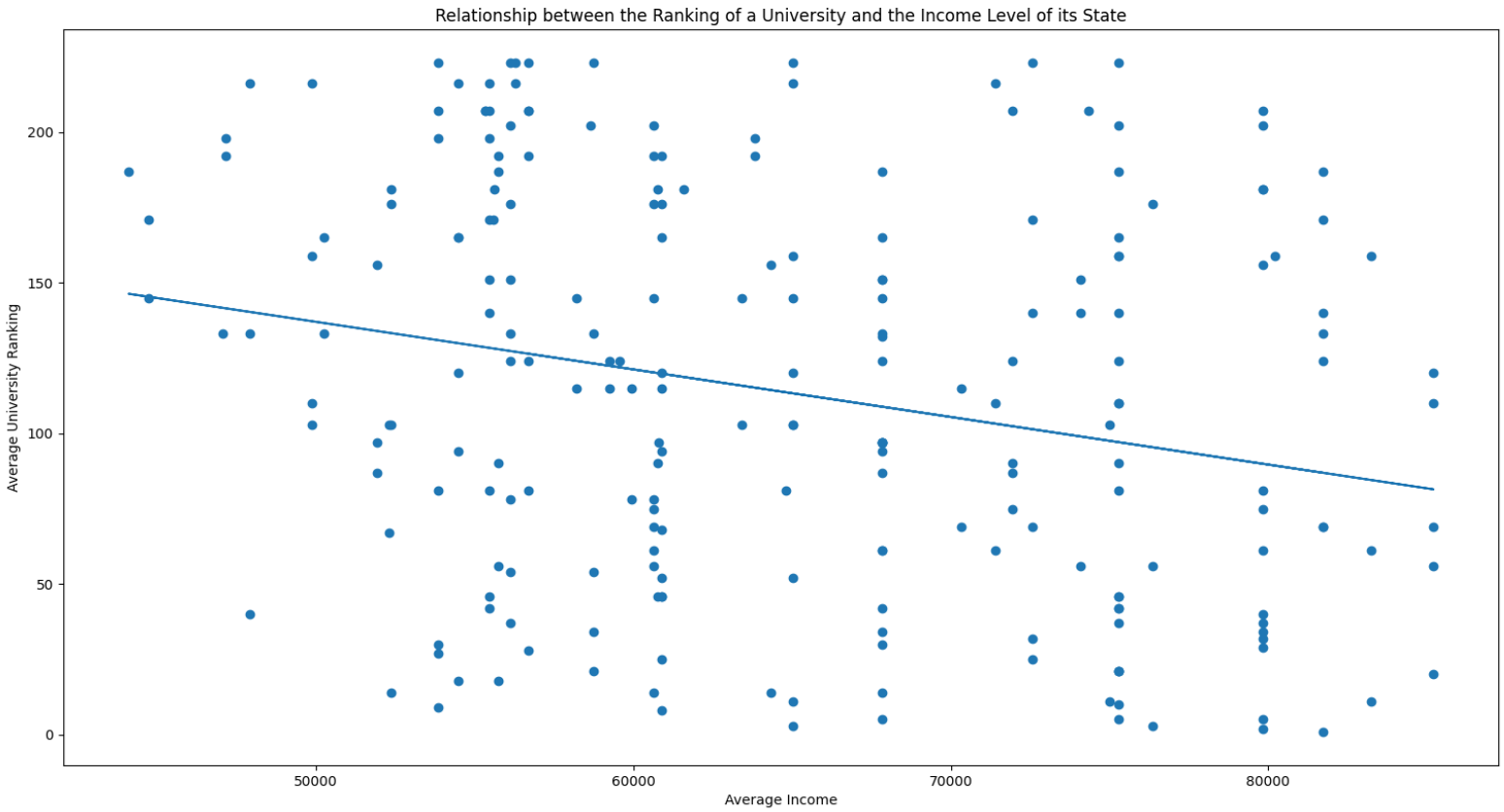
The first factor we decided to investigate was income level. We hypothesized that wealthier areas of the country may have higher rated universities than poorer areas. To test this, we first imported another dataset, containing the average household income in each state. We used these data to find the average income in each of our seven regions, then plotted these values on the average university ranking for each region. We then ran linear regression on this plot (results shown in Figure 12).

Figure 12



Based on the linear regression model, it certainly looks like there is a negative correlation between average university ranking and income level. Since having a lower average ranking is better (1 is higher rated than 2), this seems to indicate that being of higher income is correlated with having better universities. The correlation coefficient of -0.55, however, was concerning. That value is a bit ambiguous. There could be a significant correlation there, or could not. To investigate this further, we made another plot; this time, rather than grouping by region, we plotted each individual university's rank on its state's income level. We calculated a new correlation coefficient, and ran linear regression again. The resulting plot is shown in Figure 13.

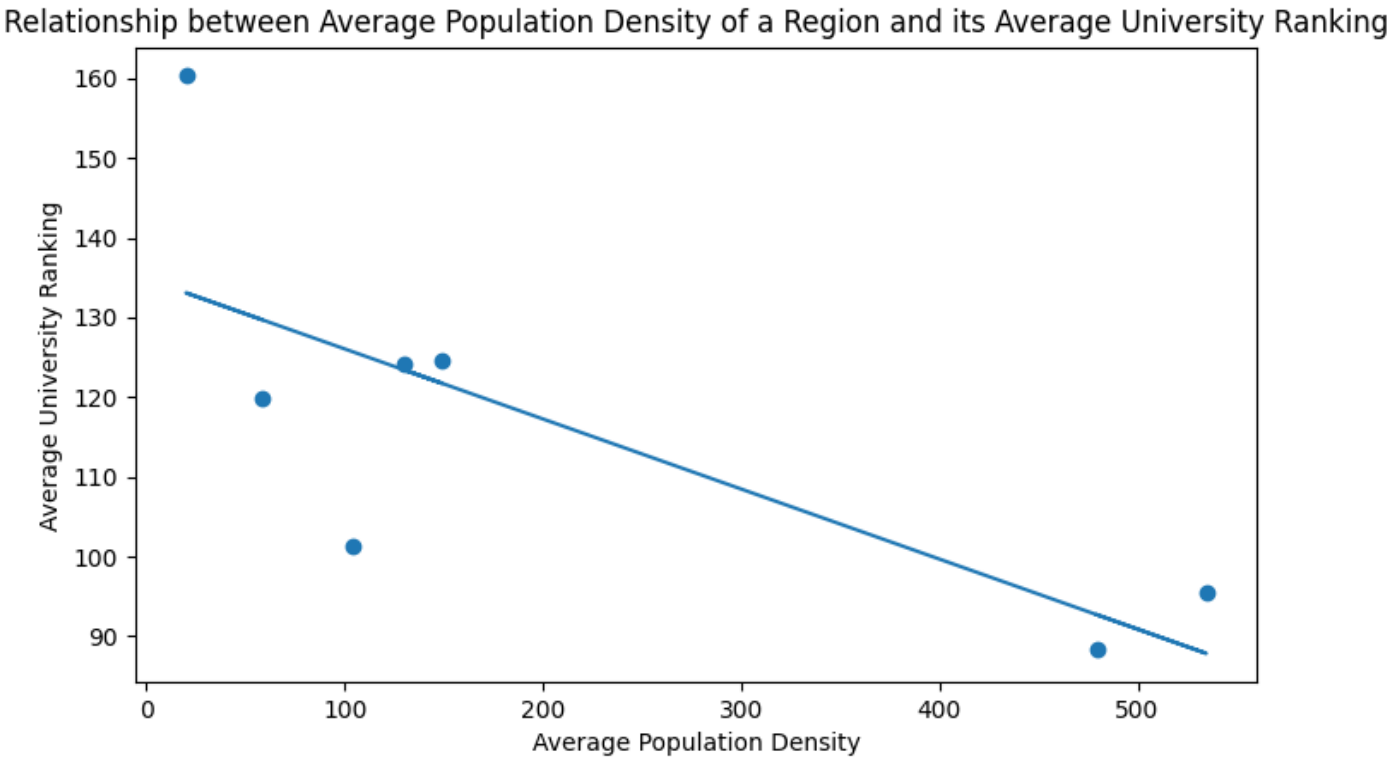
Figure 13



By plotting every university, we get a much clearer picture of this relationship. Visually, it looks like there is no pattern in the data; the points are roughly evenly distributed. The linear regression model still has a slight negative slope; however, the correlation coefficient has fallen all the way to -0.25. It seems that there is a *small* relationship between the income level of a state and the rankings of the universities within it, but it is not a large enough correlation to be worth considering.

Next, we repeated the above process, but with a different variable: population density. We hypothesized that, perhaps, the best universities tend to be located in urban areas rather than rural areas. The results of that analysis can be seen in Figure 14.

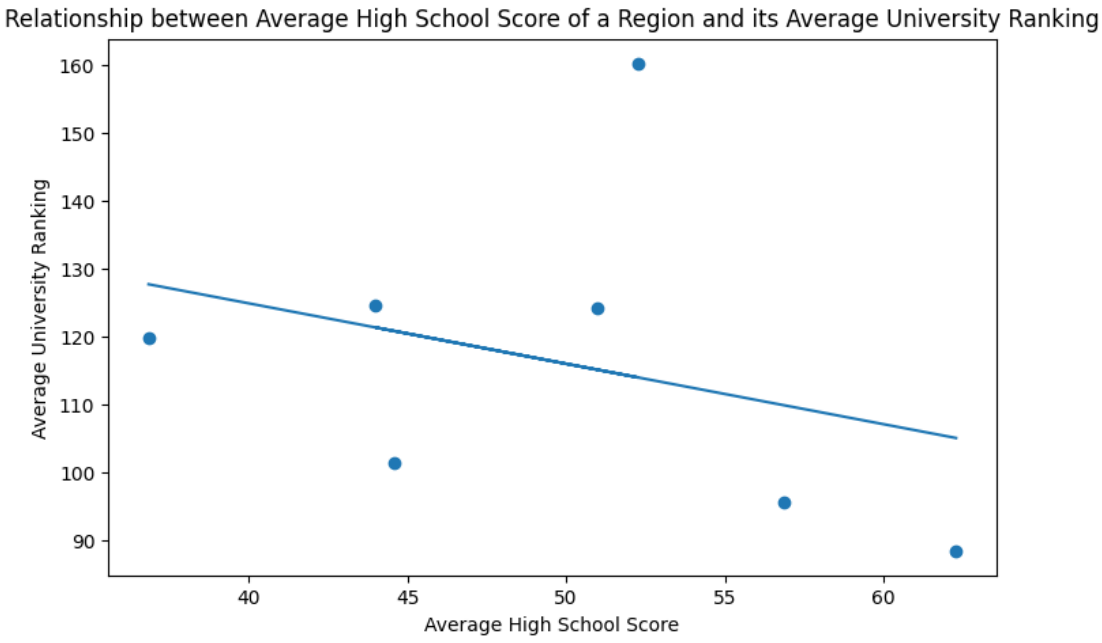
Figure 14



Clearly, the average population density of a region visually appears to have a strong negative correlation with the rankings of universities in that region. The correlation coefficient (-0.75) supports this observation. In other words: it appears that, given a region has a higher population density, we expect it to have higher-ranked universities.

We wanted to consider at least one more factor, so we continued to brainstorm which other regional attributes may be related to the rankings of universities. We realized that, while some students travel far away for school, many others may prioritize looking at colleges closer to home. As such, it seems like many colleges may be disproportionately attended by students who grew up in the area, and, therefore, there may be a relationship between the quality of the *high schools* in an area and the quality of its colleges. We discovered a dataset in which a team of experts scored and ranked each state's public high school system based on several metrics. Much as we had done with the income and population density datasets, we imported the data into our Python script, took the average score of each region, and plotted these averages with their college ranking counterparts (Figure 15).

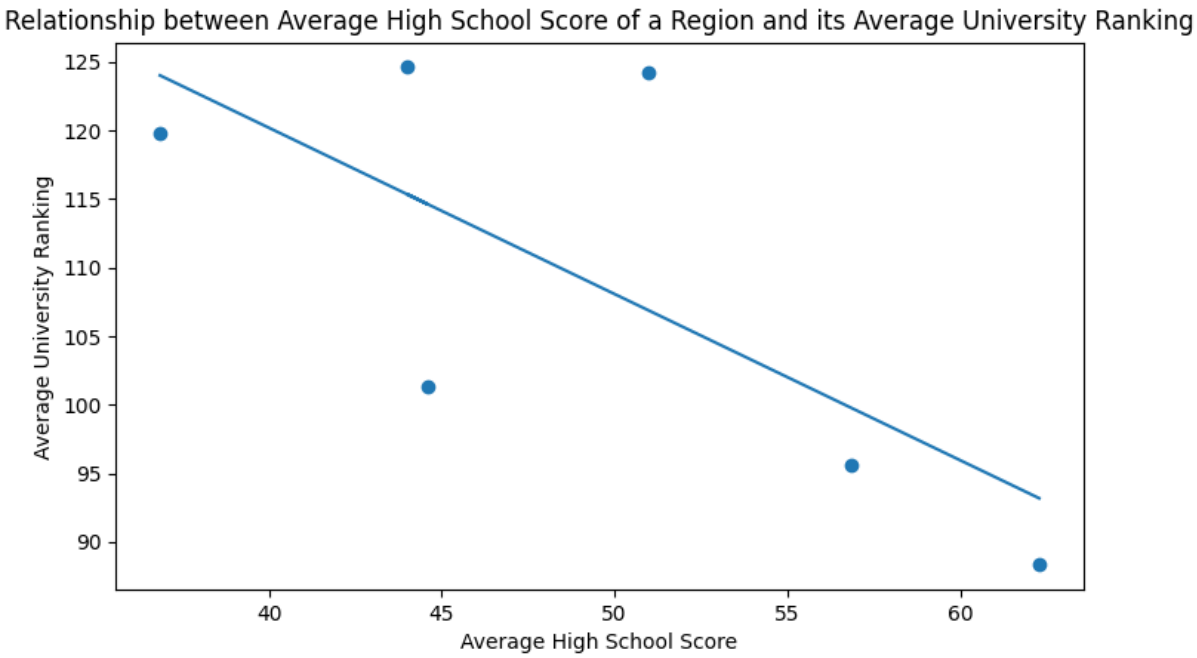
Figure 15



Initially, this seemed like a rather lackluster result. Visually, the relationship looks pretty weak, and the low correlation coefficient (-0.316) seemed to confirm this.

However, we then noticed something about the graph: there appears to be a major outlier. In the very top of the graph, the point corresponding with the Rocky Mountain region is off on its own. We thought it looked like the rest of the points were much more correlated. To test this, we re-ran the plot and regression, but with the Rocky Mountain region removed. The result is shown in Figure 16.

Figure 16



With the outlier removed, these two factors appear very obviously correlated. And, with a correlation coefficient of -0.712, the math seems to agree with this observation.

At this point, a clear picture seemed to be forming. Different regions of the United States *do* have differently ranked universities, with some (New England, Mid-Atlantic) having much higher averages than others (especially the Rocky Mountains). The average household income of a region doesn't seem to have a strong correlation with the rankings of universities in that region; however, it seems like the population density of a region and the quality of the high schools in a region *do* have a strong relationship with the rankings of its universities. Specifically, it seems like places with either higher population density have higher-ranked universities, and places with higher quality high schools have higher-ranked universities.

Further Work

This dataset, while extremely detailed, focused only on the top 200-300 universities in the United States, a nation which is home to over 5,000 of them. If we had data on every single university, the increased sample size would have helped us to be much more precise. Specifically, we could have gotten much more accurate averages for each region. Additionally, this would have allowed us to use smaller regions, or not use regions at all (just look at each state individually). With the dataset we had, we needed to cluster many states with similar qualities together; otherwise, we would have been drawing conclusions from very small groups of universities. Having a dataset with every university would eliminate this problem; thus, we could have smaller regions. This would be beneficial for two reasons: 1, smaller areas would have less variation within them, and 2, having a higher number of regions would cause us to have enough data points to make a predictive model. We looked into using a region's factors (population density, high school scores, etc.) to build a model which would predict what the average university rank of the region would be. However, 8 regions is way too small of a dataset with which to create a model. If we had enough universities to examine each state (or even each *city*) independently, we would likely have enough information to build this model.

It would also be useful to investigate more variables which could potentially be impacting the university rankings of a region. We examined three of them (average household income, population density, high school quality) but there are countless

others which could be useful to look at. If we were to find other significant factors, we could incorporate those findings into the model, thus making it more accurate.

References

Damarla, Rishi. “Data about 500 US Companies.” *Kaggle*, 20 Sept. 2020, https://www.kaggle.com/rishidamarla/data-about-500-us-companies.

Lambert, Christopher. “University Statistics.” *Kaggle*, 21 Jan. 2018, https://www.kaggle.com/theriley106/university-statistics.

“List of U.S. States and Territories by Income.” *Wikipedia*, Wikimedia Foundation, 16 Nov. 2021, https://en.wikipedia.org/wiki/List_of_U.S._states_and_territories_by_income.

“List of U.S. States by Population Density.” *Wikipedia*, Wikimedia Foundation, 13 May 2021, https://simple.wikipedia.org/wiki/List_of_U.S._states_by_population_density.

McCann, Adam. “2021's States with the Best & Worst School Systems.” *WalletHub*, 26 July 2021, https://wallethub.com/edu/e/states-with-the-best-schools/5335.

“United States Cities Database.” *Simplemaps*, 13 Aug. 2021, https://simplemaps.com/data/us-cities.

In [ ]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
df= pd.read_json("schoolInfo.json")
dframe= pd.read_csv("income.csv")
dframe=pd.DataFrame(dframe)
df=pd.DataFrame(df)
df.head(5)
df.shape

for i, row in df.iterrows():
    answer=df.iat[i,16]
    if answer in ["ME","MA","CT","NH","RI","VT"]:
     df.iat[i,17]=1
    if answer in ["NY","NJ","PA","MD","DE","VA","DC"]:
     df.iat[i,17]=2
    if answer in ["WV","OH","WI","MI","MN","IL","IN","IA","NE","KS","MO"]:
     df.iat[i,17]=3
    if answer in ["AR","LA","MS","AL","TN","GA","NC", "SC", "FL","KY"]:
     df.iat[i,17]=4
    if answer in ["WA","OR","CA","NV"]:
     df.iat[i,17]=5
    if answer in ["UT","CO","WY","ID","MT","ND","SD"]:
     df.iat[i,17]=6
    if answer in ["AZ","NM","TX","OK"]:
     df.iat[i,17]=7
    if answer in ["AK","HI"]:
     df.iat[i,17]=8

ne=0
ma=0
mw=0
se=0
wc=0
rm=0
sw=0
o=0

for i, row in df.iterrows():
    answer=df.iat[i,0]
    if answer in [1]:
     ne=df.iat[i,1]+ne
    if answer in [2]:
     ma=df.iat[i,1]+ma
    if answer in [3]:
     mw=df.iat[i,1]+mw
    if answer in [4]:
     se=df.iat[i,1]+se
    if answer in [5]:
     wc=df.iat[i,1]+wc
    if answer in [6]:
     rm=df.iat[i,1]+rm
    if answer in [7]:
     sw=df.iat[i,1]+sw
    if answer in [8]:
     o=df.iat[i,1]+o

income=[a, b, c, d, e, f, g, h]
print(income)

x = [1,2,3,4,5,6,7,8]
y = income
# plotting
plt.title("Line graph")
plt.xlabel("X axis")
plt.ylabel("Y axis")
plt.plot(x, y, color ="green")
plt.show()

a=0
b=0
c=0
d=0
e=0
f=0
g=0
h=0

for i, row in df.iterrows():
    answer=df.iat[i,16]
    if answer in ["ME","MA","CT","NH","RI","VT"]:
     a=df.iat[i,27]+a
    if answer in ["NY","NJ","PA","MD","DE","VA","DC"]:
     b=df.iat[i,27]+b
    if answer in ["WV","OH","WI","MI","MN","IL","IN","IA","NE","KS","MO"]:
     c=df.iat[i,27]+c
    if answer in ["AR","LA","MS","AL","TN","GA","NC", "SC", "FL","KY"]:
```

```
     d=df.iat[i,27]+d
    if answer in ["WA","OR","CA","NV"]:
     e=df.iat[i,27]+e
    if answer in ["UT","CO","WY","ID","MT","ND","SD"]:
     f=df.iat[i,27]+f
    if answer in ["AZ","NM","TX","OK"]:
     g=df.iat[i,27]+g
    if answer in ["AK","HI"]:
     h=df.iat[i,27]+h

tuition=[a, b, c, d, e, f, g, h]
print(tuition)

x = [1,2,3,4,5,6,7,8]
y = tuition
# plotting
plt.title("Line graph")
plt.xlabel("X axis")
plt.ylabel("Y axis")
plt.plot(x, y, color ="green")
plt.show()

income = [699720.0, 804552.0, 1026903.0, 838818.0, 639321.0, 810032.0, 582343.0, 508425.0]

inc=np.array(income)/sum(income)
tui=np.array(tuition)/sum(tuition)
plt.title("Line graph")
plt.xlabel("X axis")
plt.ylabel("Y axis")
plt.plot(inc, tui, color ="green")
plt.show()

print(np.correlate(tui, inc))
plt.bar(x, income, 0.8)
plt.bar(x, tuition, 0.8)
x=np.array(income)/8
y=np.array(tuition)/8
a, b = np.polyfit(x, y, 1)

plt.scatter(x, y)
x=np.array(x)
#add line of best fit to plot
plt.plot(x, a*x+b)
plt.title('Relationship Between Average Income and Average Tuition of the Regions')
plt.xlabel('Average Income')
plt.ylabel('Average Tuition')

names=['New England', 'Mid-Atlantic' , 'Midwest' , 'Southeast' , 'West Coast', 'Rocky Mountains', 'Southeast' , '
print(names)
plt.plot(x, a*x+b)

print(tui)
qq=0
rr=0

for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["ME","MA","CT","NH","RI","VT"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

ne=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["NY","NJ","PA","MD","DE","VA","DC"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

ma=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["WV","OH","WI","MI","MN","IL","IN","IA","NE","KS","MO"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1
```

```
mw=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["AR","LA","MS","AL","TN","GA","NC", "SC", "FL","KY"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

se=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["WA","OR","CA","NV"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

wc=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["UT","CO","WY","ID","MT","ND","SD"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

rm=rr/qq

qq=0
rr=0
for i, row in df.iterrows():
    answer=df.iat[i,16]
    ans=df.iat[i,35]
    if answer in ["AZ","NM","TX","OK"]:
        if ans=='private':
            qq=qq+1
        else:
            rr=rr+1

sw=rr/qq


ratios = [ne, ma, mw, se, wc, rm, sw]
type(ratios)
print(ratios)

x = [1,2,3,4,5,6,7]
plt.bar(x, ratios, 0.8)

x = [1,2,3,4,5,6,7,8]
plt.bar(x, inc, 0.8)

x=np.array(income)
y=np.array(ratios)
x = x[:-1]; x
# y = y[:-1]; y
a, b = np.polyfit(x, y, 1)

plt.scatter(x, y)
x=np.array(x)
#add line of best fit to plot
plt.plot(x, a*x+b)
plt.title('Relationship Between Average Income and Ratio of Public and Private Schools of the Regions')
plt.xlabel('Average Income')
plt.ylabel('Ratio of Private and Public Schools')
```

In [ ]:

```
import os
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

df = pd.read_json("schoolInfo.json")
df=pd.DataFrame(df)
print('This dataset contains {} rows and {} columns'.format(df.shape[0],df.shape[1]))
# dataframe information
df.info(max_cols = len(df))
#Check for the missing values

df.isna().any()
# drop columns which have only NaN values
df.dropna(axis=1, thresh=1, inplace=True)
# drop columns with only one distinct value
for col in df.columns:
    if len(df[col].unique()) == 1:
        df.drop(col,inplace=True,axis=1)

# drop non relevant columns or columns which semantically duplicate other columns in the DataFrame
df.drop(['primaryPhoto', 'primaryPhotoThumb', 'sortName', 'urlName', 'aliasNames', 'nonResponderText', 'nonRespor

# number of unique cities
df['city'].nunique()

# dataframe information
df.info(max_cols = len(df))
df['city'].value_counts().head(10)
df['state'].value_counts().head(10)

# create a new dataframe with the number of universities in each state and plot the graph
df_count = df['state'].value_counts().rename_axis('State').reset_index(name='Number of Universities')

df_count_to_plot = df_count

df_count_to_plot["State"] = df_count["State"]
df_count_to_plot["Number of Universities"] = df_count["Number of Universities"]

import matplotlib.pyplot as plt

plt.rcParams["figure.figsize"] = (18, 5)
df_count_to_plot.plot.bar(x='State', rot=45, color=(0.1, 0.1, 0.1, 0.1),  edgecolor='blue')

# move the column with university names to the front so the table gets more readable
df = df[ ['displayName'] + [ col for col in df.columns if col != 'displayName' ] ]

df1= pd.read_csv("uscities.csv")
df1.columns
# we see that in the 'city_ascii' column cities have the more universal spelling
df1.loc[(df1['city'] != df1['city_ascii'])]
df1.drop(['city'], axis=1, inplace=True)
df1.rename(columns={"city_ascii": "city"}, inplace=True)
# merge two tables on city names, we want citites from the 'university-statistics' table
# to have longitude and latitude data which we extract from the US Cities Database table
df_merged = pd.merge(df, df1, on='city')
df_merged
df_merged = df_merged.loc[(df_merged['state'] == df_merged['state_id'])]
df2=pd.read_csv("us_companies.csv")
# merge two tables on city names, we want citites from the 'university-statistics' table
# to have longitude and latitude data which we extract from the US Cities Database table
df_merged1 = pd.merge(df_merged, df2, on='city')
df_merged1
df_merged1.columns
# drop non relevant columns or columns which semantically duplicate other columns in the DataFrame
df_merged1.drop(['url', 'year_founded','description', 'description_short', 'source_count', 'data_types',
       'example_uses', 'data_impacts','zip_code','last_updated'''], axis=1, inplace=True)
df_merged1.columns
df_merged1['state_y'].value_counts()
# create a new dataframe with the number of companies in each state and plot the graph

df_count1 = df_merged1['state_y'].value_counts().rename_axis('State').reset_index(name='Number of Companies')

df_count_to_plot = df_count1

df_count_to_plot["State"] = df_count1["State"]
df_count_to_plot["Number of Companies"] = df_count1["Number of Companies"]

import matplotlib.pyplot as plt

plt.rcParams["figure.figsize"] = (18, 5)
df_count_to_plot.plot.bar(x='State', rot=45, color=(0.1, 0.1, 0.1, 0.1),  edgecolor='blue')
import geopandas as gpd
import math
```

```
import folium
from folium import Choropleth, Circle, Marker
from folium.plugins import HeatMap, MarkerCluster
# Create a map
m_1 = folium.Map(location=[42.32,-81.0589], tiles='openstreetmap', zoom_start=3)

# Add points to the map
for idx, row in df_merged.iterrows():
    Marker([row['lat'], row['lng']]).add_to(m_1)

# Display the map
m_1
# a new DataFrame with the top univiersities by ranking
df_top = df_merged.loc[df_merged['rankingDisplayScore'] >90]
# show a map with the top universities by ranking
m_2 = folium.Map(location=[42.32,-81.0589], tiles='openstreetmap', zoom_start=3)

for idx, row in df_top.iterrows():
    Marker([row['lat'], row['lng']]).add_to(m_2)

m_2
sns.scatterplot(data=df, x="rankingDisplayRank", y="acceptance-rate", hue="institutionalControl")
sns.lineplot(x="rankingDisplayRank", y="acceptance-rate",
             hue="institutionalControl",
             data=df)
sns.lineplot(x="rankingDisplayRank", y='tuition',
             hue="institutionalControl",
             data=df)
sns.lmplot(y='tuition', x='rankingDisplayRank', hue="institutionalControl", data=df)
sns.lmplot(x='tuition', y='cost-after-aid', data=df)

sns.lineplot(x="rankingDisplayRank", y="enrollment",
             hue="institutionalControl",
             data=df)

sns.lmplot(x="rankingDisplayRank", y="enrollment",
             data=df)
sns.lineplot(x="rankingDisplayRank", y="population",
             hue="institutionalControl",
             data=df_merged1)
# Find the correlation between our independent variables
corr_matrix = df.corr()
corr_matrix

# Let's make it look a little prettier
corr_matrix = df.corr()
plt.figure(figsize=(15, 10))
sns.heatmap(corr_matrix,
            annot=True,
            linewidths=0.5,
            fmt= ".2f",
            cmap="YlGnBu");
```

```
import csv
import matplotlib.pyplot as plt
import scipy.stats as st
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures

if __name__ == '__main__':
      # Break the United States into regions based on similar
characteristics.
      NEW_ENGLAND = ["ME", "MA", "CT", "NH", "RI", "VT"]
      MID_ATLANTIC = ["NY", "NJ", "MD", "DE", "VA", "DC", "PA"]
      MIDWEST = ["WV", "OH", "WI", "MI", "MN", "IL", "IN", "IA", "NE",
"KS", "MO"]
      SOUTHEAST = ["AR", "LA", "MS", "AL", "TN", "GA", "NC", "SC", "FL",
"KY"]
      WEST = ["WA", "OR", "CA", "NV"]
      ROCKY_MOUNTAINS = ["UT", "CO", "WY", "ID", "MT", "ND", "SD"]
      SOUTHWEST = ["AZ", "NM", "TX", "OK"]

      # Import the Universities data set.
      univ_csv = open("universities.csv", "r")
      univ_dict = csv.DictReader(univ_csv)

      # Assign each university to a region, based on the state it's in.
      univ_list = []
      for school in univ_dict:
            if school["state"] in NEW_ENGLAND:
                  school["region"] = "New England"
            elif school["state"] in MID_ATLANTIC:
                  school["region"] = "Mid-Atlantic"
            elif school["state"] in MIDWEST:
                  school["region"] = "Midwest"
            elif school["state"]  in SOUTHEAST:
                  school["region"] = "Southeast"
            elif school["state"] in WEST:
                  school["region"] = "West"
            elif school["state"] in ROCKY_MOUNTAINS:
                  school["region"] = "Rocky Mountains"
            elif school["state"] in SOUTHWEST:
                  school["region"] = "Southwest"

            # Check for whether a University is actually ranked. Discard
universities that aren't.
            if int(school["rankingSortRank"]) >= 1:
                  univ_list.append(school)

      # Retrieve the rankings of each university in a given region.
      ne_rank = []
      ma_rank = []
      mw_rank = []
      se_rank = []
      we_rank = []
```

```
      rm_rank = []
      sw_rank = []
      all_rank = []

      for school in univ_list:
            if school["region"] == "New England":
                  all_rank.append(int(school["rankingSortRank"]))
                  ne_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "Mid-Atlantic":
                  all_rank.append(int(school["rankingSortRank"]))
                  ma_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "Midwest":
                  all_rank.append(int(school["rankingSortRank"]))
                  mw_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "Southeast":
                  all_rank.append(int(school["rankingSortRank"]))
                  se_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "West":
                  all_rank.append(int(school["rankingSortRank"]))
                  we_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "Rocky Mountains":
                  all_rank.append(int(school["rankingSortRank"]))
                  rm_rank.append(int(school["rankingSortRank"]))
            elif school["region"] == "Southwest":
                  all_rank.append(int(school["rankingSortRank"]))
                  sw_rank.append(int(school["rankingSortRank"]))

      # Build boxplots for each region to visualize these rankings.
      plt.boxplot([ne_rank, ma_rank, mw_rank, se_rank, we_rank, rm_rank,
sw_rank], labels = ["New England", "Mid-Atlantic", "Midwest",
"Southeast", "West", "Rocky Mountains", "Southwest"], vert = True)
      plt.title("Rankings of Universities within each Region of the
United States")
      plt.xlabel("Regions")
      plt.ylabel("Ranking")

      plt.show()

      # Find the mean ranking of each region.
      def avg(ranks):
            return(sum(ranks) / len(ranks))

      means = [avg(ne_rank), avg(ma_rank), avg(mw_rank), avg(se_rank),
avg(we_rank), avg(rm_rank), avg(sw_rank)]
      print(means)


      # Import the Average Household Income by Region dataset.
      income_csv = open("income.csv", "r")
      income_dict = csv.DictReader(income_csv)
      income_list = []

      for state in income_dict:
            income_list.append(state)
```

```
      ne_inc = []
      ma_inc = []
      mw_inc = []
      se_inc = []
      we_inc = []
      rm_inc = []
      sw_inc = []

      for state in income_list:
            if state["state"] in NEW_ENGLAND:
                  ne_inc.append(int(state["income"]))
            elif state["state"] in MID_ATLANTIC:
                  ma_inc.append(int(state["income"]))
            elif state["state"] in MIDWEST:
                  mw_inc.append(int(state["income"]))
            elif state["state"]  in SOUTHEAST:
                  se_inc.append(int(state["income"]))
            elif state["state"] in WEST:
                  we_inc.append(int(state["income"]))
            elif state["state"] in ROCKY_MOUNTAINS:
                  rm_inc.append(int(state["income"]))
            elif state["state"] in SOUTHWEST:
                  sw_inc.append(int(state["income"]))

      # Compute the mean income for each region.
      inc_means = [avg(ne_inc), avg(ma_inc), avg(mw_inc), avg(se_inc),
avg(we_inc), avg(rm_inc), avg(sw_inc)]
      print(inc_means)

      # Plot income on ranking and look for a correlation.
      (m, b) = np.polyfit(inc_means, means, 1)
      print(m, b)
      reg_line = np.polyval([m, b], inc_means)
      plt.plot(inc_means, reg_line)

      plt.title("Relationship between Average Income of a Region and its
Average University Ranking")
      plt.xlabel("Average Income")
      plt.ylabel("Average University Ranking")
      plt.scatter(inc_means, means)
      plt.show()

      print(np.corrcoef(inc_means, means))

      # Plot income on ranking by state, not region.
      state_inc = []
      school_rank = []
      for school in univ_list:
            for state in income_list:
                  if school["state"] == state["state"]:
                        state_inc.append(int(state["income"]))
            school_rank.append(int(school["rankingSortRank"]))
```

```
      (m, b) = np.polyfit(state_inc, school_rank, 1)
      print(m, b)
      reg_line = np.polyval([m, b], state_inc)
      plt.plot(state_inc, reg_line)

      plt.title("Relationship between the Ranking of a University and the
Income Level of its State")
      plt.xlabel("Average Income")
      plt.ylabel("Average University Ranking")
      plt.scatter(state_inc, school_rank)
      plt.show()

      print(np.corrcoef(state_inc, school_rank))

      # Construct a 95% CI for each region.
      # Use above to conclude that region and ranking are not
independent.
      # Investigate the factors that cause this. Population density?
      # Build a logistic regression model using the significant factors
to classify a university based only on its region's features.
      print(st.norm.interval(alpha = .9, loc = avg(ne_rank), scale =
st.sem(ne_rank)))
      print(avg(ne_rank))
      print(st.norm.interval(alpha = .9, loc = avg(ma_rank), scale =
st.sem(ma_rank)))
      print(avg(ma_rank))
      print(st.norm.interval(alpha = .9, loc = avg(mw_rank), scale =
st.sem(mw_rank)))
      print(avg(mw_rank))
      print(st.norm.interval(alpha = .9, loc = avg(se_rank), scale =
st.sem(se_rank)))
      print(avg(se_rank))
      print(st.norm.interval(alpha = .9, loc = avg(we_rank), scale =
st.sem(we_rank)))
      print(avg(we_rank))
      print(st.norm.interval(alpha = .9, loc = avg(rm_rank), scale =
st.sem(rm_rank)))
      print(avg(rm_rank))
      print(st.norm.interval(alpha = .9, loc = avg(sw_rank), scale =
st.sem(sw_rank)))
      print(avg(sw_rank))

      print(avg(all_rank))

      # Import the Population Density dataset.
      density_csv = open("density.csv", "r")
      density_dict = csv.DictReader(density_csv)
      density_list = []

      for state in density_dict:
            density_list.append(state)

      ne_den = []
      ma_den = []
```

```
      mw_den = []
      se_den = []
      we_den = []
      rm_den = []
      sw_den = []

      for state in density_list:
            if state["state"] in NEW_ENGLAND:
                  ne_den.append(float(state["density"]))
            elif state["state"] in MID_ATLANTIC:
                  ma_den.append(float(state["density"]))
            elif state["state"] in MIDWEST:
                  mw_den.append(float(state["density"]))
            elif state["state"]  in SOUTHEAST:
                  se_den.append(float(state["density"]))
            elif state["state"] in WEST:
                  we_den.append(float(state["density"]))
            elif state["state"] in ROCKY_MOUNTAINS:
                  rm_den.append(float(state["density"]))
            elif state["state"] in SOUTHWEST:
                  sw_den.append(float(state["density"]))

        # Compute the mean population density for each region.
      den_means = [avg(ne_den), avg(ma_den), avg(mw_den), avg(se_den),
avg(we_den), avg(rm_den), avg(sw_den)]
      print(den_means)

      (m, b) = np.polyfit(den_means, means, 1)
      print(m, b)
      reg_line = np.polyval([m, b], den_means)
      plt.plot(den_means, reg_line)

      plt.title("Relationship between Average Population Density of a
Region and its Average University Ranking")
      plt.xlabel("Average Population Density")
      plt.ylabel("Average University Ranking")
      plt.scatter(den_means, means)
      plt.show()

      print(np.corrcoef(den_means, means))

      hs_csv = open("hsrankings.csv", "r")
      hs_dict = csv.DictReader(hs_csv)
      hs_list = []

      for state in hs_dict:
            hs_list.append(state)

      ne_hs = []
      ma_hs = []
      mw_hs = []
      se_hs = []
      we_hs = []
      rm_hs = []
```

```
      sw_hs = []

      for state in hs_list:
            if state["state"] in NEW_ENGLAND:
                  ne_hs.append(float(state["score"]))
            elif state["state"] in MID_ATLANTIC:
                  ma_hs.append(float(state["score"]))
            elif state["state"] in MIDWEST:
                  mw_hs.append(float(state["score"]))
            elif state["state"]  in SOUTHEAST:
                  se_hs.append(float(state["score"]))
            elif state["state"] in WEST:
                  we_hs.append(float(state["score"]))
            elif state["state"] in ROCKY_MOUNTAINS:
                  rm_hs.append(float(state["score"]))
            elif state["state"] in SOUTHWEST:
                  sw_hs.append(float(state["score"]))

        # Compute the mean high school score for each region.
      hs_means = [avg(ne_hs), avg(ma_hs), avg(mw_hs), avg(se_hs),
avg(we_hs), avg(rm_hs), avg(sw_hs)]
      print(hs_means)

      (m, b) = np.polyfit(hs_means, means, 1)
      print(m, b)
      reg_line = np.polyval([m, b], hs_means)
      plt.plot(hs_means, reg_line)

      plt.title("Relationship between Average High School Score of a
Region and its Average University Ranking")
      plt.xlabel("Average High School Score")
      plt.ylabel("Average University Ranking")
      plt.scatter(hs_means, means)
      plt.show()

      print(np.corrcoef(hs_means, means))

      # Removing Rocky Mountain region (outlier)
      means_without_rm = [means[0], means[1], means[2], means[3],
means[4], means[6]]
      hs_means_without_rm = [avg(ne_hs), avg(ma_hs), avg(mw_hs),
avg(se_hs), avg(we_hs), avg(sw_hs)]
      print(hs_means_without_rm)

      (m, b) = np.polyfit(hs_means_without_rm, means_without_rm, 1)
      print(m, b)
      reg_line = np.polyval([m, b], hs_means_without_rm)
      plt.plot(hs_means_without_rm, reg_line)

      plt.title("Relationship between Average High School Score of a
Region and its Average University Ranking")
      plt.xlabel("Average High School Score")
      plt.ylabel("Average University Ranking")
      plt.scatter(hs_means_without_rm, means_without_rm)
```

```
      plt.show()

      print(np.corrcoef(hs_means_without_rm, means_without_rm))

      # Building a model (WE SCRAPPED THIS BECAUSE IT DIDN'T WORK)
      df_list = []
      for i in range (0, 7):
            df_list.append([means[i], den_means[i], hs_means[i]])

      df = pd.DataFrame(df_list, columns = ["rank", "density",
"high_school_score"])

      print(df)

      y = df["rank"]
      x = df[["density", "high_school_score"]]

      model = PolynomialFeatures(degree = 2)
      x_modified = model.fit_transform(x)
      model.fit(x_modified, y)
      model = LinearRegression()
      model.fit(x_modified, y)

      print(model.coef_)
```