Author: Chaojie Gong CMSC320 Final Tutorial

Lots of people think the universities in the United States have a comprehensive good reputation over the world. Therefore, their prioritized option is to pursue a degree in the U.S.

In fact, with over 4,000 colleges and universities, the United States has more institutions of higher learning than any other country in the world. Many of them are highly ranked, offering top-notch educational programs, opportunities for hands-on learning, and cutting-edge research at the graduate and undergraduate levels. Many professors at U.S. institutions have terminal degrees in their field of expertise, are internationally recognized for their scholarship, and represent a diversity of ethnicities and cultural backgrounds. Besides, a significant number of the teaching staff have traveled or lived abroad, which contributes to an enriched classroom experience. Moreover, graduates from a U.S. university or college often find enormous success in the international job market. Employers recognize the value of such an education and the unique skills and qualities that these graduates possess. In short, a degree from a U.S. institution opens doors and is recognized around the world.

The QS World University Rankings comprises of the 150 universities of the top international study destination, US. More than 1.18 million international students were studying in the US in 2017. 77% of these have come from Asia. As per the Institute of International Education’s Open Doors report, the most popular courses are Business and Management, Computer Science, Engineering, and Mathematics. Apart from this, the most popular study destinations for students are New York, Texas, and California.

The main highlight of the US universities is their focus on research-oriented learning. Researchers are always at the forefront and are always look out to develop something new. Innovation and creativity always remain at the core of their educational philosophy. In the US, regular testing/homework and classroom participation is mandatory for getting a good result. Students are encouraged to discuss the issues and focus on providing ideas.

In this project, I will try to analyze the world university ranking data to give me a better idea of how U.S. universities have such a good reputation and the whole picture of top university performance in other countries and their distribution over the world.

College and university rankings are rankings of institutions in higher education which have been ranked on the basis of various combinations of various factors. None of the rankings give a comprehensive overview of the strengths of the institutions ranked because all select a range of easily quantifiable characteristics to base their results on. Rankings have most often been conducted by magazines, newspapers, websites, governments, or academics. In addition to ranking entire institutions, organizations perform rankings of specific programs, departments, and schools. Various rankings consider combinations of measures of funding and endowment, research excellence and influence, specialization expertise, admissions, student options, award numbers, internationalization, graduate employment, industrial linkage, historical reputation and other criteria. Various rankings mostly evaluating on institutional output by research. Some rankings evaluate institutions within a single country, while others assess institutions worldwide.

Every published ranking uses multiple factors. Some factors are arguably less causative and/or less correlated than others. Some rankings rely on publicly available data, while others give weight to surveys and/or comments from students, parents, and admission staff.

Errors or misreporting can happen, which may affect results. A recent book indicates that true shifts in the top 25-30 schools would require significant funds over time, and thus are unlikely to occur. Easily gathered data may not be the most valuable. Arbitrary weighting of specific factors may also skew results.

In here, I will mainly use the data from the Center for World University Rankings (CWUR). CWUR publishes the only global university ranking that measures the quality of education and training of students as well as the prestige of the faculty members and the quality of their research without relying on surveys and university data submissions.

CWUR uses seven objective and robust indicators to rank the world's universities:

1) Quality of Education, measured by the number of a university's alumni who have won major academic distinctions relative to the university's size (25%) 2) Alumni Employment, measured by the number of a university's alumni who have held top executive positions at the world's largest companies relative to the university's size (25%) 3) Quality of Faculty, measured by the number of faculty members who have won major academic distinctions (10%) 4) Research Performance: i) Research Output, measured by the the total number of research papers (10%) ii) High-Quality Publications, measured by the number of research papers appearing in top-tier journals (10%) iii) Influence, measured by the number of research papers appearing in highly-influential journals (10%) iv) Citations, measured by the number of highly-cited research papers (10%)

In [32]:

```
! pip3 install lxml
import pandas as pd

url = "https://cwur.org/2020-21.php"
tables = pd.read_html(url)
df = tables[0]
df
```

```
Requirement already satisfied: lxml in /opt/conda/lib/python3.8/site-packages
(4.6.2)
```

Out[32]:

	World Rank	Institution	Location	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	3	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7
2	3	Stanford University	USA	3	10	4	3	2
3	4	University of Cambridge	United Kingdom	1	1	21	4	11
4	5	University of Oxford	United Kingdom	2	7	26	9	4
...	...	...	...	...	...	...	...	...
1995	1996	Polytechnic Institute of Bragança	Portugal	15	-	-	-	1917
1996	1997	Federal University of Maranhão	Brazil	57	-	-	-	1918
1997	1998	Autonomous University of Baja California	Mexico	20	-	-	-	1921
1998	1999	American University in Cairo	Egypt	18	-	299	-	-
1999	2000	Kyonggi University	South Korea	61	-	-	-	1922

2000 rows × 9 columns

In [2]:

```
# I will focus on the top 10 countries that have the most universities posted
 on the ranking list

df_Country = df["Location"].value_counts()
df_Country.head(10)
```

Out[2]:

```
USA                357
China              267
Japan              126
United Kingdom      95
France              82
Germany             70
Italy               66
India               64
South Korea         61
Brazil              57
Name: Location, dtype: int64
```

From the table, we can clearly see the number of top universities located in each country. Not surprisingly, the U.S. occupied a large proportion of the list. It surpasses the second country which is China by almost 100 units. Also, the table helps me to have a better understanding of how much weight the other countries take up.

In [3]:

```
# Calculate the comprehensive score according to each country
df["Mean"] = "Nah"

mean_USA = df.loc[df["Location"] == "USA", "Score"].sum()
mean_USA /= df_Country[0]
df.loc[df["Location"] == "USA", "Mean"] = mean_USA

mean_China = df.loc[df["Location"] == "China", "Score"].sum()
mean_China /= df_Country[1]
df.loc[df["Location"] == "China", "Mean"] = mean_China

mean_Japan = df.loc[df["Location"] == "Japan", "Score"].sum()
mean_Japan /= df_Country[2]
df.loc[df["Location"] == "Japan", "Mean"] = mean_Japan

mean_UnitedKingdom = df.loc[df["Location"] == "United Kingdom", "Score"].sum()
mean_UnitedKingdom /= df_Country[3]
df.loc[df["Location"] == "United Kingdom", "Mean"] = mean_UnitedKingdom

mean_France = df.loc[df["Location"] == "France", "Score"].sum()
mean_France /= df_Country[4]
df.loc[df["Location"] == "France", "Mean"] = mean_France

mean_Germany = df.loc[df["Location"] == "Germany", "Score"].sum()
mean_Germany /= df_Country[5]
df.loc[df["Location"] == "Germany", "Mean"] = mean_Germany

mean_Italy = df.loc[df["Location"] == "Italy", "Score"].sum()
mean_Italy /= df_Country[6]
df.loc[df["Location"] == "Italy", "Mean"] = mean_Italy

mean_India = df.loc[df["Location"] == "India", "Score"].sum()
mean_India /= df_Country[7]
df.loc[df["Location"] == "India", "Mean"] = mean_India

mean_SouthKorea  = df.loc[df["Location"] == "South Korea", "Score"].sum()
mean_SouthKorea /= df_Country[8]
df.loc[df["Location"] == "South Korea", "Mean"] = mean_SouthKorea

mean_Brazil = df.loc[df["Location"] == "Brazil", "Score"].sum()
mean_Brazil /= df_Country[9]
df.loc[df["Location"] == "Brazil", "Mean"] = mean_Brazil

df
```

Out[3]:

	World Rank	Institution	Location	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	3	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7
2	3	Stanford University	USA	3	10	4	3	2
3	4	University of Cambridge	United Kingdom	1	1	21	4	11
4	5	University of Oxford	United Kingdom	2	7	26	9	4
...	...	...	...	...	...	...	...	...
1995	1996	Polytechnic Institute of Bragança	Portugal	15	-	-	-	1917
1996	1997	Federal University of Maranhão	Brazil	57	-	-	-	1918
1997	1998	Autonomous University of Baja California	Mexico	20	-	-	-	1921
1998	1999	American University in Cairo	Egypt	18	-	299	-	-
1999	2000	Kyonggi University	South Korea	61	-	-	-	1922

2000 rows × 10 columns

In [4]:
```
df.rename(columns={'Location':'Country'}, inplace=True)
df.head(10)
```

Out[4]:

	World Rank	Institution	Country	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance	S
0	1	Harvard University	USA	1	3	1	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7	
2	3	Stanford University	USA	3	10	4	3	2	
3	4	University of Cambridge	United Kingdom	1	1	21	4	11	
4	5	University of Oxford	United Kingdom	2	7	26	9	4	
5	6	Columbia University	USA	4	11	14	10	15	
6	7	Princeton University	USA	5	6	15	7	71	
7	8	University of Pennsylvania	USA	6	14	9	43	12	
8	9	University of Chicago	USA	7	8	18	29	22	
9	10	Yale University	USA	8	5	35	11	20	

In [9]:

```
# Integrate the number of university and comprehensive score upon each country

d = {'Country': ["USA", "China", "Japan", "United Kingdom", "France", "German
y", "Italy", "India", "South Korea", "Brazil"],
    'Number of University on the list': df_Country.head(10),
    'Mean Score': [mean_USA, mean_China, mean_Japan, mean_UnitedKingdom, mean_
France, mean_Germany, mean_Italy, mean_India,
                  mean_SouthKorea, mean_Brazil]}
df_1 = pd.DataFrame(data=d)
df_1.sort_values("Mean Score", inplace = True)
df_1
```

Out[9]:

	Country	Number of University on the list	Mean Score
India	India	64	68.842187
Brazil	Brazil	57	69.771930
Japan	Japan	126	70.269048
China	China	267	70.793258
South Korea	South Korea	61	71.193443
France	France	82	72.396341
Italy	Italy	66	72.684848
United Kingdom	United Kingdom	95	73.746316
USA	USA	357	73.832773
Germany	Germany	70	74.514286

However, the quantity can not directly reflects the quality. After knowing how many universities each country has, I calculate the average score by summing up the total university score sorted by each country and divide by the total number of universities. And I add the result to the table. After sorting up by the mean score, now I have a more comprehensive result and knowing which country has the best university education standard.

In [10]:

```python
# Add an external country average income resource to help with the analysis

url = "https://www.worlddata.info/average-income.php"
table_income = pd.read_html(url)
df_2 = table_income[0]
df_2["Country"] = df_2["Country"].replace(["United States"], "USA")
df_2.head()
```

Out[10]:

	Rank	Country	Average incomeannually	Monthly
0	1	Monaco	186,080 $	15,507 $
1	2	Liechtenstein	116,430 $	9,703 $
2	3	Bermuda	106,140 $	8,845 $
3	4	Switzerland	85,500 $	7,125 $
4	5	Norway	82,500 $	6,875 $

In [11]:

```python
# Use inner-join to combine two tables for better visualization

df_income = pd.merge(df_1, df_2, on ='Country', how ='inner')
df_income.rename(columns={'Average incomeannually':'Country Average Income Annually ($)'}, inplace=True)
df_income.pop("Monthly")
df_income.pop("Rank")
df_income['Country Average Income Annually ($)'] = df_income['Country Average Income Annually ($)'].str.replace('$', '')
df_income['Country Average Income Annually ($)'] = df_income['Country Average Income Annually ($)'].str.replace(',', '')
df_income['Country Average Income Annually ($)'] = df_income['Country Average Income Annually ($)'].astype(int)
df_income.sort_values("Country Average Income Annually ($)", inplace = True)
df_income
```

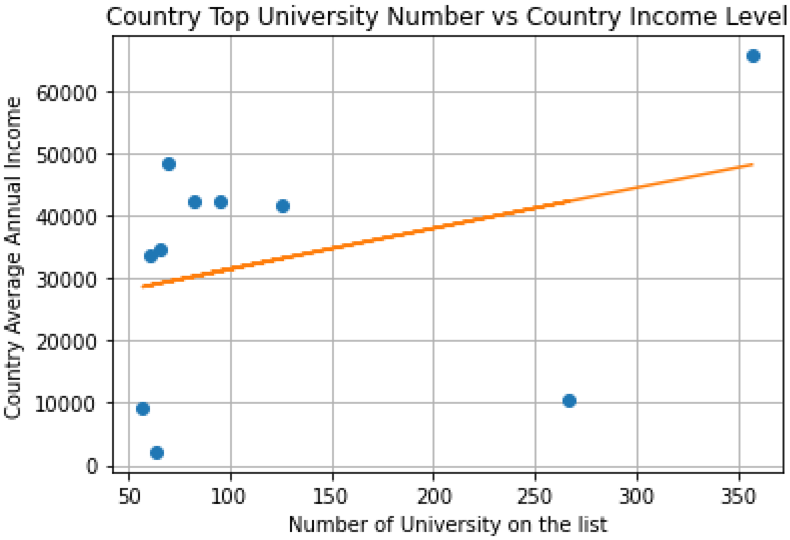
Out[11]:

	Country	Number of University on the list	Mean Score	Country Average Income Annually ($)
0	India	64	68.842187	2130
1	Brazil	57	69.771930	9130
3	China	267	70.793258	10410
4	South Korea	61	71.193443	33720
6	Italy	66	72.684848	34460
2	Japan	126	70.269048	41690
7	United Kingdom	95	73.746316	42370
5	France	82	72.396341	42400
9	Germany	70	74.514286	48520
8	USA	357	73.832773	65760

The importance of the earnings benefit of schooling is vital for a variety of social issues. These include economic and social policy, racial and ethnic discrimination, gender discrimination, income distribution, and the determinants of the demand for education. This link between education and earnings is formally made in the calculation of the rate of return to investment in education.

I think there exists some kind of relationship between a country's education level and people's income. Therefore, I used an external resource of people's average income by country and put this as a variable into the table.

In [12]:
```
# Make a scatter plot to visualize the data through the chart

import matplotlib.pyplot as plt
import numpy as np

x = df_income['Number of University on the list']
y = df_income['Country Average Income Annually ($)']
plt.xlabel("Number of University on the list")
plt.ylabel("Country Average Annual Income")
plt.title("Country Top University Number vs Country Income Level")
plt.plot(x, y, 'o')
m, b = np.polyfit(x, y, 1)
plt.plot(x, m*x + b)
plt.grid()
plt.show()
```



According to the linear regression plot we have, we can observe that there are three outliers corresponding to India, China, and Brazil. I realized that China has the second top university quantity in the world, while it does not match the country's average income level. Also, one thing noticeable is that among the three outliers: India, China, and brazil, those three are all developing countries, which implicitly explains the reason why their performance is far away from the line of best fit.

In [13]:

```
# Compare the data with previous year's data

url = "https://cwur.org/2019-20.php"
tables1 = pd.read_html(url)
df_previous = tables1[0]
df.rename(columns={'Score':'Score 2020-2021'}, inplace=True)
df_previous.rename(columns={'Score':'Score 2019-2020'}, inplace=True)
df_previous
```

Out[13]:

	World Rank	Institution	Location	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	2	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	1	10	2	5
2	3	Stanford University	USA	3	9	3	3	2
3	4	University of Cambridge	United Kingdom	1	4	19	5	11
4	5	University of Oxford	United Kingdom	2	10	24	10	4
...	...	...	...	...	...	...	...	...
1995	1996	University of Klagenfurt	Austria	17	-	-	-	1931
1996	1997	National University of Río Cuarto	Argentina	11	-	-	-	1932
1997	1998	Osmania University	India	68	505	1035	-	1942
1998	1999	Muğla Sıtkı Koçman University	Turkey	61	-	-	-	1933
1999	2000	Government College University Faisalabad	Pakistan	11	-	-	-	1934

2000 rows × 9 columns

In [14]:

```
# Eliminate the unnecessary factor

df_previous.pop("World Rank")
df_previous.pop("Location")
df_previous.pop("National Rank")
df_previous.pop("Quality\xa0of Education")
df_previous.pop("Alumni Employment")
df_previous.pop("Quality\xa0of Faculty")
df_previous.pop("Research Performance")
```

Out[14]:

```
0          1
1          5
2          2
3         11
4          4
        ...
1995    1931
1996    1932
1997    1942
1998    1933
1999    1934
Name: Research Performance, Length: 2000, dtype: object
```

In [15]:

```
# Use inner-join to combine two tables for better visualization

df_new = pd.merge(df, df_previous, on ='Institution', how ='inner')
df_new
```

Out[15]:

	World Rank	Institution	Country	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	3	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7
2	3	Stanford University	USA	3	10	4	3	2
3	4	University of Cambridge	United Kingdom	1	1	21	4	11
4	5	University of Oxford	United Kingdom	2	7	26	9	4
...	...	...	...	...	...	...	...	...
1886	1996	Polytechnic Institute of Bragança	Portugal	15	-	-	-	1917
1887	1997	Federal University of Maranhão	Brazil	57	-	-	-	1918
1888	1998	Autonomous University of Baja California	Mexico	20	-	-	-	1921
1889	1999	American University in Cairo	Egypt	18	-	299	-	-
1890	2000	Kyonggi University	South Korea	61	-	-	-	1922

1891 rows × 11 columns

In [16]:

```
# Compute each university's rating change

df_new["Floating Ratio"] = (df_new["Score 2020-2021"] - df_new["Score 2019-2020"])/df_new["Score 2020-2021"] * 100
df_new
```

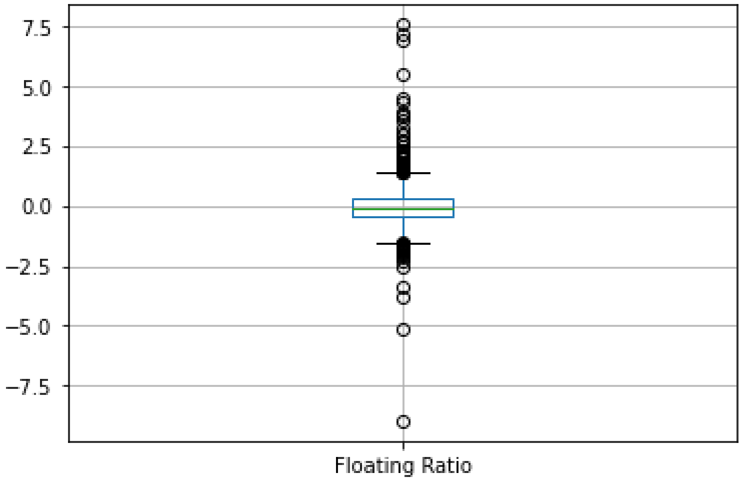
Out[16]:

	World Rank	Institution	Country	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	3	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7
2	3	Stanford University	USA	3	10	4	3	2
3	4	University of Cambridge	United Kingdom	1	1	21	4	11
4	5	University of Oxford	United Kingdom	2	7	26	9	4
...	...	...	...	...	...	...	...	...
1886	1996	Polytechnic Institute of Bragança	Portugal	15	-	-	-	1917
1887	1997	Federal University of Maranhão	Brazil	57	-	-	-	1918
1888	1998	Autonomous University of Baja California	Mexico	20	-	-	-	1921
1889	1999	American University in Cairo	Egypt	18	-	299	-	-
1890	2000	Kyonggi University	South Korea	61	-	-	-	1922

1891 rows × 12 columns

In [17]:

```
# Use boxplot to visualize the data

boxplot = df_new.boxplot(column=['Floating Ratio'])
```



The boxplot shows the floating score if we compare it with the data from 2019-2020. There exist a few outliers, but not many. It reflects the university has big changes in terms of the ranking score. If we put a certain condition on the table such as below, we can easily find out the list.

In [33]:

```
# Add condition to filter the universities have the large ranking change

df_new.loc[df_new["Floating Ratio"] > 5]
```

Out[33]:

	World Rank	Institution	Country	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
303	311	International Institute for Management Develop...	Switzerland	8	-	8	-	-
559	575	Haverford College	USA	156	18	-	-	1879
776	803	USI - University of Italian Speaking Switzerland	Switzerland	11	-	-	-	761
877	908	Antioch College	USA	207	22	-	-	-

In [19]:
```
# Add a map to help with better data visualization

! pip install pycountry-convert
from pycountry_convert import country_alpha2_to_continent_code, country_name_t
o_country_alpha2

def get_continent(col):
    cn_a2_code =  country_name_to_country_alpha2(col)
    cn_continent = country_alpha2_to_continent_code(cn_a2_code)
    return (cn_a2_code, cn_continent)
```

```
Requirement already satisfied: pycountry-convert in /opt/conda/lib/python3.8/
site-packages (0.7.2)
Requirement already satisfied: pytest-cov>=2.5.1 in /opt/conda/lib/python3.8/
site-packages (from pycountry-convert) (2.10.1)
Requirement already satisfied: pprintpp>=0.3.0 in /opt/conda/lib/python3.8/si
te-packages (from pycountry-convert) (0.4.0)
Requirement already satisfied: repoze.lru>=0.7 in /opt/conda/lib/python3.8/si
te-packages (from pycountry-convert) (0.7)
Requirement already satisfied: pycountry>=16.11.27.1 in /opt/conda/lib/python
3.8/site-packages (from pycountry-convert) (20.7.3)
Requirement already satisfied: pytest>=3.4.0 in /opt/conda/lib/python3.8/site
-packages (from pycountry-convert) (6.2.1)
Requirement already satisfied: wheel>=0.30.0 in /opt/conda/lib/python3.8/site
-packages (from pycountry-convert) (0.35.1)
Requirement already satisfied: pytest-mock>=1.6.3 in /opt/conda/lib/python3.
8/site-packages (from pycountry-convert) (3.4.0)
Requirement already satisfied: coverage>=4.4 in /opt/conda/lib/python3.8/site
-packages (from pytest-cov>=2.5.1->pycountry-convert) (5.3.1)
Requirement already satisfied: py>=1.8.2 in /opt/conda/lib/python3.8/site-pac
kages (from pytest>=3.4.0->pycountry-convert) (1.10.0)
Requirement already satisfied: packaging in /opt/conda/lib/python3.8/site-pac
kages (from pytest>=3.4.0->pycountry-convert) (20.4)
Requirement already satisfied: iniconfig in /opt/conda/lib/python3.8/site-pac
kages (from pytest>=3.4.0->pycountry-convert) (1.1.1)
Requirement already satisfied: toml in /opt/conda/lib/python3.8/site-packages
(from pytest>=3.4.0->pycountry-convert) (0.10.2)
Requirement already satisfied: pluggy<1.0.0a1,>=0.12 in /opt/conda/lib/python
3.8/site-packages (from pytest>=3.4.0->pycountry-convert) (0.13.1)
Requirement already satisfied: attrs>=19.2.0 in /opt/conda/lib/python3.8/site
-packages (from pytest>=3.4.0->pycountry-convert) (20.1.0)
Requirement already satisfied: pyparsing>=2.0.2 in /opt/conda/lib/python3.8/s
ite-packages (from packaging->pytest>=3.4.0->pycountry-convert) (2.4.7)
Requirement already satisfied: six in /opt/conda/lib/python3.8/site-packages
(from packaging->pytest>=3.4.0->pycountry-convert) (1.15.0)
```

In [20]:

```
# Add the country code and continent code according to the country name

import pandas as pd
pd.options.mode.chained_assignment = None

df["Country Code"] = "Unknown"
df["Continent Code"] = "Unknown"
count = 0
while count < df.shape[0]:
    cn_a2_code =  country_name_to_country_alpha2(df["Country"][count])
    cn_continent = country_alpha2_to_continent_code(cn_a2_code)
    df["Country Code"][count] = cn_a2_code
    df["Continent Code"][count] = cn_continent
    count += 1

df
```

Out[20]:

	World Rank	Institution	Country	National Rank	Quality of Education	Alumni Employment	Quality of Faculty	Research Performance
0	1	Harvard University	USA	1	3	1	1	1
1	2	Massachusetts Institute of Technology	USA	2	4	11	2	7
2	3	Stanford University	USA	3	10	4	3	2
3	4	University of Cambridge	United Kingdom	1	1	21	4	11
4	5	University of Oxford	United Kingdom	2	7	26	9	4
...	...	...	...	...	...	...	...	...
1995	1996	Polytechnic Institute of Bragança	Portugal	15	-	-	-	1917
1996	1997	Federal University of Maranhão	Brazil	57	-	-	-	1918
1997	1998	Autonomous University of Baja California	Mexico	20	-	-	-	1921
1998	1999	American University in Cairo	Egypt	18	-	299	-	-
1999	2000	Kyonggi University	South Korea	61	-	-	-	1922

2000 rows × 12 columns

In [21]:

```
# Combine the country code and continent code for further process

i = 0
df["Code"] = "Unknown"
while i < df.shape[0]:
    a = df["Country Code"][i]
    b = df["Continent Code"][i]
    df["Code"][i] = (a, b)
    i += 1
df.pop("National Rank")
df.pop("Alumni Employment")
df.pop("Quality\xa0of Education")
df.pop("Quality\xa0of Faculty")
df.pop("Research Performance")

df
```

Out[21]:

	World Rank	Institution	Country	Score 2020-2021	Mean	Country Code	Continent Code	Code
0	1	Harvard University	USA	100.0	73.8328	US	NA	(US, NA)
1	2	Massachusetts Institute of Technology	USA	96.7	73.8328	US	NA	(US, NA)
2	3	Stanford University	USA	95.2	73.8328	US	NA	(US, NA)
3	4	University of Cambridge	United Kingdom	94.1	73.7463	GB	EU	(GB, EU)
4	5	University of Oxford	United Kingdom	93.3	73.7463	GB	EU	(GB, EU)
...	...	...	...	...	...	...	...	...
1995	1996	Polytechnic Institute of Bragança	Portugal	65.8	Nah	PT	EU	(PT, EU)
1996	1997	Federal University of Maranhão	Brazil	65.8	69.7719	BR	SA	(BR, SA)
1997	1998	Autonomous University of Baja California	Mexico	65.8	Nah	MX	NA	(MX, NA)
1998	1999	American University in Cairo	Egypt	65.8	Nah	EG	AF	(EG, AF)
1999	2000	Kyonggi University	South Korea	65.8	71.1934	KR	AS	(KR, AS)

2000 rows × 8 columns

In [22]:

```
# Get the latitude and longitude upon given country

! pip install geopy
from geopy.geocoders import Nominatim

geolocator = Nominatim(user_agent = "--")

def geolocate_latitude(country):
    loc = geolocator.geocode(country)
    return loc.latitude

def geolocate_longitude(country):
    loc = geolocator.geocode(country)
    return loc.longitude
```

```
Requirement already satisfied: geopy in /opt/conda/lib/python3.8/site-package
s (2.0.0)
Requirement already satisfied: geographiclib<2,>=1.49 in /opt/conda/lib/pytho
n3.8/site-packages (from geopy) (1.50)
```

In [23]:

```
# Add the latitude and longitude to the dataframe

df["Latitude"] = 0
df["Longitude"] = 0

for p in range(0, 20):
    lat = geolocate_latitude(df["Country Code"][p])
    df["Latitude"][p] = lat

df.head(20)
```

Out[23]:

	World Rank	Institution	Country	Score 2020-2021	Mean	Country Code	Continent Code	Code	Latitude	Longitu
0	1	Harvard University	USA	100.0	73.8328	US	NA	(US, NA)	39	
1	2	Massachusetts Institute of Technology	USA	96.7	73.8328	US	NA	(US, NA)	39	
2	3	Stanford University	USA	95.2	73.8328	US	NA	(US, NA)	39	
3	4	University of Cambridge	United Kingdom	94.1	73.7463	GB	EU	(GB, EU)	54	
4	5	University of Oxford	United Kingdom	93.3	73.7463	GB	EU	(GB, EU)	54	
5	6	Columbia University	USA	92.6	73.8328	US	NA	(US, NA)	39	
6	7	Princeton University	USA	92.0	73.8328	US	NA	(US, NA)	39	
7	8	University of Pennsylvania	USA	91.6	73.8328	US	NA	(US, NA)	39	
8	9	University of Chicago	USA	91.1	73.8328	US	NA	(US, NA)	39	
9	10	Yale University	USA	90.7	73.8328	US	NA	(US, NA)	39	
10	11	California Institute of Technology	USA	90.4	73.8328	US	NA	(US, NA)	39	
11	12	University of California, Berkeley	USA	90.1	73.8328	US	NA	(US, NA)	39	
12	13	Cornell University	USA	89.8	73.8328	US	NA	(US, NA)	39	
13	14	University of Tokyo	Japan	89.5	70.269	JP	AS	(JP, AS)	36	
14	15	Johns Hopkins University	USA	89.3	73.8328	US	NA	(US, NA)	39	
15	16	University of Michigan, Ann Arbor	USA	89.0	73.8328	US	NA	(US, NA)	39	
16	17	Northwestern University	USA	88.8	73.8328	US	NA	(US, NA)	39	
17	18	University of California, Los Angeles	USA	88.6	73.8328	US	NA	(US, NA)	39	
18	19	University College London	United Kingdom	88.4	73.7463	GB	EU	(GB, EU)	54	
19	20	Duke University	USA	88.2	73.8328	US	NA	(US, NA)	39	

In [30]:

```
# From here I will add the map feature on top 20 university on the list since the whole ranking list size is very large

for q in range(0, 20):
    log = geolocate_longitude(df["Country Code"][q])
    df["Longitude"][q] = log

df.head(20)
```

Out[30]:

	World Rank	Institution	Country	Score 2020-2021	Mean	Country Code	Continent Code	Code	Latitude	Longitu
0	1	Harvard University	USA	100.0	73.8328	US	NA	(US, NA)	39	-1
1	2	Massachusetts Institute of Technology	USA	96.7	73.8328	US	NA	(US, NA)	39	-1
2	3	Stanford University	USA	95.2	73.8328	US	NA	(US, NA)	39	-1
3	4	University of Cambridge	United Kingdom	94.1	73.7463	GB	EU	(GB, EU)	54	
4	5	University of Oxford	United Kingdom	93.3	73.7463	GB	EU	(GB, EU)	54	
5	6	Columbia University	USA	92.6	73.8328	US	NA	(US, NA)	39	-1
6	7	Princeton University	USA	92.0	73.8328	US	NA	(US, NA)	39	-1
7	8	University of Pennsylvania	USA	91.6	73.8328	US	NA	(US, NA)	39	-1
8	9	University of Chicago	USA	91.1	73.8328	US	NA	(US, NA)	39	-1
9	10	Yale University	USA	90.7	73.8328	US	NA	(US, NA)	39	-1
10	11	California Institute of Technology	USA	90.4	73.8328	US	NA	(US, NA)	39	-1
11	12	University of California, Berkeley	USA	90.1	73.8328	US	NA	(US, NA)	39	-1
12	13	Cornell University	USA	89.8	73.8328	US	NA	(US, NA)	39	-1
13	14	University of Tokyo	Japan	89.5	70.269	JP	AS	(JP, AS)	36	1
14	15	Johns Hopkins University	USA	89.3	73.8328	US	NA	(US, NA)	39	-1
15	16	University of Michigan, Ann Arbor	USA	89.0	73.8328	US	NA	(US, NA)	39	-1
16	17	Northwestern University	USA	88.8	73.8328	US	NA	(US, NA)	39	-1
17	18	University of California, Los Angeles	USA	88.6	73.8328	US	NA	(US, NA)	39	-1
18	19	University College London	United Kingdom	88.4	73.7463	GB	EU	(GB, EU)	54	
19	20	Duke University	USA	88.2	73.8328	US	NA	(US, NA)	39	-1

In [25]:

```
# Combine the latitude and longitude to cooperate the further process action

r = 0
df["Geolocate"] = "Unknown"
while r < 20:
    e = df["Latitude"][r]
    f = df["Longitude"][r]
    df["Geolocate"][r] = (e, f)
    r += 1

df.head(20)
```

Out[25]:

	World Rank	Institution	Country	Score 2020-2021	Mean	Country Code	Continent Code	Code	Latitude	Longitu
0	1	Harvard University	USA	100.0	73.8328	US	NA	(US, NA)	39	-1
1	2	Massachusetts Institute of Technology	USA	96.7	73.8328	US	NA	(US, NA)	39	-1
2	3	Stanford University	USA	95.2	73.8328	US	NA	(US, NA)	39	-1
3	4	University of Cambridge	United Kingdom	94.1	73.7463	GB	EU	(GB, EU)	54	
4	5	University of Oxford	United Kingdom	93.3	73.7463	GB	EU	(GB, EU)	54	
5	6	Columbia University	USA	92.6	73.8328	US	NA	(US, NA)	39	-1
6	7	Princeton University	USA	92.0	73.8328	US	NA	(US, NA)	39	-1
7	8	University of Pennsylvania	USA	91.6	73.8328	US	NA	(US, NA)	39	-1
8	9	University of Chicago	USA	91.1	73.8328	US	NA	(US, NA)	39	-1
9	10	Yale University	USA	90.7	73.8328	US	NA	(US, NA)	39	-1
10	11	California Institute of Technology	USA	90.4	73.8328	US	NA	(US, NA)	39	-1
11	12	University of California, Berkeley	USA	90.1	73.8328	US	NA	(US, NA)	39	-1
12	13	Cornell University	USA	89.8	73.8328	US	NA	(US, NA)	39	-1
13	14	University of Tokyo	Japan	89.5	70.269	JP	AS	(JP, AS)	36	1
14	15	Johns Hopkins University	USA	89.3	73.8328	US	NA	(US, NA)	39	-1
15	16	University of Michigan, Ann Arbor	USA	89.0	73.8328	US	NA	(US, NA)	39	-1
16	17	Northwestern University	USA	88.8	73.8328	US	NA	(US, NA)	39	-1
17	18	University of California, Los Angeles	USA	88.6	73.8328	US	NA	(US, NA)	39	-1
18	19	University College London	United Kingdom	88.4	73.7463	GB	EU	(GB, EU)	54	
19	20	Duke University	USA	88.2	73.8328	US	NA	(US, NA)	39	-1

In [26]:

```
# Cut the unnecessary rows from the table

df.drop(df.tail(1980).index, inplace=True)
df
```

Out[26]:

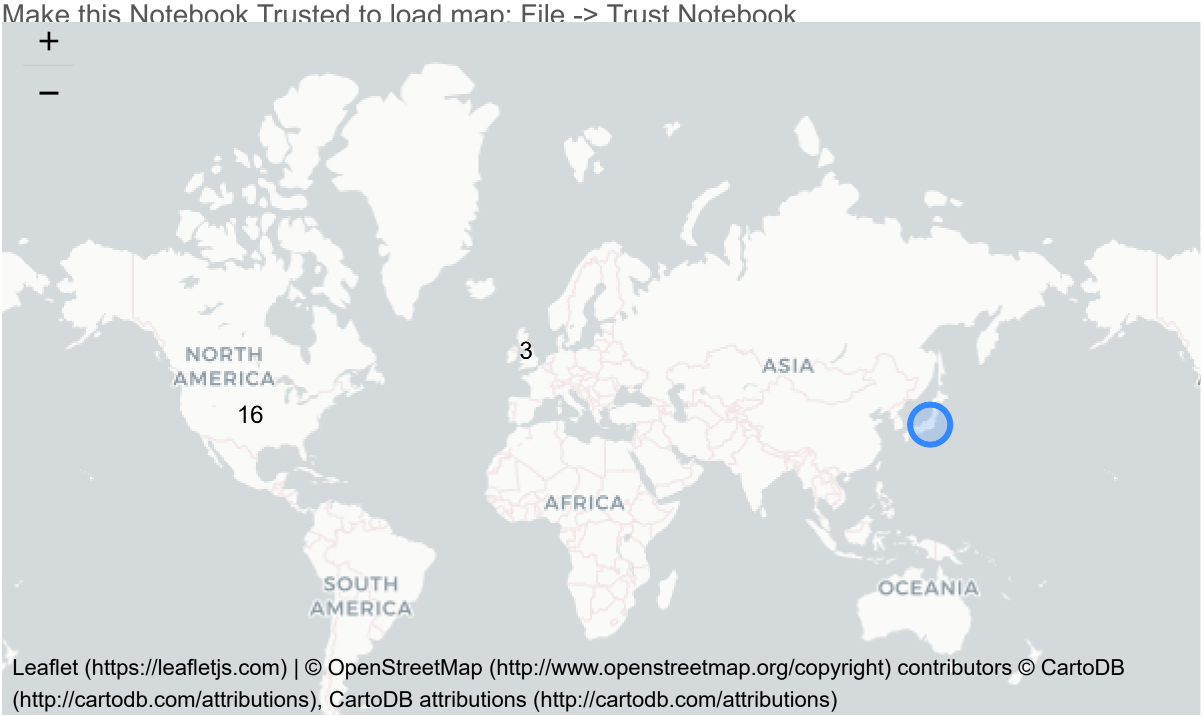
	World Rank	Institution	Country	Score 2020-2021	Mean	Country Code	Continent Code	Code	Latitude	Longitu
0	1	Harvard University	USA	100.0	73.8328	US	NA	(US, NA)	39	-1
1	2	Massachusetts Institute of Technology	USA	96.7	73.8328	US	NA	(US, NA)	39	-1
2	3	Stanford University	USA	95.2	73.8328	US	NA	(US, NA)	39	-1
3	4	University of Cambridge	United Kingdom	94.1	73.7463	GB	EU	(GB, EU)	54	
4	5	University of Oxford	United Kingdom	93.3	73.7463	GB	EU	(GB, EU)	54	
5	6	Columbia University	USA	92.6	73.8328	US	NA	(US, NA)	39	-1
6	7	Princeton University	USA	92.0	73.8328	US	NA	(US, NA)	39	-1
7	8	University of Pennsylvania	USA	91.6	73.8328	US	NA	(US, NA)	39	-1
8	9	University of Chicago	USA	91.1	73.8328	US	NA	(US, NA)	39	-1
9	10	Yale University	USA	90.7	73.8328	US	NA	(US, NA)	39	-1
10	11	California Institute of Technology	USA	90.4	73.8328	US	NA	(US, NA)	39	-1
11	12	University of California, Berkeley	USA	90.1	73.8328	US	NA	(US, NA)	39	-1
12	13	Cornell University	USA	89.8	73.8328	US	NA	(US, NA)	39	-1
13	14	University of Tokyo	Japan	89.5	70.269	JP	AS	(JP, AS)	36	1
14	15	Johns Hopkins University	USA	89.3	73.8328	US	NA	(US, NA)	39	-1
15	16	University of Michigan, Ann Arbor	USA	89.0	73.8328	US	NA	(US, NA)	39	-1
16	17	Northwestern University	USA	88.8	73.8328	US	NA	(US, NA)	39	-1
17	18	University of California, Los Angeles	USA	88.6	73.8328	US	NA	(US, NA)	39	-1
18	19	University College London	United Kingdom	88.4	73.7463	GB	EU	(GB, EU)	54	
19	20	Duke University	USA	88.2	73.8328	US	NA	(US, NA)	39	-1

In [31]:

```
# Visualize the university data on the map

! pip install folium
import folium
from folium.plugins import MarkerCluster

world_map= folium.Map(tiles="cartodbpositron")
marker_cluster = MarkerCluster().add_to(world_map)

for i in range(len(df)):
        lat = df.iloc[i]['Latitude']
        long = df.iloc[i]['Longitude']
        radius = 10
        popup_text = """Country : {}<br>
                    Institution : {}<br>"""
        popup_text = popup_text.format(df.iloc[i]['Country'], df.iloc[i]['Institution'])
        folium.CircleMarker(location = [lat, long], radius = radius, popup = popup_text, fill = True).add_to(marker_cluster)

world_map
```

```
Requirement already satisfied: folium in /opt/conda/lib/python3.8/site-packag
es (0.11.0)
Requirement already satisfied: jinja2>=2.9 in /opt/conda/lib/python3.8/site-p
ackages (from folium) (2.11.2)
Requirement already satisfied: numpy in /opt/conda/lib/python3.8/site-package
s (from folium) (1.19.1)
Requirement already satisfied: branca>=0.3.0 in /opt/conda/lib/python3.8/site
-packages (from folium) (0.4.1)
Requirement already satisfied: requests in /opt/conda/lib/python3.8/site-pack
ages (from folium) (2.24.0)
Requirement already satisfied: MarkupSafe>=0.23 in /opt/conda/lib/python3.8/s
ite-packages (from jinja2>=2.9->folium) (1.1.1)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/lib/python3.8/site-
packages (from requests->folium) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.
8/site-packages (from requests->folium) (2020.6.20)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /op
t/conda/lib/python3.8/site-packages (from requests->folium) (1.25.10)
Requirement already satisfied: chardet<4,>=3.0.2 in /opt/conda/lib/python3.8/
site-packages (from requests->folium) (3.0.4)
```

Out[31]: Make this Notebook Trusted to load map: File -> Trust Notebook



Leaflet (https://leafletjs.com) | © OpenStreetMap (http://www.openstreetmap.org/copyright) contributors © CartoDB (http://cartodb.com/attributions), CartoDB attributions (http://cartodb.com/attributions)

In the end, I integrate the map feature with the data for better visualization just like what we did on project 4. I take a sample of the top 20 universities from the list and put their location on the map. North America has 16 universities and all of them are from the United States. The United Kingdom has three universities from the top 20 and Japan has one university from the top 20.

Overall, I used what I have learned from CMSC320 and I benefit a lot from this final project. Since the project is n open topic and I could choose the one I am interested in and with no obligation. By the process of building this assignment, I feel like I keep more knowledge in my mind not only by figuring out the approach to deal with the problem, but also by getting more practice from the side of how to start from beginning with zero direction.