COVID-19 Data Visualization Report for CSE5544

Author: Fiona Fei

#Title: <Coursera Project-based Course: COVID19 Data Visualization Using Python/source code>

#Author:

#Date: <10-1-2020>

#Code version: <Python 3>

#Availability: https://www.coursera.org/learn/covid19-data-visualization-using-python/supplement/MhGNK/project-based-course-overview (https://www.coursera.org/learn/covid19-data-visualization-using-python/supplement/MhGNK/project-based-course-overview)

This project was completed by taking the coursera project-based course 'COVID19 Data Visualization Using Python'.

Part 1: choose your datasets and include the dataset links in the proposal

The dataset was chosen from Github, the link is below:
https://raw.githubusercontent.com/datasets/covid-19/master/data/countries-aggregated.csv (https://raw.githubusercontent.com/datasets/covid-19/master/data/countries-aggregated.csv)

Part 2: write down the libraries that you are going to use for your plottings

The libraries that I am going to use for my plottings are pandas, numpy, plotly, and matplotlib.pyplot.

Part 3: write down six COVID-19 related questions that you want to ask from the data, analyze the information, design visualizations to answer your questions, and explain why/how your visualizations answer your question.

#Question 1: How does the COVID-19 confirmed cases spread in globally over time?

#Question 2: How does the COVID-19 dead cases increases globally over time?

#Question 3: How does the infection rate in China increases overtime?

#Question 4: What is the maximum infection rate in China?

#Question 5: What is the maximum infection rate for every countries?

#Question 6: How does the infection rate changes one month after the lockeddown of Wuhan, China?¶

In [2]:
```
import pandas as pd
import numpy as np
import plotly.express as px
import matplotlib.pyplot as plt
import plotly.express as px
import plotly.io as pio
pio.renderers.default='notebook'
```

In [3]:
```
dataset_url = 'https://raw.githubusercontent.com/datasets/covid-19/master/d
df = pd.read_csv(dataset_url)
```

In [4]:
```
#df.head()
```

In [5]:
```
#df.tail()
```

Problem 2: How many variable components does this dataset contains? What are the types of the selected datasets, data, and attributes?

This dataset contains five variable components including 'Date','Country','Confirmed','Recovered',and 'Deaths'. This is a table dataset, with data types of items and attributes. The types of attributes includes categorical, nominal, and quantitative.

Problem 3: Describe what data analysis you need to do, that is, tools (e.g. Pandas) and operations you will use and need to do in order to extract useful information related to answering you questions.

In order to extract useful information, I need to:

1. Know about what does the dataset looks like;
2. Preprocesing the dataset by only take the data where confirmed case is greater than zero;
3. Sort out the columns and country I need, such as 'Date','Country','Confirmed' for columns, and 'China' for country.
4. Calculate the infection rate by using diff() function between daily confirmed cases.
5. Find out the maximum infection rate using the function max();
6. List all the countries by using the function unique();
7. Create a new dataframe of the maximum infection rate in each country using pandas;

Problem 4: Describe the visualization design process: what are the marks and channels used in each of your visualizations; What visual variables (e.g., size, value, texture, color, orientation, shape, 2D plane) you are using in your visualizations; How do you map each component in your data to the visual variables (answered after each visualization).

In [6]:
```
#Check how does the dataset looks like

df.shape
```

Out[6]:
```
(48316, 5)
```

In [7]:
```
#Preprocessing of the dataset
df = df[df.Confirmed > 0]
df.head()
```

Out[7]:

	Date	Country	Confirmed	Recovered	Deaths
33	2020-02-24	Afghanistan	1	0	0
34	2020-02-25	Afghanistan	1	0	0
35	2020-02-26	Afghanistan	1	0	0
36	2020-02-27	Afghanistan	1	0	0
37	2020-02-28	Afghanistan	1	0	0

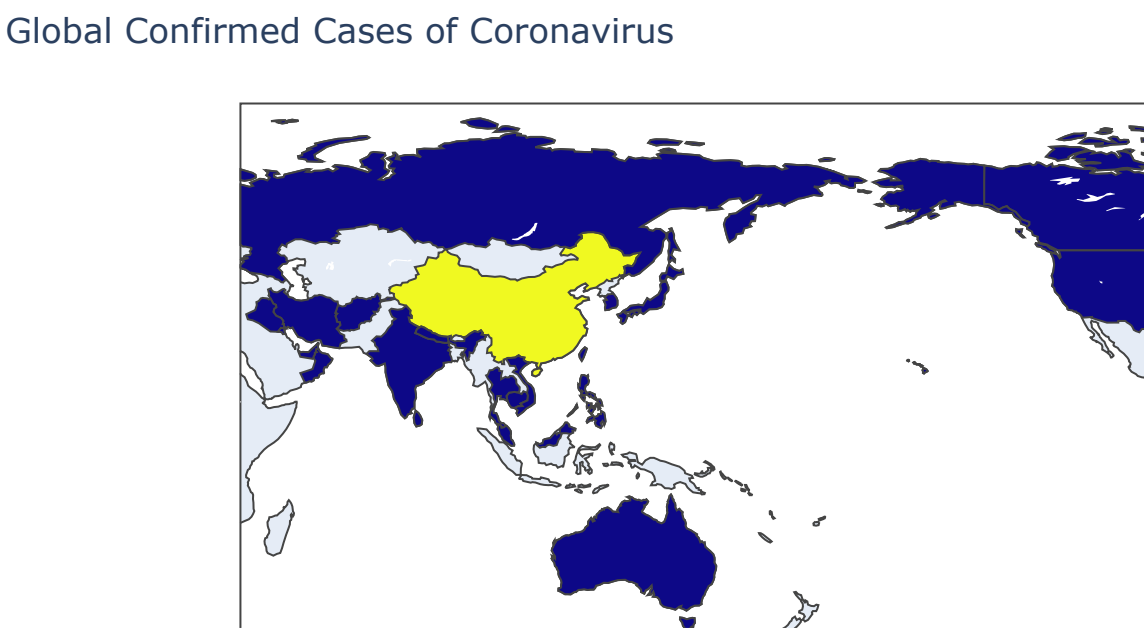
Question #1: How does the COVID-19 confirmed cases spread in globally over time?

In [8]:
```
df_china = df[df.Country == 'China']
df_china.head()
```

Out[8]:

	Date	Country	Confirmed	Recovered	Deaths
9252	2020-01-22	China	548	28	17
9253	2020-01-23	China	643	30	18
9254	2020-01-24	China	920	36	26
9255	2020-01-25	China	1406	39	42
9256	2020-01-26	China	2075	49	56

In [9]:
```
fig = px.choropleth(df, locations = 'Country', locationmode = 'country name
                    ,animation_frame = 'Date')
fig.update_layout(title_text = 'Global Confirmed Cases of Coronavirus')
fig.show()
```

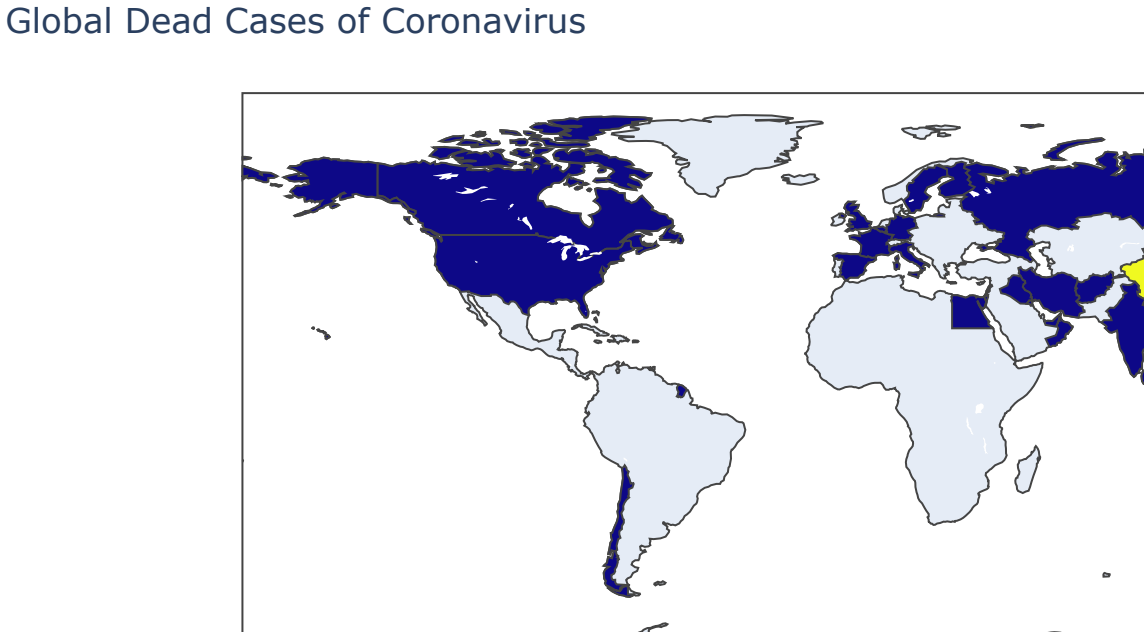


Visualization #1:

1. Marks and Channels: marks(area),channels(value, color)
2. Visual variables(level of perception):Color(selective), Shape(selective)
3. How to map each components in data to visual variables: confirmed cases-color, countries - shape.
4. How questions are answered through the visualization: Through this visualization, we can see that how does the confirmed cases changes overtime by looking at the color on top of each country. When the color is purple, we know that the confirmed cases is low. When the color gets close to yellow, we will know that the confirmed cases is high.

Question 2: How does the COVID-19 dead cases increases globally over time?

In [10]:
```
#Global deaths of COVID-19
fig = px.choropleth(df, locations = 'Country', locationmode = 'country name
                    ,animation_frame = 'Date')
fig.update_layout(title_text = 'Global Dead Cases of Coronavirus')
fig.show()
```



Visualization #2:

1. Marks and Channels: marks(area),channels(value, color)
2. Visual variables(level of perception):Color(selective, associative), Shape(selective)
3. How to map each components in data to visual variables: death cases-color, countries - shape.
4. How questions are answered through the visualization: Through this visualization, we can see that how does the death cases changes overtime by looking at the color on top of each country. When the color is purple, we know that the death cases is low. When the color gets close to yellow, we will know that the deadth cases is high.

Question 3: How does the infection rate in China increases overtime?

In [11]:
```
df_china = df_china[['Date', 'Confirmed']]
df_china.head()
```

Out[11]:

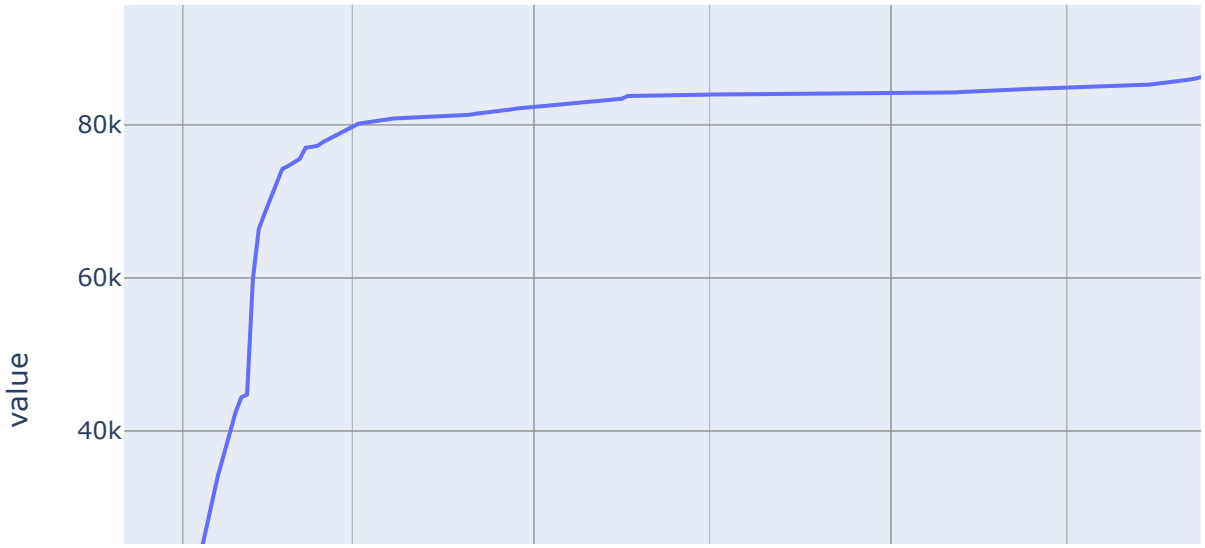
	Date	Confirmed
9252	2020-01-22	548
9253	2020-01-23	643
9254	2020-01-24	920
9255	2020-01-25	1406
9256	2020-01-26	2075

In [12]:
```
df_china['Infection Rate'] = df_china['Confirmed'].diff()
df_china.head()
```

Out[12]:

	Date	Confirmed	Infection Rate
9252	2020-01-22	548	NaN
9253	2020-01-23	643	95.0
9254	2020-01-24	920	277.0
9255	2020-01-25	1406	486.0
9256	2020-01-26	2075	669.0

In [13]:
```
#How does the Confirmed cases and Infection Rate increases over time in Chi
px.line(df_china, x = 'Date', y = ['Confirmed', 'Infection Rate'])
```



Visualization #3:

1. Marks and Channels: mark(line), channel(tilt, curvature)
2. Visual variables(level of perception): color(selective,associative), 2D Plane(Quantitative)
3. How to map each components in data to visual variables: Confirmed cases - blue line, infection rate - red line.
4. How questions are answered through the visualization: The x-axis stands for the date, and the y-axis stands for the total number of cases. We can see how does the confirmed cases and infection rate grow over time from this visualization.

Question 4: What is the maximum infection rate in China?

In [14]:
```
#Maximum infection rate in China
df_china['Infection Rate'].max()
```

Out[14]:
```
15136.0
```

Question 5: What is the maximum infection rate for every countries?

In [15]:
```
countries = list(df['Country'].unique())
#countries
max_infection_rates = []
for c in countries:
    MIR = df[df.Country == c].Confirmed.diff().max()
    max_infection_rates.append(MIR)
print(max_infection_rates)
```

```
[915.0, 178.0, 675.0, 130.0, 159.0, 39.0, 14687.0, 771.0, 716.0, 1321.0,
590.0, 220.0, 841.0, 4019.0, 12.0, 973.0, 3389.0, 113.0, 139.0, 29.0, 203
6.0, 811.0, 354.0, 69074.0, 26.0, 503.0, 193.0, 1291.0, 78.0, 149.0, 31.
0, 2324.0, 2778.0, 216.0, 83.0, 13990.0, 15136.0, 15318.0, 44.0, 649.0, 4
14.0, 1907.0, 430.0, 369.0, 93.0, 58.0, 5336.0, 823.0, 99.0, 280.0, 6.0,
2147.0, 11536.0, 1774.0, 449.0, 1750.0, 31.0, 134.0, 147.0, 1829.0, 5.0,
267.0, 29237.0, 570.0, 248.0, 554.0, 6933.0, 1513.0, 460.0, 6.0, 4233.0,
278.0, 156.0, 133.0, 332.0, 3.0, 1141.0, 1322.0, 99.0, 97894.0, 4823.0, 3
825.0, 5055.0, 1515.0, 11316.0, 6557.0, 244.0, 1762.0, 1776.0, 18757.0, 9
60.0, 851.0, 695.0, 1073.0, 11505.0, 3.0, 95.0, 1321.0, 98.0, 45.0, 1085.
0, 21.0, 172.0, 234.0, 7.0, 614.0, 257.0, 317.0, 215.0, 81.0, 106.0, 227.
0, 41.0, 9556.0, 1022.0, 12.0, 56.0, 714.0, 2760.0, 305.0, 316.0, 2722.0,
6980.0, 89.0, 480.0, 69.0, 790.0, 240.0, 386.0, 2685.0, 12073.0, 1540.0,
73.0, 1217.0, 21358.0, 6725.0, 2367.0, 1516.0, 2355.0, 2343.0, 11656.0, 2
31.0, 5.0, 6.0, 7.0, 54.0, 151.0, 4919.0, 223.0, 878.0, 50.0, 86.0, 1426.
0, 818.0, 238.0, 152.0, 13944.0, 323.0, 31785.0, 300.0, 410.0, 139.0, 169
8.0, 1321.0, 105.0, 27.0, 211.0, 181.0, 188.0, 10.0, 39.0, 217.0, 1722.0,
5138.0, 77362.0, 423.0, 4785.0, 1231.0, 22965.0, 36.0, 981.0, 1281.0, 50.
0, 888.0, 3.0, 116.0, 915.0, 490.0]
```

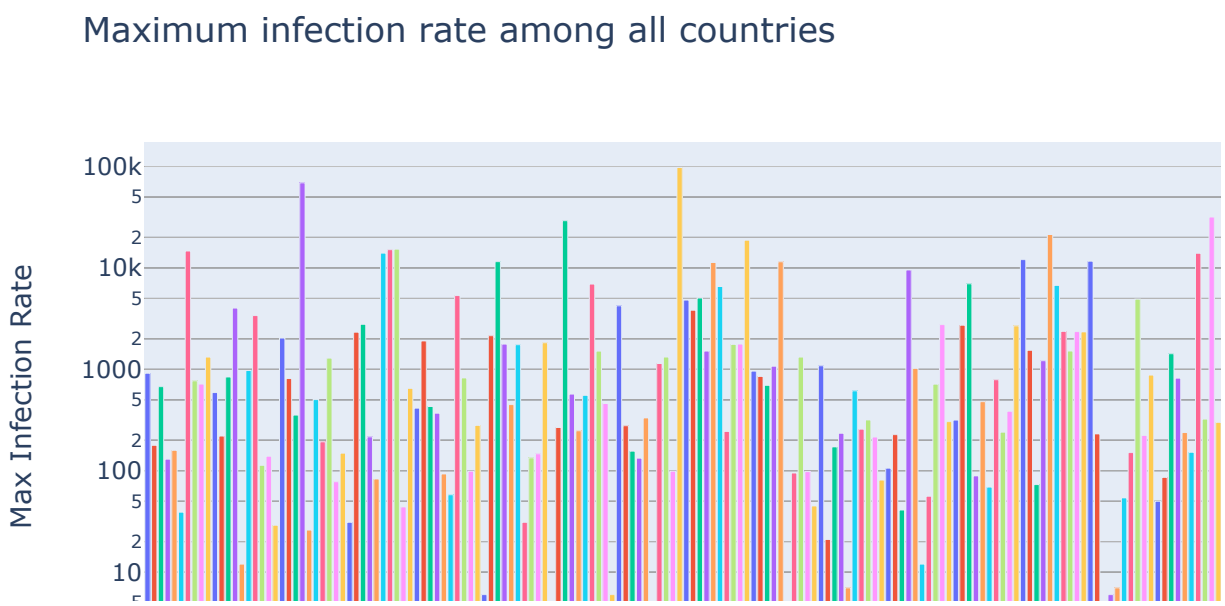
In [16]:
```
#Create new Dataframe

df_MIR = pd.DataFrame()
df_MIR['Country'] = countries
df_MIR['Max Infection Rate'] = max_infection_rates
df_MIR.head()
```

Out[16]:

	Country	Max Infection Rate
0	Afghanistan	915.0
1	Albania	178.0
2	Algeria	675.0
3	Andorra	130.0
4	Angola	159.0

In [17]:
```
#Plot maximum infection rate of each country
px.bar(df_MIR, x = 'Country', y = 'Max Infection Rate', color = 'Country'
      , title = 'Maximum infection rate among all countries', log_y = True)
```



Visualization #4:

1. Marks and Channels: mark(lines),channel(color, length)
2. Visual variables(level of perception):color(selective, associative), 2D plane(quantitative)
3. How to map each components in data to visual variables: country-color, Max infection rate-2D plane's y axis, country-2D plane's x axis.
4. How questions are answered through the visualization: For the x axis we can see there are different countries sorted in alphabetical order. On the y-axis, we are able to identify the maximum infection rate in the country.

Question 6: How does the infection rate changes one month after the lockeddown of Wuhan, China?

The lockedown date was found from an article from the World Health Organization:
https://www.who.int/bulletin/volumes/98/7/20-254045/en/
(https://www.who.int/bulletin/volumes/98/7/20-254045/en/)

In [18]:
```
china_lockdown_start_date = '2020-01-23'
china_lockdown_a_month_later = '2020-02-23'
```

In [19]:
```
df_china = df[df.Country == 'China']
```

In [20]:
```
df_china.head()
```

Out[20]:

	Date	Country	Confirmed	Recovered	Deaths
9252	2020-01-22	China	548	28	17
9253	2020-01-23	China	643	30	18
9254	2020-01-24	China	920	36	26
9255	2020-01-25	China	1406	39	42
9256	2020-01-26	China	2075	49	56

In [21]:
```
df_china['Infection Rate'] = df_china.Confirmed.diff()
```

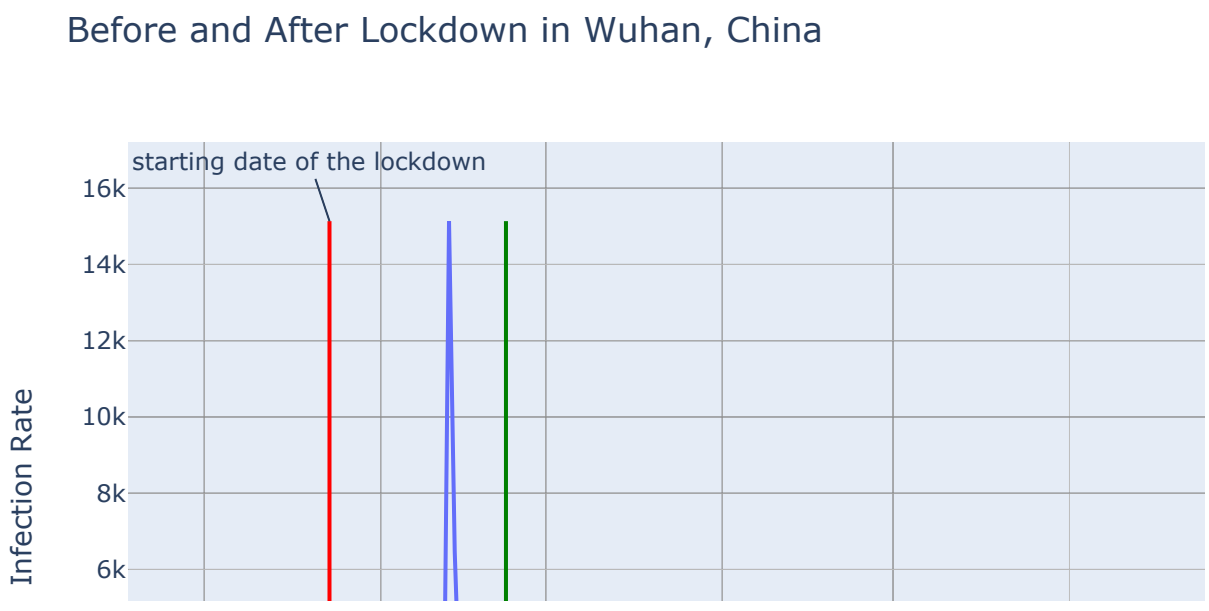
```
<ipython-input-21-9864c9978302>:1: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
```

In [22]:
```
fig = px.line(df_china, x = 'Date', y = 'Infection Rate', title = "Before a
fig.add_shape(
    dict(
    type = "line",
    x0=china_lockdown_start_date,
    y0=0,
    x1=china_lockdown_start_date,
    y1=df_china['Infection Rate'].max(),
    line = dict(color='red', width=2)
    )
)
fig.add_annotation(
    dict(
    x = china_lockdown_start_date,
    y = df_china['Infection Rate'].max(),
    text = 'starting date of the lockdown'
    )
)

fig.add_shape(
    dict(
    type = "line",
    x0=china_lockdown_a_month_later,
    y0=0,
    x1=china_lockdown_a_month_later,
    y1=df_china['Infection Rate'].max(),
    line = dict(color='green', width=2)
    )
)
fig.add_annotation(
    dict(
    x = china_lockdown_a_month_later,
    y = 0,
    text = 'One month after the lockdown'
    )
)
```



Visualization #5:

1. Marks and Channels:mark(lines), channel(tilt, color, curvature)
2. Visual variables(level of perception): color(selective, associative), 2D plane(quantitative)
3. How to map each components in data to visual variables: color-starting date, one month after date, infection rate, 2D plane-the value of infection rate and the date.
4. How questions are answered through the visualization: From this visualization, we can see how does the infection rate of China grows over time. The red line stands for the starting date of the lockdown, and the green line stand for one month after the lockdown in Wuhan, China. We are able to see how does the trend of the infection rate changes between two lines.

In [ ]:

In [ ]:

In [ ]:

In [ ]: