avs-nexus

July 25, 2024

[1]:

```
import json
import folium
import requests
import mimetypes
import http.client
import numpy as np
import pandas as pd
import plotly.express as px
import matplotlib.pyplot as plt
import plotly.graph_objects as go
from folium.plugins import HeatMap
from pandas.io.json import json_normalize
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:

```
# conn= http.client.HTTPSConnection("api.covid19api.com")
# payload=''
# headers={}
# conn.request("GET","/summary",payload,headers)
# res=conn.getresponse()
# data=res.read().decode('UTF-8')
```

[3]:

```
# pd.set_option("display.max_rows", None, "display.max_columns", None)
```

[4]:

```
# covid1=json.loads(data)
```

[5]:

```
# pd.json_normalize(covid1['Countries'],sep=",")
# df=pd.DataFrame(covid1['Countries'])
# df
```

[6]:

```
# covid2=df.drop(columns=['ID','CountryCode','Slug','Date','Premium'],axis=1)
# covid2=covid2.sort_values(by=['TotalConfirmed'],ascending=False)
# covid2['Country'] = covid2['Country'].replace('Russian Federation', 'Russia')
# covid2['Country'] = covid2['Country'].replace('Iran, Islamic Republic of',␣
 ↪'Iran')
# covid2['Country'] = covid2['Country'].replace('Venezuela (Bolivarian␣
 ↪Republic)', 'Venezuela')
```

```
# covid3=covid2
# covid2
```

[7]:

```
# SARS
sars_03 = pd.read_csv(r"C:
  ↪\Users\Asus\Desktop\Project\DV\DV_Project\sars_2003_complete_dataset_clean.
  ↪csv",parse_dates=['Date'])
sars_03 = sars_03[['Date', 'Country', 'Cumulative number of case(s)','Number of␣
  ↪deaths', 'Number recovered']]
sars_03.columns = ['Date', 'Country', 'Cases', 'Deaths', 'Recovered']
sars_03 = sars_03.groupby(['Date', 'Country'])
sars_03 = sars_03.sum().reset_index()
sars_03['Cases'] = sars_03['Cases'].fillna(0)
sars_03['Deaths'] = sars_03['Deaths'].fillna(0)
sars_03['Recovered'] = sars_03['Recovered'].fillna(0)
sars_latest = sars_03[sars_03['Date'] == max(sars_03['Date'])].reset_index()
sars_latest=sars_latest.sort_values(by=['Cases'],ascending=False)

# H1N1
h1n1_09 = pd.read_csv(r"C:
  ↪\Users\Asus\Desktop\Project\DV\DV_Project\Pandemic_H1N1_2009.
  ↪csv",parse_dates=['Update Time'])
h1n1_09 = h1n1_09[['Update Time', 'Country', 'Cases','Deaths']]
h1n1_09 = h1n1_09.groupby(['Update Time', 'Country'])
h1n1_09 = h1n1_09.sum().reset_index()
h1n1_09['Cases'] = h1n1_09['Cases'].fillna(0)
h1n1_09['Deaths'] = h1n1_09['Deaths'].fillna(0)
h1n1_09['Deaths'] = h1n1_09['Deaths'].astype('int')
h1n1_latest=h1n1_09[h1n1_09['Update Time'] == max(h1n1_09['Update Time'])].
  ↪reset_index()
h1n1_latest=h1n1_latest.sort_values(by=['Cases'],ascending=False)
h1n1_latest=h1n1_latest.drop([43])

# EBOLA
ebola_14 = pd.read_csv(r"C:
  ↪\Users\Asus\Desktop\Project\DV\DV_Project\ebola_2014_2016_clean.
  ↪csv",parse_dates=['Date'])
ebola_14 = ebola_14[['Date', 'Country',
 'Cumulative no. of confirmed, probable and suspected cases','Cumulative no. of␣
  ↪confirmed, probable and suspected deaths']]
ebola_14.columns = ['Date', 'Country', 'Cases', 'Deaths']
ebola_14 = ebola_14.groupby(['Date', 'Country'])
ebola_14 = ebola_14.sum().reset_index()
ebola_14['Cases'] = ebola_14['Cases'].fillna(0)
ebola_14['Deaths'] = ebola_14['Deaths'].fillna(0)
ebola_14['Cases'] = ebola_14['Cases'].astype('int')
```

```
ebola_14['Deaths'] = ebola_14['Deaths'].astype('int')
ebola_latest = ebola_14[ebola_14['Date'] == max(ebola_14['Date'])].reset_index()
ebola_latest=ebola_latest.sort_values(by=['Cases'],ascending=False)

#MERS
mers_latest = pd.read_csv(r"C:
 ↪\Users\Asus\Desktop\Project\DV\DV_Project\country_count_latest.csv")
mers_latest=mers_latest.sort_values(by=['Confirmed'],ascending=False)
```

[8]:
```
sars_latest
```

[8]:
```
    index        Date                Country  Cases  Deaths  Recovered
2    2508  2003-07-11                  China   5327     348       4941
7    2513  2003-07-11     Hong Kong SAR, China   1755     298       1433
26   2532  2003-07-11         Taiwan, China    671      84        507
1    2507  2003-07-11                 Canada    250      38        194
21   2527  2003-07-11              Singapore    206      32        172
29   2535  2003-07-11          United States     75       0         67
30   2536  2003-07-11               Viet Nam     63       5         58
16   2522  2003-07-11            Philippines     14       2         12
6    2512  2003-07-11                Germany     10       0          9
14   2520  2003-07-11               Mongolia      9       0          9
27   2533  2003-07-11               Thailand      9       2          7
5    2511  2003-07-11                 France      7       1          6
13   2519  2003-07-11               Malaysia      5       2          3
10   2516  2003-07-11                  Italy      4       0          4
28   2534  2003-07-11         United Kingdom      4       0          4
8    2514  2003-07-11                  India      3       0          3
18   2524  2003-07-11      Republic of Korea      3       0          3
24   2530  2003-07-11                 Sweden      3       0          3
9    2515  2003-07-11              Indonesia      2       0          2
22   2528  2003-07-11           South Africa      1       1          0
25   2531  2003-07-11            Switzerland      1       0          1
23   2529  2003-07-11                  Spain      1       0          1
0    2506  2003-07-11                 Brazil      1       0          1
20   2526  2003-07-11     Russian Federation      1       0          0
19   2525  2003-07-11                Romania      1       0          1
17   2523  2003-07-11    Republic of Ireland      1       0          1
12   2518  2003-07-11       Macao SAR, China      1       0          1
11   2517  2003-07-11                 Kuwait      1       0          1
4    2510  2003-07-11                Finland      1       0          1
3    2509  2003-07-11               Colombia      1       0          1
15   2521  2003-07-11            New Zealand      1       0          1
```

[9]:
```
h1n1_latest
```

[9]:
```
     index         Update Time                   Country  Cases  Deaths
116   1802 2009-07-06 09:00:00  United States of America  33902     170
72    1758 2009-07-06 09:00:00                    Mexico  10262     119
18    1704 2009-07-06 09:00:00                    Canada   7983      25
115   1801 2009-07-06 09:00:00            United Kingdom   7447       3
21    1707 2009-07-06 09:00:00                     Chile   7376      14
..     …                   …                       …     …      …
106   1792 2009-07-06 09:00:00                     Syria      1       0
75    1761 2009-07-06 09:00:00                   Myanmar      1       0
82    1768 2009-07-06 09:00:00                     Palau      1       0
84    1770 2009-07-06 09:00:00          Papua New Guinea      1       0
135   1821 2009-07-06 09:00:00            Virgin Islands      1       0

[135 rows x 5 columns]
```

[10]:
```
ebola_latest
```

[10]:
```
   index        Date                   Country  Cases  Deaths
6   2375  2016-03-23              Sierra Leone  14122    3955
2   2371  2016-03-23                   Liberia  10680    4813
0   2369  2016-03-23                    Guinea   3804    2536
4   2373  2016-03-23                   Nigeria     20       8
3   2372  2016-03-23                      Mali      8       6
9   2378  2016-03-23  United States of America      4       1
1   2370  2016-03-23                     Italy      1       0
5   2374  2016-03-23                   Senegal      1       0
7   2376  2016-03-23                     Spain      1       0
8   2377  2016-03-23            United Kingdom      1       0
```

[11]:
```
mers_latest
```

[11]:
```
                     Country  Confirmed
19              Saudi Arabia       2058
18        Republic of Korea        186
24      United Arab Emirates         87
10                    Jordan         28
15                      Oman         24
17                     Qatar         19
8                       Iran          6
23            United Kingdom          5
11                    Kuwait          4
21                   Tunisia          3
6                    Germany          3
20                  Thailand          3
0                    Algeria          2
25  United States of America          2
16               Philippines          2
```

```
13                 Malaysia          2
14              Netherlands          2
1                   Austria          2
12                  Lebanon          2
5                    France          2
9                     Italy          1
7                    Greece          1
22                   Turkey          1
4                     Egypt          1
3                     China          1
2                   Bahrain          1
26                    Yemen          1
```

[12]:
```
# fig = px.treemap(covid3, names='Country', parents=[""]*len(covid3),␣
 ↪values='TotalConfirmed')
# fig.add_trace(go.Treemap(
#     labels = list(covid3.Country),
#     parents = [""]*len(covid3),
#     values =  list(covid3.TotalConfirmed),
#     textinfo = "label+value+percent parent",
#     ),row = 1, col = 1)
# fig.show()
```

[13]:
```
# m=folium.Map(tiles="Stamen Terrain",min_zoom=1.5)
```

[14]:
```
#ERROR
url='https://raw.githubusercontent.com/python-visualization/folium/master/
 ↪examples/data'
country_shapes=f'{url}/world-countries.json'
```

[15]:
```
# folium.Choropleth(
#     geo_data=country_shapes,
#     min_zoom=2,
#     name='Covid-19',
#     data=covid3,
#     columns=['Country','TotalConfirmed'],
#     key_on='feature.properties.name',
#     fill_color='OrRd',
#     nan_fill_color='white',
#     legend_name='Total Confirmed Covid Cases',
# ).add_to(m)
```

[16]:
```
# covid3.update(covid3['TotalConfirmed'].map('Total Confirmed:{}'.format))
# covid3.update(covid3['TotalRecovered'].map('Total Recovered:{}'.format))
```

[14]:
```
coordinates=pd.read_csv(r"C:
 ↪\Users\Asus\Desktop\Project\DV\DV_Project\world_country_and_usa_states_latitude_and_longitu
 ↪csv")
coordinates = coordinates.rename(columns={'country': 'Country'})
coordinates= coordinates.drop(columns=['country_code',␣
 ↪'usa_state_code','usa_state_latitude','usa_state_longitude','usa_state'])
coordinates
```

[14]:
```
      latitude  longitude               Country
0    42.546245   1.601554               Andorra
1    23.424076  53.847818  United Arab Emirates
2    33.939110  67.709953           Afghanistan
3    17.060816 -61.796428   Antigua and Barbuda
4    18.220554 -63.068615              Anguilla
..         …          …                   …
240  15.552727  48.516388                 Yemen
241 -12.827500  45.166244               Mayotte
242 -30.559482  22.937506          South Africa
243 -13.133897  27.849332                Zambia
244 -19.015438  29.154857              Zimbabwe

[245 rows x 3 columns]
```

[15]:
```
# covid_final=pd.merge(covid3,coordinates,on='Country')
```

[16]:
```
def plotDot(point):
    folium.CircleMarker(location=[point.latitude,point.longitude],
                        radius=5,
                        weight=2,
                        popup=[point.Country,point.TotalConfirmed,point.
 ↪TotalRecovered],
                        fill_color='#000000').add_to(m)
```

[20]:
```
# covid_final.apply(plotDot,axis=1)
# m.fit_bounds(m.get_bounds())
# m
```

[21]:
```
# m1=folium.Map(tiles="StamenToner",min_zoom=2)
```

[22]:
```
# deaths=covid_final['TotalDeaths'].astype(float)
# lat=covid_final['latitude'].astype(float)
# lon=covid_final['longitude'].astype(float)
# m1=m1.add_child(HeatMap(zip(lat,lon,deaths),radius=0))
```

[23]:
```
# def plotDot(point):
#     folium.CircleMarker(location=[point.latitude,point.longitude],
#                         radius=5,
#                          weight=2,
#                          popup=[point.Country,point.TotalDeaths],
#                          fill_color='#000000').add_to(m1)
```

[24]:

```
# covid_final.apply(plotDot,axis=1)
# m1.fit_bounds(m1.get_bounds())
# m1
```

[17]:

```
h_df=h1n1_latest
fig = px.treemap(h_df, names='Country', parents=[""]*len(h_df), values='Cases')
fig.add_trace(go.Treemap(
    labels = list(h_df.Country),
    parents = [""]*len(h_df),
    values =  list(h_df.Cases),
    textinfo = "label+value+percent parent",
    ),row = 1, col = 1)
fig.show()
```

[18]:

```
#ERROR IN URL
m6=folium.Map(tiles="Stamen Terrain",min_zoom=1.5)
url='https://raw.githubusercontent.com/python-visualization/folium/master/
 ↪examples/data'
country_shapes=f'{url}/world-countries.json'
folium.Choropleth(
    geo_data=country_shapes,
    min_zoom=2,
    name='H1N1',
    data=h1n1_latest,
    columns=['Country','Cases'],
    key_on='feature.properties.name',
    fill_color='OrRd',
    nan_fill_color='white',
    legend_name='Total Confirmed H1N1 Cases',
).add_to(m6)
h1n1_final=pd.merge(h1n1_latest,coordinates,on='Country')
h1n1_final
```

[18]:

```
     index          Update Time           Country  Cases  Deaths   latitude  \
0     1758  2009-07-06 09:00:00            Mexico  10262     119  23.634501
1     1704  2009-07-06 09:00:00            Canada   7983      25  56.130366
2     1801  2009-07-06 09:00:00    United Kingdom   7447       3  55.378051
3     1707  2009-07-06 09:00:00             Chile   7376      14 -35.675147
4     1689  2009-07-06 09:00:00         Australia   5298      10 -25.274398
..     …                    …                …     …       …         …
101   1752  2009-07-06 09:00:00             Libya      1       0  26.335100
102   1757  2009-07-06 09:00:00         Mauritius      1       0 -20.348404
103   1792  2009-07-06 09:00:00             Syria      1       0  34.802075
```

```
104   1768 2009-07-06 09:00:00             Palau       1        0   7.514980
105   1770 2009-07-06 09:00:00  Papua New Guinea       1        0  -6.314993

      longitude
0   -102.552784
1   -106.346771
2     -3.435973
3    -71.542969
4    133.775136
..          …
101   17.228331
102   57.552152
103   38.996815
104  134.582520
105  143.955550

[106 rows x 7 columns]
```

[27]:

```
def plotDot(point):
    folium.CircleMarker(location=[point.latitude,point.longitude],
                        radius=5,
                        weight=2,
                        popup=[point.Country,point.Cases,point.Deaths],
                        fill_color='#000000').add_to(m6)
```

[28]:

```
h1n1_final.apply(plotDot,axis=1)
m6.fit_bounds(m6.get_bounds())
m6
```

[28]:

```
<folium.folium.Map at 0x1ef57b5b0a0>
```

[29]:

```
s_df=sars_latest
fig = px.treemap(s_df, names='Country', parents=[""]*len(s_df), values='Cases')
fig.add_trace(go.Treemap(
    labels = list(s_df.Country),
    parents = [""]*len(s_df),
    values =  list(s_df.Cases),
    textinfo = "label+value+percent parent",
    ),row = 1, col = 1)
fig.show()
```

[30]:

```
m4=folium.Map(tiles="Stamen Terrain",min_zoom=1.5)
url='https://raw.githubusercontent.com/python-visualization/folium/master/
 ↪examples/data'
country_shapes=f'{url}/world-countries.json'
folium.Choropleth(
    geo_data=country_shapes,
    min_zoom=2,
    name='SARS',
    data=sars_latest,
    columns=['Country','Cases'],
    key_on='feature.properties.name',
    fill_color='OrRd',
    nan_fill_color='white',
    legend_name='Total Confirmed SARS Cases',
).add_to(m4)
sars_final=pd.merge(sars_latest,coordinates,on='Country')
sars_final
```

[30]:

```
    index       Date         Country  Cases  Deaths  Recovered   latitude  \
0    2508 2003-07-11           China   5327     348       4941  35.861660
1    2507 2003-07-11          Canada    250      38        194  56.130366
2    2527 2003-07-11       Singapore    206      32        172   1.352083
3    2535 2003-07-11   United States     75       0         67  37.090240
4    2522 2003-07-11     Philippines     14       2         12  12.879721
5    2512 2003-07-11         Germany     10       0          9  51.165691
6    2520 2003-07-11        Mongolia      9       0          9  46.862496
7    2533 2003-07-11        Thailand      9       2          7  15.870032
8    2511 2003-07-11          France      7       1          6  46.227638
9    2519 2003-07-11        Malaysia      5       2          3   4.210484
10   2516 2003-07-11           Italy      4       0          4  41.871940
11   2534 2003-07-11  United Kingdom      4       0          4  55.378051
12   2514 2003-07-11           India      3       0          3  20.593684
13   2530 2003-07-11          Sweden      3       0          3  60.128161
14   2515 2003-07-11       Indonesia      2       0          2  -0.789275
15   2528 2003-07-11    South Africa      1       1          0 -30.559482
16   2531 2003-07-11     Switzerland      1       0          1  46.818188
17   2529 2003-07-11           Spain      1       0          1  40.463667
18   2506 2003-07-11          Brazil      1       0          1 -14.235004
19   2525 2003-07-11         Romania      1       0          1  45.943161
20   2517 2003-07-11          Kuwait      1       0          1  29.311660
21   2510 2003-07-11         Finland      1       0          1  61.924110
22   2509 2003-07-11        Colombia      1       0          1   4.570868
23   2521 2003-07-11     New Zealand      1       0          1 -40.900557

     longitude
0   104.195397
1  -106.346771
2   103.819836
3   -95.712891
4   121.774017
5    10.451526
6   103.846656
7   100.992541
```

```
8       2.213749
9     101.975766
10     12.567380
11     -3.435973
12     78.962880
13     18.643501
14    113.921327
15     22.937506
16      8.227512
17     -3.749220
18    -51.925280
19     24.966760
20     47.481766
21     25.748151
22    -74.297333
23    174.885971
```

[31]:

```
def plotDot(point):
    folium.CircleMarker(location=[point.latitude,point.longitude],
                        radius=5,
                        weight=2,
                        popup=[point.Country,point.Cases,point.Deaths],
                        fill_color='#000000').add_to(m4)
```

[32]:

```
sars_final.apply(plotDot,axis=1)
m4.fit_bounds(m4.get_bounds())
m4
```

[32]:

```
<folium.folium.Map at 0x1ef57e7fa00>
```

[33]:

```
m_df=mers_latest
fig = px.treemap(m_df, names='Country', parents=[""]*len(m_df),␣
 ↪values='Confirmed')
fig.add_trace(go.Treemap(
    labels = list(m_df.Country),
    parents = [""]*len(m_df),
    values =  list(m_df.Confirmed),
    textinfo = "label+value+percent parent",
    ),row = 1, col = 1)
fig.show()
```

[34]:

```
m5=folium.Map(tiles="Stamen Terrain",min_zoom=1.5)
url='https://raw.githubusercontent.com/python-visualization/folium/master/
 ↪examples/data'
country_shapes=f'{url}/world-countries.json'
folium.Choropleth(
    geo_data=country_shapes,
    min_zoom=2,
    name='MERS',
    data=mers_latest,
    columns=['Country','Confirmed'],
    key_on='feature.properties.name',
    fill_color='OrRd',
    nan_fill_color='white',
    legend_name='Total Confirmed MERS Cases',
).add_to(m5)
```

[34]:

```
<folium.features.Choropleth at 0x1ef58152700>
```

[35]:

```
mers_final=pd.merge(mers_latest,coordinates,on='Country')
mers_final
```

[35]:

```
                 Country  Confirmed   latitude   longitude
0           Saudi Arabia       2058  23.885942   45.079162
1   United Arab Emirates         87  23.424076   53.847818
2                 Jordan         28  30.585164   36.238414
3                   Oman         24  21.512583   55.923255
4                  Qatar         19  25.354826   51.183884
5                   Iran          6  32.427908   53.688046
6         United Kingdom          5  55.378051   -3.435973
7                 Kuwait          4  29.311660   47.481766
8                Tunisia          3  33.886917    9.537499
9                Germany          3  51.165691   10.451526
10              Thailand          3  15.870032  100.992541
11               Algeria          2  28.033886    1.659626
12           Philippines          2  12.879721  121.774017
13              Malaysia          2   4.210484  101.975766
14           Netherlands          2  52.132633    5.291266
15               Austria          2  47.516231   14.550072
16               Lebanon          2  33.854721   35.862285
17                France          2  46.227638    2.213749
18                 Italy          1  41.871940   12.567380
19                Greece          1  39.074208   21.824312
20                Turkey          1  38.963745   35.243322
21                 Egypt          1  26.820553   30.802498
22                 China          1  35.861660  104.195397
23               Bahrain          1  25.930414   50.637772
24                 Yemen          1  15.552727   48.516388
```

[36]:

```
def plotDot(point):
    folium.CircleMarker(location=[point.latitude,point.longitude],
                        radius=5,
                        weight=2,
                        popup=[point.Country,point.Confirmed],
                                    fill_color='#000000').add_to(m5)
```

[37]:
```
mers_final.apply(plotDot,axis=1)
m5.fit_bounds(m5.get_bounds())
m5
```

[37]:
```
<folium.folium.Map at 0x1ef581525b0>
```

[38]:
```
e_df=ebola_latest
fig = px.treemap(e_df, names='Country', parents=[""]*len(e_df), values='Cases')
fig.add_trace(go.Treemap(
    labels = list(e_df.Country),
    parents = [""]*len(e_df),
    values =  list(e_df.Cases),
    textinfo = "label+value+percent parent",
    ),row = 1, col = 1)
fig
```

[39]:
```
m3=folium.Map(tiles="Stamen Terrain",min_zoom=1.5)
url='https://raw.githubusercontent.com/python-visualization/folium/master/
 ↪examples/data'
country_shapes=f'{url}/world-countries.json'
folium.Choropleth(
    geo_data=country_shapes,
    min_zoom=2,
    name='EBOLA',
    data=ebola_latest,
    columns=['Country','Cases'],
    key_on='feature.properties.name',
    fill_color='OrRd',
    nan_fill_color='white',
    legend_name='Total Confirmed EBOLA Cases',
).add_to(m3)
```

[39]:
```
<folium.features.Choropleth at 0x1ef58152c40>
```

[40]:
```
ebola_final=pd.merge(ebola_latest,coordinates,on='Country')
ebola_final
```

[40]:
```
   index        Date       Country  Cases  Deaths   latitude  longitude
0   2375  2016-03-23  Sierra Leone  14122    3955   8.460555 -11.779889
1   2371  2016-03-23       Liberia  10680    4813   6.428055  -9.429499
2   2369  2016-03-23        Guinea   3804    2536   9.945587  -9.696645
3   2373  2016-03-23       Nigeria     20       8   9.081999   8.675277
4   2372  2016-03-23          Mali      8       6  17.570692  -3.996166
5   2370  2016-03-23         Italy      1       0  41.871940  12.567380
6   2374  2016-03-23       Senegal      1       0  14.497401 -14.452362
```

```
7   2376  2016-03-23          Spain      1        0  40.463667  -3.749220
8   2377  2016-03-23  United Kingdom      1        0  55.378051  -3.435973
```

[41]:
```
def plotDot(point):
    folium.CircleMarker(location=[point.latitude,point.longitude],
                        radius=5,
                        weight=2,
                        popup=[point.Country,point.Cases,point.Deaths],
                        fill_color='#000000').add_to(m3)
```

[42]:
```
ebola_final.apply(plotDot,axis=1)
m3.fit_bounds(m3.get_bounds())
m3
```

[42]:
```
<folium.folium.Map at 0x1ef583d4640>
```

[43]:
```
df = pd.read_csv(r'C:\Users\Asus\Desktop\Project\DV\DV_Project\PANDEMIC.csv')
df.sort_values('CONFIRMED', ascending =False)
```

[43]:
```
   EPIDEMIC  CONFIRMED   DEATHS  RECOVERED  FATILITY RATE  \
4  COVID-19  261442954  5215001  236167794       1.994699
0      H1N1     491382    18499     472883       3.764688
3     EBOLA      28646    11323      17323      39.527334
1      SARS       8437      813       7452       9.636127
2      MERS       2562      881       1681      34.387197

   NUMBER OF COUNTRIES FIRST REPORTED COUNTRY  YEAR
4                  224           Wuhan, China  2019
0                  213        Veracruz,Mexico  2009
3                   11                Guinea   2013
1                   29                  China  2002
2                   27   Jeddah, Saudi Arabia  2012
```

[44]:
```
color = ['aqua', 'lightcoral', 'yellowgreen', 'mediumpurple', 'mediumturquoise']
rcolor = ['mediumturquoise', 'mediumpurple', 'yellowgreen', 'lightcoral',␣
 ↪'aqua']
df2 = df.sort_values('DEATHS')
fig = px.bar(df2, x='EPIDEMIC', y='CONFIRMED',color = "EPIDEMIC")
fig.show()
```

[45]:
```
df2 = df.sort_values('CONFIRMED')
fig = px.bar(df2, x='EPIDEMIC', y='CONFIRMED',color = "EPIDEMIC", log_y=True)
fig.show()
```

[46]:
```
df2 = df.sort_values('DEATHS')
fig = px.bar(df2, x='EPIDEMIC', y='DEATHS',color = "EPIDEMIC", log_y=True)
fig.show()
```

[47]:
```
df2 = df.sort_values('FATILITY RATE')
fig = px.bar(df2, x='EPIDEMIC', y='FATILITY RATE',color = "EPIDEMIC")
fig.show()
```

[48]:
```
color = ['aqua', 'lightcoral', 'yellowgreen', 'mediumpurple', 'mediumturquoise']
rcolor = ['mediumturquoise', 'mediumpurple', 'yellowgreen', 'lightcoral',␣
 ↪'aqua']
df2 = df.sort_values('DEATHS')
fig = px.bar(df2, x='EPIDEMIC', y='DEATHS',color = "EPIDEMIC")
fig.show()
```

[49]:
```
df2 = df.sort_values('RECOVERED')
fig = px.bar(df2, x='EPIDEMIC', y='RECOVERED',color = "EPIDEMIC")
fig.show()
```

[50]:
```
df2 = df.sort_values('RECOVERED')
fig = px.bar(df2, x='EPIDEMIC', y='RECOVERED',color = "EPIDEMIC",log_y=True)
fig.show()
```

[51]:
```
df2 = df.sort_values('CONFIRMED')
fig = px.bar(df2, x='EPIDEMIC', y='CONFIRMED',color = "EPIDEMIC")
fig.show()
```

[52]:
```
df2 = df.sort_values('NUMBER OF COUNTRIES',ascending=False)
fig = px.bar(df2, x='NUMBER OF COUNTRIES', y='EPIDEMIC',color = "EPIDEMIC",␣
 ↪orientation='h')
fig.show()
```

[53]:
```
fig = px.scatter(df2, x="NUMBER OF COUNTRIES", y="FATILITY RATE",size="FATILITY␣
 ↪RATE", color="EPIDEMIC", hover_name="EPIDEMIC")
fig
```

[54]:
```
df2 = df.sort_values('FATILITY RATE')
fig = px.bar(df2, x='EPIDEMIC', y='FATILITY RATE',color = "EPIDEMIC")
fig.show()
```

[55]:
```
fig = px.scatter(df, x="CONFIRMED", y="DEATHS",size="FATILITY RATE",␣
 ↪color="EPIDEMIC", hover_name="EPIDEMIC", log_x=True, log_y=True)
fig
```

[56]:
```
fig = px.scatter(df, x="CONFIRMED", y="RECOVERED",size="FATILITY RATE",␣
 ↪color="EPIDEMIC", hover_name="EPIDEMIC", log_x=True, log_y=True)
fig
```

[57]:
```
df2 = df.sort_values('NUMBER OF COUNTRIES',ascending=False)
fig = px.scatter(df2, x='NUMBER OF COUNTRIES', y='EPIDEMIC')
```

```
fig.show()
```

[58]:

```
df2 = df.sort_values('DEATHS')
fig = px.area(df2, x='EPIDEMIC', y='CONFIRMED',color = "EPIDEMIC")
fig.show()
```

[59]:

```
df2 = df.sort_values('DEATHS')
fig = px.area(df2, x='EPIDEMIC', y='CONFIRMED',color = "EPIDEMIC",log_y=True)
fig.show()
```

[60]:

```
df2 = df.sort_values('FATILITY RATE')
fig = px.line(df2, x='EPIDEMIC', y='FATILITY RATE')
fig.show()
```

[61]:

```
df2 = df.sort_values('NUMBER OF COUNTRIES',ascending=False)
fig = px.area(df2, x='EPIDEMIC', y='NUMBER OF COUNTRIES')
fig.show()
```

[62]:

```
covid = pd.read_csv(r"C:
 ↪\Users\Asus\Desktop\Project\DV\DV_Project\owid-covid-data1.
 ↪csv",parse_dates=['date'])
covid = covid[['continent', 'location', 'date','total_cases',␣
 ↪'new_cases','new_cases_smoothed','total_deaths','new_deaths','total_vaccinations']]
# covid.columns = ['Date', 'Country', 'Cases', 'Deaths', 'Recovered']
covid = covid.groupby(['date', 'location'])
covid = covid.sum().reset_index()
covid['total_cases'] = covid['total_cases'].fillna(0)
covid['total_deaths'] = covid['total_deaths'].fillna(0)
covid['total_vaccinations'] = covid['total_vaccinations'].fillna(0)
covid_latest = covid[covid['date'] == max(covid['date'])].reset_index()
covid_latest=covid_latest.sort_values(by=['total_cases'],ascending=False)
```

[63]:

```
def plot_map(df, col, pal):
    df = df[df[col]>0]
    fig = px.choropleth(df, locations="location", locationmode='country names',
                  color=col, hover_name="location",
                  title=col, hover_data=[col], color_continuous_scale=pal)
#     fig.update_layout(coloraxis_showscale=False)
    fig.show()
```

[64]:

```
# # plot_map(covid, 'total_cases', [(0,"blue"), (1,"red")])
# plot_map(covid, 'total_cases', 'matter')
```

[65]:

```
# plot_map(covid, 'total_deaths', 'matter')
```

[66]:

```
# plot_map(covid, 'total_vaccinations', 'matter')
```

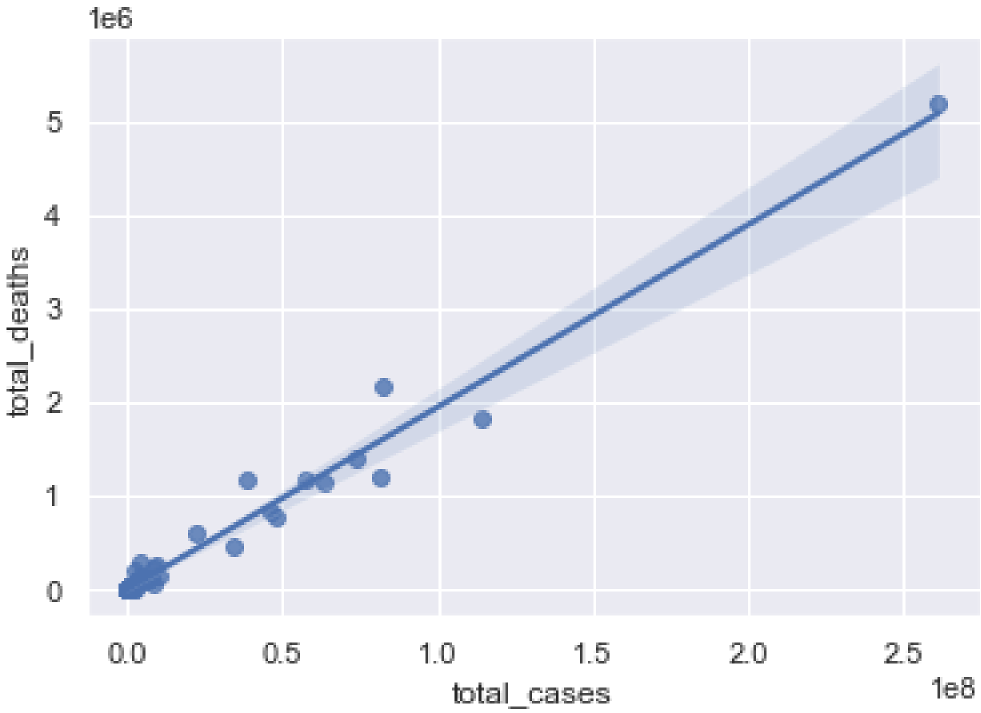
[67]:
```
# def plot_daywise(col, hue):
#     fig = px.bar(covid_latest, x="date", y=col, width=700,␣
 ↪color_discrete_sequence=[hue])
#     fig.update_layout(title=col, xaxis_title="", yaxis_title="")
#     fig.show()
```

[68]:
```
# def plot_daywise_line(col, hue):
#     fig = px.line(covid_latest, x="date", y=col, width=700,␣
 ↪color_discrete_sequence=[hue])
#     fig.update_layout(title=col, xaxis_title="", yaxis_title="")
#     fig.show()
```

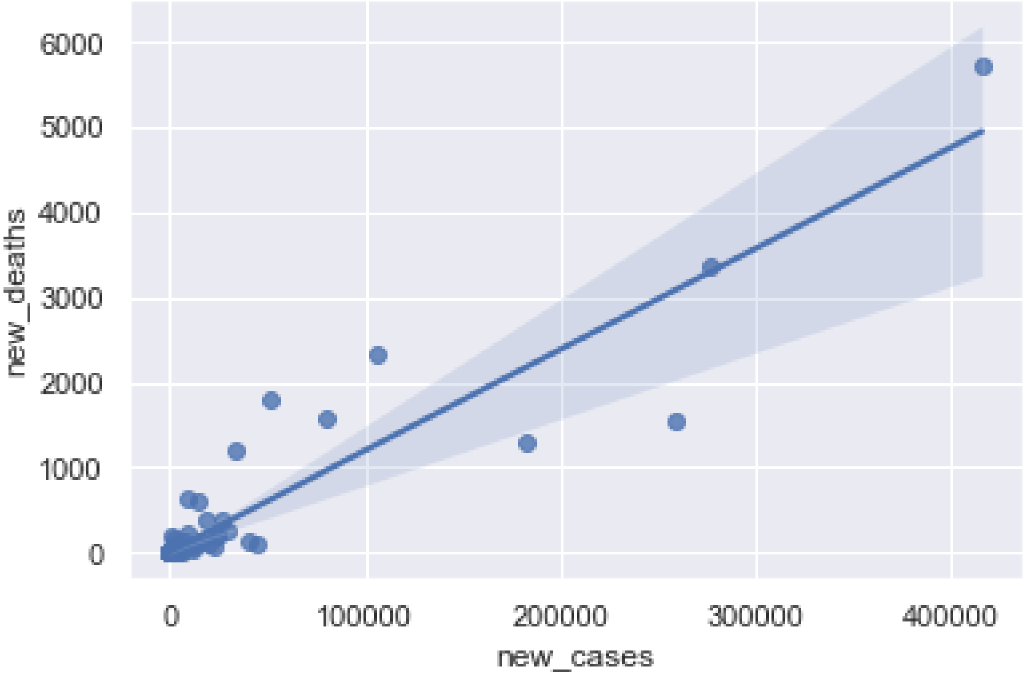
[69]:
```
# temp = covid_latest['date'].groupby('date')['total_cases', 'total_deaths'].
 ↪sum().reset_index()
# temp = temp.melt(id_vars="date", value_vars=['total_cases',␣
 ↪'total_deaths'],var_name='Case', value_name='Count')
# temp.head()

# fig = px.area(temp, x="date", y="Count", color='Case', height=600, width=700,
#               title='Cases over time', color_discrete_sequence = [rec, dth,␣
 ↪act])
# fig.update_layout(xaxis_rangeslider_visible=True)
# fig.show()
```

[70]:
```
sns.set_theme(color_codes=True)
```

[71]:
```
# covid = sns.load_dataset(r"C:
 ↪\Users\Asus\Desktop\Project\DV\DV_Project\owid-covid-data.csv")
sns.regplot(x="total_cases", y="total_deaths", data=covid_latest);
```



[72]:
```
sns.regplot(x="new_cases", y="new_deaths", data=covid_latest);
```



[73]:

```
temp = covid[['date', 'total_cases', 'total_deaths']]
temp = temp.melt(id_vars='date', value_vars=[ 'total_cases','total_deaths'],
                 var_name='Variable', value_name='Count')
px.line(temp, x='date', y='Count', color='Variable')
```

[74]:

```
temp = covid[['date', 'total_cases', 'total_deaths']]
temp = temp.melt(id_vars='date', value_vars=[ 'total_cases','total_deaths'],
                 var_name='Variable', value_name='Count')
px.line(temp, x='date', y='Count', color='Variable',log_y=True)
```

[75]:

```
fig = px.scatter(covid_latest, x='total_cases', y='total_deaths',␣
 ↪color='total_cases',
height=700, hover_name='location',
title='Confirmed vs Deaths',
color_discrete_sequence=px.colors.qualitative.Vivid)
fig.update_traces(textposition='top center')
# fig.update_layout(showlegend=False)
# fig.update_layout(xaxis_rangeslider_visible=True)
fig.show()
```

[76]:

```
fig = px.scatter(covid_latest, x='total_cases', y='total_deaths',␣
 ↪color='total_cases',
height=700, hover_name='location', log_x=True, log_y=True,
title='Confirmed vs Deaths',
color_discrete_sequence=px.colors.qualitative.Vivid)
fig.update_traces(textposition='top center')
# fig.update_layout(showlegend=False)
# fig.update_layout(xaxis_rangeslider_visible=True)
fig.show()
```

[77]:

```
def plot_bubble(col, pal):
    temp = covid_latest[covid_latest[col]>0].sort_values('location',␣
 ↪ascending=False)
    fig = px.scatter(temp, x='date', y='location', size=col, color=col,␣
 ↪height=3000,
                    color_continuous_scale=pal)
    fig.update_layout(yaxis = dict(dtick = 1))
    fig.update(layout_coloraxis_showscale=False)
    fig.show()
```

[78]:

```
plot_bubble('new_cases', 'Viridis')
```

[79]:

```
def plot_treemap(col):
    fig = px.treemap(covid_latest, path=["location"], values=col, height=700,
                 title=col, color_discrete_sequence = px.colors.qualitative.
 ↪Dark2)
```

```
    fig.data[0].textinfo = 'label+text+value'
    fig.show()
```

[80]:
```
plot_treemap('total_cases')
```

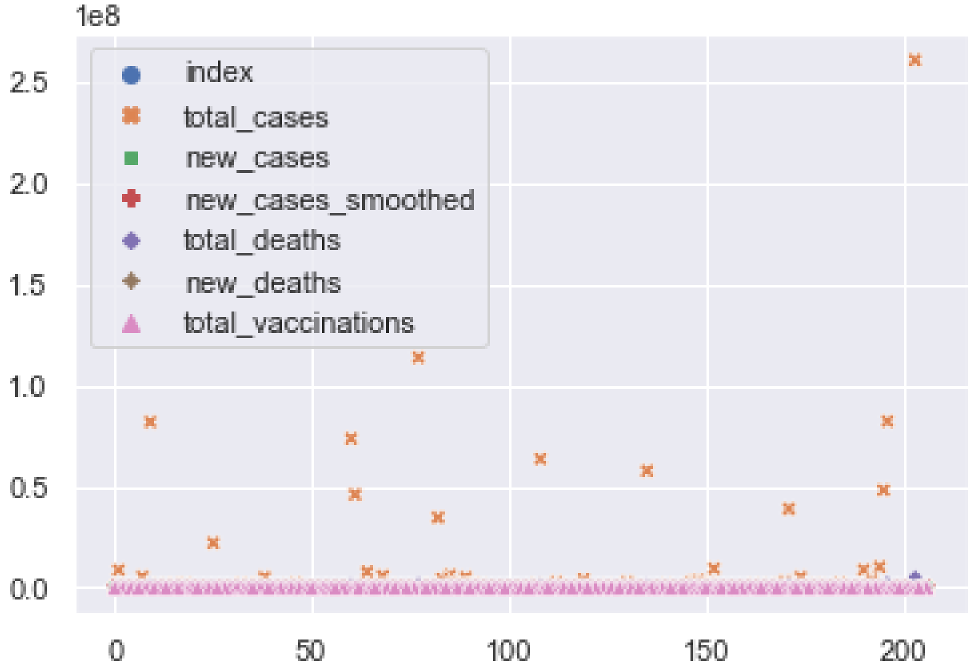
[81]:
```
plot_treemap('total_deaths')
```

[82]:
```
# plot_treemap('total_vaccinations')
```

[83]:
```
fig = px.scatter(covid_latest.sort_values('total_deaths', ascending=False).
 ↪iloc[:20, :],
                 x='total_cases', y='total_deaths', color='location',␣
 ↪size='total_cases',
                 height=700, text='location', log_x=True, log_y=True,
                 title='Deaths vs Confirmed (Scale is in log10)')
fig.update_traces(textposition='top center')
fig.update_layout(showlegend=False)
fig.update_layout(xaxis_rangeslider_visible=True)
fig.show()
```

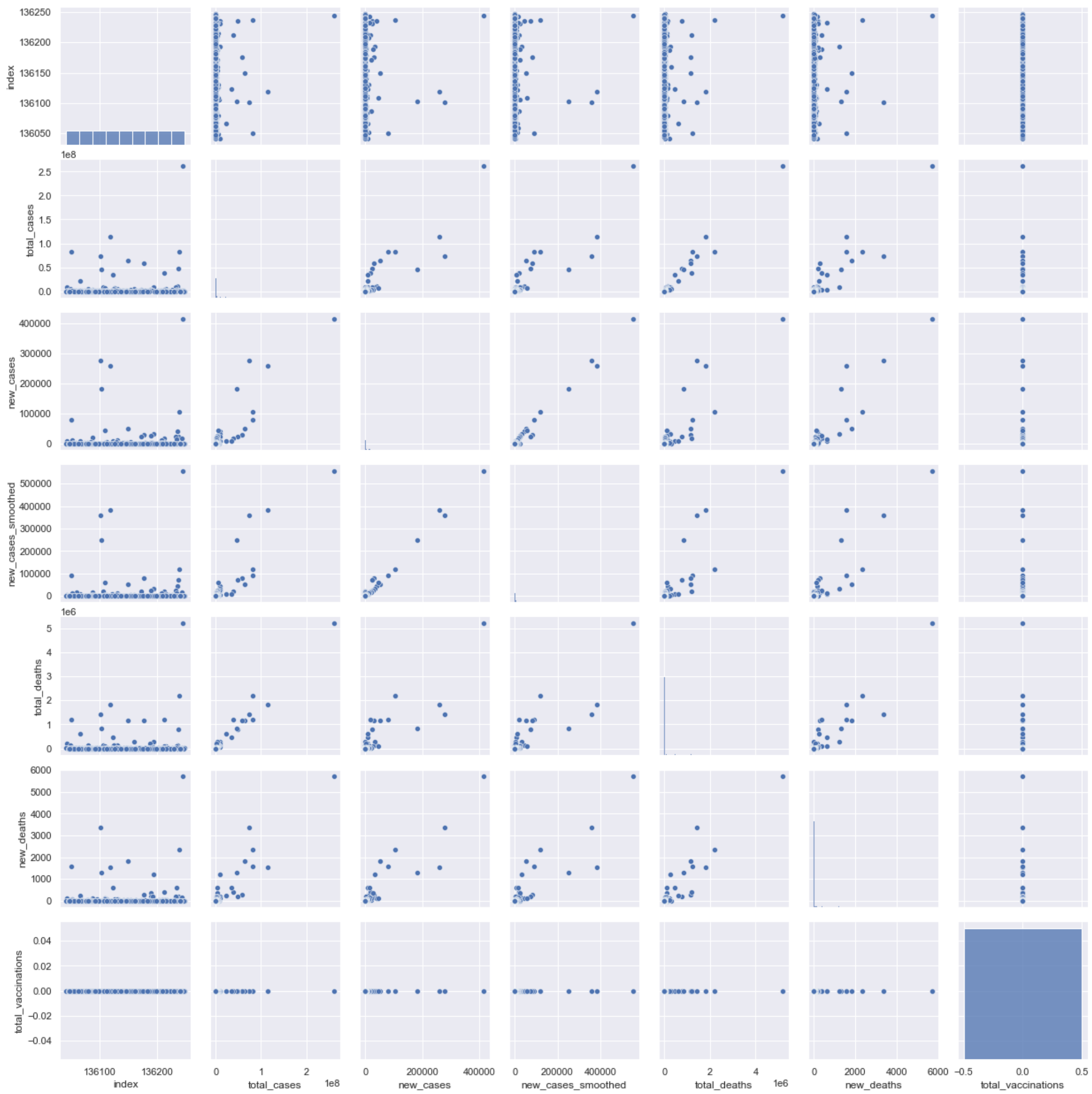
[84]:
```
sns.scatterplot(data=covid_latest)
```

[84]:
```
<AxesSubplot:>
```



```
[85]: sns.pairplot(data=covid_latest)
```

```
[85]: <seaborn.axisgrid.PairGrid at 0x1ef56423a30>
```



```
[ ]:
```