# MSDS 422 ASSIGNMENT 1 – EXPLORING AND VISUALIZING DATA

Anaswar Jayakumar

Dr. Lawrence Fulton

**Data preparation**

The original data set contained negative values for cases which would affect how the case rate was calculated and in turn the log of the case rate since negative numbers aren't included in the domain of the log function. Likewise, there were also negative values in the deaths column of the original data set as well and such values needed to be taken care before proceeding to analyzing the data. I also needed to check the presence of null values in the original data set and remove them accordingly. This was especially important as null values could potentially interfere with later analysis. Specifically, I noticed the presence of null values in the popData2019 and the countryterritoryCode columns which would affect the calculation of the case and death rates. Min-max scaling was performed on the COVID-19 dataset in order to normalize the variables and standard scaling was performed in order to scale the variables to a standard range. Lastly, I created a time series for the cases, deaths, the cumulative number of cases, the case rate, the death rate, and the log of the case and death rates. When calculating the case rate and death rate, I used 100,000 as the denominator since the cumulative case count used 100,000 as well.

**Data exploration/Data visualization**

After preparing the dataset for analysis, I then analyzed the data to better understand the spread of COVID-19 as well as its fatality rate. To do this, I first created two additional columns, case_rate and death_rate and then took the log of the case_rate and death_rate columns. When calculating the log of the case and death rates, I added 0.1 since the log of 0 is undefined but case rate and death rate happened to have 0 as valid values. Adding a small constant ensures that the domain of the log is satisfied while not impacting the data significantly. Once, the case rate, death rate, log case rate and log death rate columns were added, I then generated the descriptive statistics of the updated COVID-19 dataset. To better visualize and understand the data, I generated a scatter plot modeling the case rate versus the death rate, histograms modeling the distribution of the case rate and death rate as well as the log of the case rate and death rate, time series graphs for the cases, deaths, the cumulative number of cases in the past 14 days, the case and death rate, and the log of the case and death rate, correlation matrices of the COVID-19 data and the time series, and boxplots of the case and death rates as well as the boxplots of the log of the case and death rates. The plots helped better understand the data as a whole as well as the spread of the disease and whether or not it is a predictor of the fatality rate of the disease. Because the distribution of the case and death rate seemed exponential, it made sense to perform a log transformation on these columns. I chose log2.

**Data scaling and comparisons**

Min-max scaling was performed on the final COVID-19 dataset in order to normalize the variables and standard scaling was performed in order to scale the variables to a standard range. For both the min-max and standard scaling, I chose the columns cases, deaths, case_rate, death_rate, log_case_rate, and log_death_rate. Scaling methods such as the min-max and standard scaling methods preserve the overall shape of the distribution which is exponential in the case of cases, deaths, case_rate, and death_rate. In addition, min-max scaling was performed on the final COVID-19 dataset since the variables in the dataset are measured at different scales and therefore contribute unequally to the fitting of the model, thereby causing an inherent bias in the final model. One difference between the two methods is the range of the data and this is because the formula used in standard scaling involves dividing by the standard deviation as opposed to the min-max scaling which involves dividing by the range. The distributions generated by both methods are similar and comparable to the original distribution.

**Insights from analysis**

The descriptive statistics did provide some insight into the spread of COVID-19 as well as its fatality rate. I looked at the descriptive statistics of the original COVID-19 data, the descriptive statistics of the prepared COVID-19 data, and the descriptive statistics of the time series data. For the updated COVID-19 dataset, I noticed that the mean of the cases is larger than the upper quartile, the mean of the deaths is larger than the upper quartile as well, and that the standard deviation for the cases and deaths was significantly large as well. The boxplots of the case and death rates and the log of the case and death rates show most of the points outside of the $75^{th}$ percentile. This makes sense as the distribution is exponential. The histogram confirms the shape of the distribution as well. This of course makes sense since cases of COVID-19 have experienced exponential growth and likewise, deaths from COVID-19 have also experienced exponential growth. For the prepared COVID-19 dataset, I generated a correlation matrix. I ignored columns such as day, month, and year and only looked at columns such as cases, deaths, total population, case rate and death rate. I expected case and deaths to be highly correlated and this was reflected in the correlation matrix, however the correlation between cases and population data was weaker than expected. Potential reasons include countries experiencing an unequal number of cases and inaccuracy when reporting the actual number of cases and deaths. Even though the cases and deaths are highly correlated, the correlation between case rate and death rate seems to be weaker. Since the data was measured over a year, it made sense to model a time series for the cases, deaths, case rate, death rate, and the cumulative number of cases. The time series provided better insight into how the cases, deaths, case rate, death rate, and cumulative number of cases changes over time. The time series of the log of the death rate shows a peak followed by a decrease even though the log of the case rate stays relatively constant.

# MSDS 422 - COVID 19 EDA (final)

April 4, 2021

## 1 Data Exploration, Data Preparation, and Data Visualization

Import all required Python libraries such as pandas, numpy, matplotlib, seaborn, etc

[1]:
```python
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report, roc_auc_score
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVC
```

Import COVID-19 data which is a CSV file

[2]:
```python
covid_19_dataset = '/Users/anaswarjayakumar/Desktop/Desktop - Anaswar's MacBook␣
↪Pro/COVID-19 Data.csv'

covid_19_df = pd.read_csv(covid_19_dataset, sep = ",")

covid_19_df
```

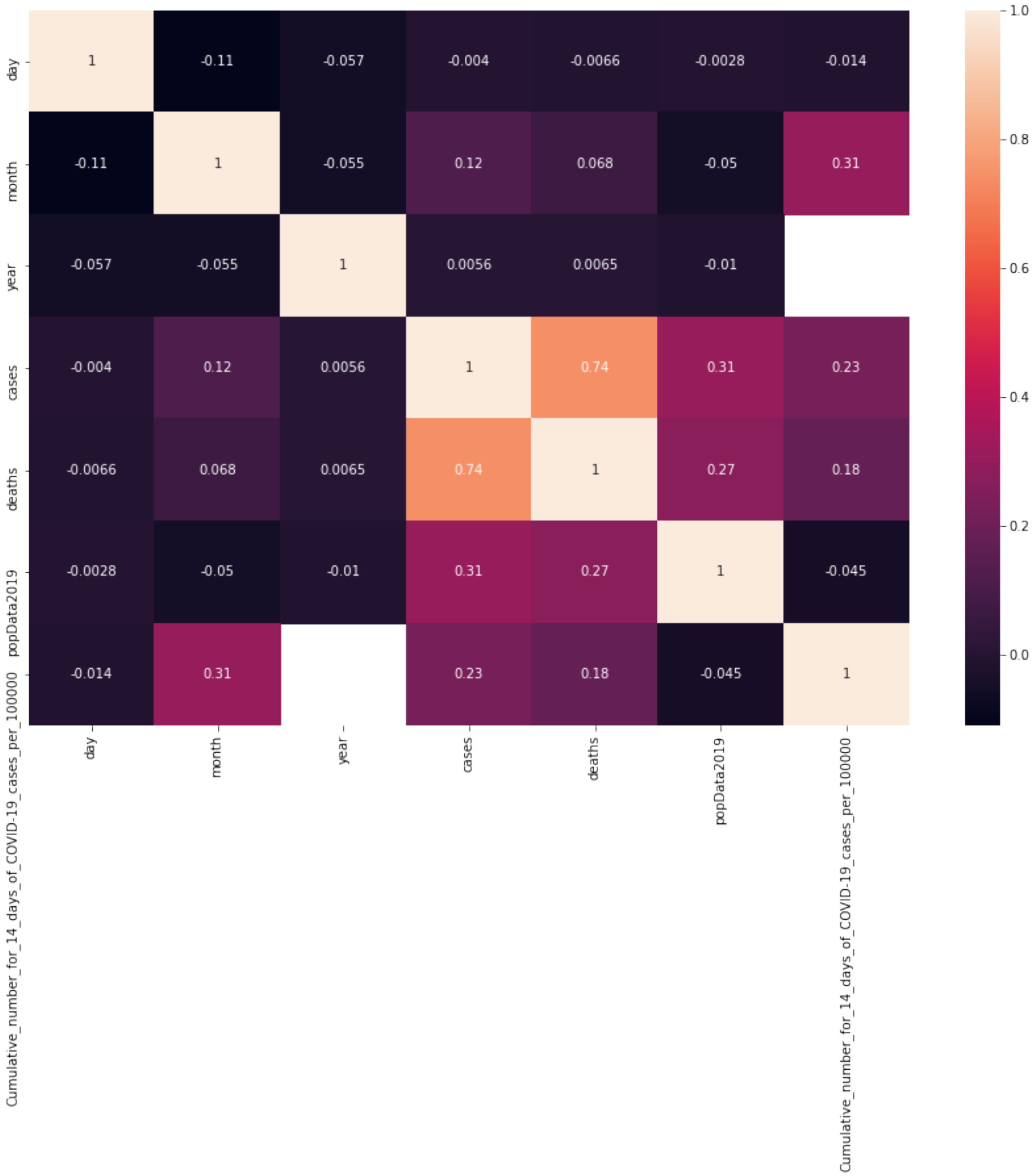
[2]:
```
          dateRep  day  month  year  cases  deaths   countriesAndTerritories  \
0      14/12/2020   14     12  2020    746       6               Afghanistan
1      13/12/2020   13     12  2020    298       9               Afghanistan
2        12/12/20   12     12  2020    113      11               Afghanistan
3        11/12/20   11     12  2020     63      10               Afghanistan
4        10/12/20   10     12  2020    202      16               Afghanistan
…             …  …    …     …    …      …                           …
61895  31/12/2019   31     12  2019      0       0  United_States_of_America
61896  24/03/2020   24      3  2020      0       1                  Zimbabwe
61897  31/12/2019   31     12  2019      0       0                   Vietnam
```

```
61898  22/03/2020   22       3  2020       1        0                  Zimbabwe
61899  21/03/2020   21       3  2020       1        0                  Zimbabwe

      geoId countryterritoryCode  popData2019 continentExp  \
0        AF                  AFG   38041757.0         Asia
1        AF                  AFG   38041757.0         Asia
2        AF                  AFG   38041757.0         Asia
3        AF                  AFG   38041757.0         Asia
4        AF                  AFG   38041757.0         Asia
…      …                   …           …            …
61895    US                  USA  329064917.0      America
61896    ZW                  ZWE   14645473.0       Africa
61897    VN                  VNM   96462108.0         Asia
61898    ZW                  ZWE   14645473.0       Africa
61899    ZW                  ZWE   14645473.0       Africa

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000
0                                               9.013779
1                                               7.052776
2                                               6.868768
3                                               7.134266
4                                               6.968658
…                                                    …
61895                                                NaN
61896                                                NaN
61897                                                NaN
61898                                                NaN
61899                                                NaN

[61900 rows x 12 columns]
```

Generate heat map for COVID-19 data frame

[3]:
```
plt.figure(figsize=(15, 10))

sns.heatmap(covid_19_df.corr(), annot=True)
```

[3]:
```
<AxesSubplot:>
```




Get descriptive statistics for COVID-19 data frame

[4]:
```
covid_19_df.describe()
```

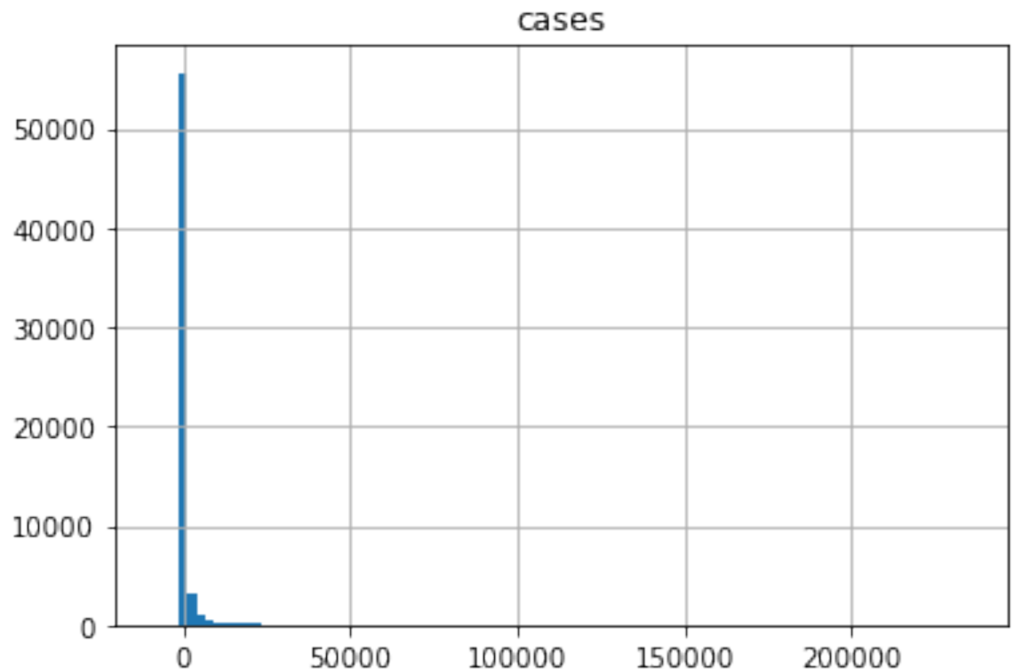
[4]:

| | day | month | year | cases | deaths \ |
|---|---|---|---|---|---|
| count | 61900.000000 | 61900.000000 | 61900.000000 | 61900.000000 | 61900.000000 |
| mean | 15.628934 | 7.067157 | 2019.998918 | 1155.147237 | 26.055460 |
| std | 8.841582 | 2.954776 | 0.032882 | 6779.224479 | 131.227055 |
| min | 1.000000 | 1.000000 | 2019.000000 | -8261.000000 | -1918.000000 |

```
25%        8.000000      5.000000   2020.000000       0.000000      0.000000
50%       15.000000      7.000000   2020.000000      15.000000      0.000000
75%       23.000000     10.000000   2020.000000     273.000000      4.000000
max       31.000000     12.000000   2020.000000  234633.000000   4928.000000

        popData2019  \
count  6.177700e+04
mean   4.098770e+07
std    1.531294e+08
min    8.150000e+02
25%    1.293120e+06
50%    7.169456e+06
75%    2.851583e+07
max    1.433784e+09

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000
count                                       59021.000000
mean                                           66.320586
std                                           162.329240
min                                          -147.419587
25%                                             0.757526
50%                                             6.724045
75%                                            52.572719
max                                          1900.836210
```

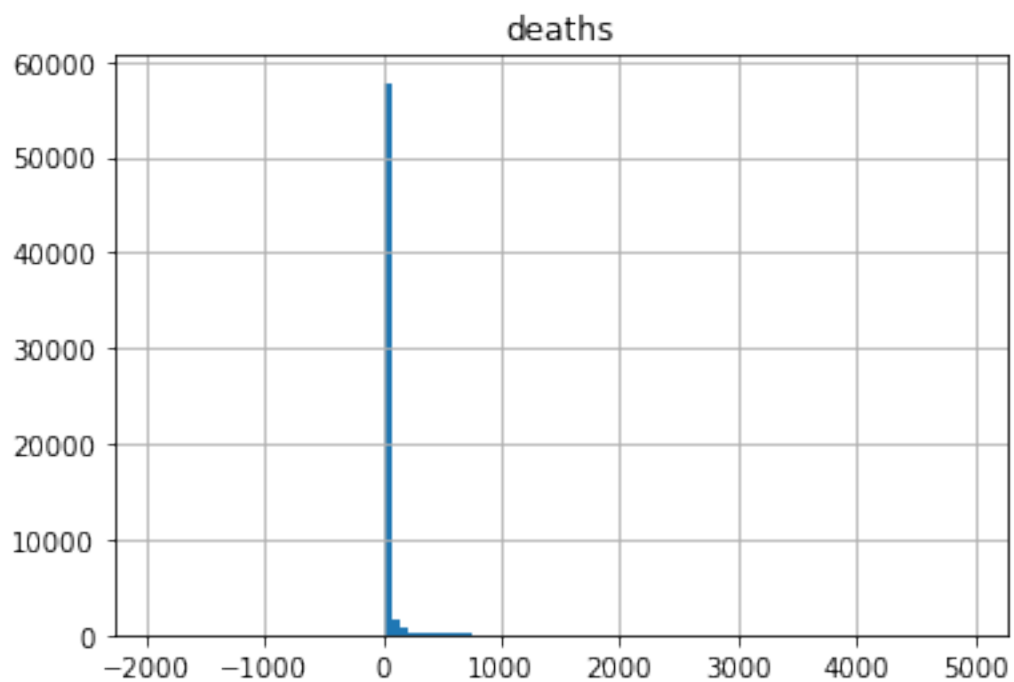
[5]:
```
covid_19_df.hist('cases', bins = 100)
```

[5]:
```
array([[<AxesSubplot:title={'center':'cases'}>]], dtype=object)
```




[6]:
```
covid_19_df.hist('deaths', bins = 100)
```

[6]:
```
array([[<AxesSubplot:title={'center':'deaths'}>]], dtype=object)
```



Convert the values in the geoId column to string and check which rows in the data frame contain null values in the popData2019 column

[7]:
```
covid_19_df.geoId = covid_19_df.geoId.astype(str)

covid_19_df[covid_19_df['popData2019'].isnull()]
```

[7]:
```
          dateRep  day  month  year  cases  deaths  \
125        1/1/20    1      1  2020      0       0
332        1/2/20    1      2  2020      0       0
942        1/3/20    1      3  2020      0       0
1838      1/11/20    1     11  2020      0       0
1888      1/12/20    1     12  2020      0       0
…            …   …     …    …     …      …
60858  30/11/2020   30     11  2020      0       0
60871  31/01/2020   31      1  2020      0       0
60877  17/10/2020   17     10  2020      1       0
61557  31/10/2020   31     10  2020      0       0
61610  31/12/2019   31     12  2019      0       0

                         countriesAndTerritories     geoId  \
125    Cases_on_an_international_conveyance_Japan  JPG11668
332    Cases_on_an_international_conveyance_Japan  JPG11668
942    Cases_on_an_international_conveyance_Japan  JPG11668
1838                             Wallis_and_Futuna        WF
1888                             Wallis_and_Futuna        WF
…                                              …       …
60858                            Wallis_and_Futuna        WF
60871  Cases_on_an_international_conveyance_Japan  JPG11668
60877                            Wallis_and_Futuna        WF
61557                            Wallis_and_Futuna        WF
61610  Cases_on_an_international_conveyance_Japan  JPG11668

      countryterritoryCode  popData2019 continentExp  \
125                    NaN          NaN        Other
332                    NaN          NaN        Other
942                    NaN          NaN        Other
1838                   NaN          NaN      Oceania
1888                   NaN          NaN      Oceania
…                     …          …          …
60858                  NaN          NaN      Oceania
60871                  NaN          NaN        Other
60877                  NaN          NaN      Oceania
61557                  NaN          NaN      Oceania
61610                  NaN          NaN        Other
```

```
      Cumulative_number_for_14_days_of_COVID-19_cases_per_100000
125                                                         NaN
332                                                         NaN
942                                                         NaN
1838                                                        NaN
1888                                                        NaN
…                                                         …
60858                                                       NaN
60871                                                       NaN
60877                                                       NaN
61557                                                       NaN
61610                                                       NaN

[123 rows x 12 columns]
```

Check which rows in the data frame contain null values in the countryterritoryCode column

[8]:
```
covid_19_df[covid_19_df['countryterritoryCode'].isnull()]
```

[8]:
```
         dateRep  day  month  year  cases  deaths  \
125       1/1/20    1      1  2020      0       0
332       1/2/20    1      2  2020      0       0
942       1/3/20    1      3  2020      0       0
1838     1/11/20    1     11  2020      0       0
1888     1/12/20    1     12  2020      0       0
…            …   …    …     …     …      …
60858  30/11/2020   30     11  2020      0       0
60871  31/01/2020   31      1  2020      0       0
60877  17/10/2020   17     10  2020      1       0
61557  31/10/2020   31     10  2020      0       0
61610  31/12/2019   31     12  2019      0       0

                          countriesAndTerritories     geoId  \
125    Cases_on_an_international_conveyance_Japan  JPG11668
332    Cases_on_an_international_conveyance_Japan  JPG11668
942    Cases_on_an_international_conveyance_Japan  JPG11668
1838                            Wallis_and_Futuna        WF
1888                            Wallis_and_Futuna        WF
…                                             …        …
60858                           Wallis_and_Futuna        WF
60871  Cases_on_an_international_conveyance_Japan  JPG11668
60877                           Wallis_and_Futuna        WF
61557                           Wallis_and_Futuna        WF
61610  Cases_on_an_international_conveyance_Japan  JPG11668

      countryterritoryCode  popData2019 continentExp  \
125                    NaN          NaN        Other
332                    NaN          NaN        Other
```

```
942                      NaN          NaN        Other
1838                     NaN          NaN      Oceania
1888                     NaN          NaN      Oceania
…                      …          …          …
60858                    NaN          NaN      Oceania
60871                    NaN          NaN        Other
60877                    NaN          NaN      Oceania
61557                    NaN          NaN      Oceania
61610                    NaN          NaN        Other

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000
125                                                          NaN
332                                                          NaN
942                                                          NaN
1838                                                         NaN
1888                                                         NaN
…                                                          …
60858                                                        NaN
60871                                                        NaN
60877                                                        NaN
61557                                                        NaN
61610                                                        NaN

[123 rows x 12 columns]
```

Create a new COVID-19 data frame such that it doesnt contain any null values in the popData2019 and it doesnt contain any negatative values in the cases and deaths columns. Having null and negative values will interfere with later analysis of the COVID-19 data frame

[9]:

```
clean_covid_19_df = covid_19_df[~covid_19_df['popData2019'].isnull()]

clean_covid_19_df = clean_covid_19_df[(clean_covid_19_df['cases'] >= 0) &␣
↪(clean_covid_19_df['deaths'] >= 0)]

clean_covid_19_df
```

[9]:

```
         dateRep  day  month  year  cases  deaths   countriesAndTerritories  \
0      14/12/2020   14     12  2020    746       6               Afghanistan
1      13/12/2020   13     12  2020    298       9               Afghanistan
2        12/12/20   12     12  2020    113      11               Afghanistan
3        11/12/20   11     12  2020     63      10               Afghanistan
4        10/12/20   10     12  2020    202      16               Afghanistan
…            …   …    …    …     …      …                     …
61895  31/12/2019   31     12  2019      0       0  United_States_of_America
61896  24/03/2020   24      3  2020      0       1                  Zimbabwe
61897  31/12/2019   31     12  2019      0       0                   Vietnam
61898  22/03/2020   22      3  2020      1       0                  Zimbabwe
61899  21/03/2020   21      3  2020      1       0                  Zimbabwe
```

```
      geoId countryterritoryCode  popData2019 continentExp  \
0        AF                  AFG   38041757.0         Asia
1        AF                  AFG   38041757.0         Asia
2        AF                  AFG   38041757.0         Asia
3        AF                  AFG   38041757.0         Asia
4        AF                  AFG   38041757.0         Asia
…      …                   …          …            …
61895    US                  USA  329064917.0      America
61896    ZW                  ZWE   14645473.0       Africa
61897    VN                  VNM   96462108.0         Asia
61898    ZW                  ZWE   14645473.0       Africa
61899    ZW                  ZWE   14645473.0       Africa

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000
0                                               9.013779
1                                               7.052776
2                                               6.868768
3                                               7.134266
4                                               6.968658
…                                                    …
61895                                                NaN
61896                                                NaN
61897                                                NaN
61898                                                NaN
61899                                                NaN

[61753 rows x 12 columns]
```

Calculate the death rate and case rate and create two such columns in the new COVID-19 data frame. In addition calculate the log of the case rate and death rate and create two such columns as well. When calculating the log of the case and death rates, I added 0.1 in order to ensure that the log function doesnt deal with any potential negative numbers as negative numbers are not included in the domain of the log function

[10]:
```
clean_covid_19_df['death_rate'] = clean_covid_19_df['deaths'] /␣
↪(clean_covid_19_df['popData2019'] / 100000)

clean_covid_19_df['case_rate'] = clean_covid_19_df['cases'] /␣
↪(clean_covid_19_df['popData2019'] / 100000)

clean_covid_19_df['log_case_rate'] = np.log2(clean_covid_19_df['case_rate'] + 0.
↪1)

clean_covid_19_df['log_death_rate'] = np.log2(clean_covid_19_df['death_rate'] +␣
↪0.1)
```

```
clean_covid_19_df
```

[10]:

```
          dateRep  day  month  year  cases  deaths    countriesAndTerritories  \
0      14/12/2020   14     12  2020    746       6                Afghanistan
1      13/12/2020   13     12  2020    298       9                Afghanistan
2        12/12/20   12     12  2020    113      11                Afghanistan
3        11/12/20   11     12  2020     63      10                Afghanistan
4        10/12/20   10     12  2020    202      16                Afghanistan
…             …  …    …    …     …      …                          …
61895  31/12/2019   31     12  2019      0       0  United_States_of_America
61896  24/03/2020   24      3  2020      0       1                   Zimbabwe
61897  31/12/2019   31     12  2019      0       0                    Vietnam
61898  22/03/2020   22      3  2020      1       0                   Zimbabwe
61899  21/03/2020   21      3  2020      1       0                   Zimbabwe

      geoId countryterritoryCode  popData2019 continentExp  \
0        AF                  AFG   38041757.0         Asia
1        AF                  AFG   38041757.0         Asia
2        AF                  AFG   38041757.0         Asia
3        AF                  AFG   38041757.0         Asia
4        AF                  AFG   38041757.0         Asia
…      …                    …           …          …
61895    US                  USA  329064917.0      America
61896    ZW                  ZWE   14645473.0       Africa
61897    VN                  VNM   96462108.0         Asia
61898    ZW                  ZWE   14645473.0       Africa
61899    ZW                  ZWE   14645473.0       Africa

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000  death_rate  \
0                                               9.013779            0.015772
1                                               7.052776            0.023658
2                                               6.868768            0.028916
3                                               7.134266            0.026287
4                                               6.968658            0.042059
…                                                    …                 …
61895                                                NaN            0.000000
61896                                                NaN            0.006828
61897                                                NaN            0.000000
61898                                                NaN            0.000000
61899                                                NaN            0.000000

       case_rate  log_case_rate  log_death_rate
0       1.961003       1.043347       -3.110640
1       0.783350      -0.178943       -3.015570
2       0.297042      -1.332636       -2.955501
3       0.165607      -1.912632       -2.985223
4       0.530995      -0.664298       -2.815437
```

```
…       …           …             …
61895   0.000000    -3.321928     -3.321928
61896   0.000000    -3.321928     -3.226638
61897   0.000000    -3.321928     -3.321928
61898   0.006828    -3.226638     -3.321928
61899   0.006828    -3.226638     -3.321928

[61753 rows x 16 columns]
```

Get the descriptive statistics of the new COVID-19 data frame

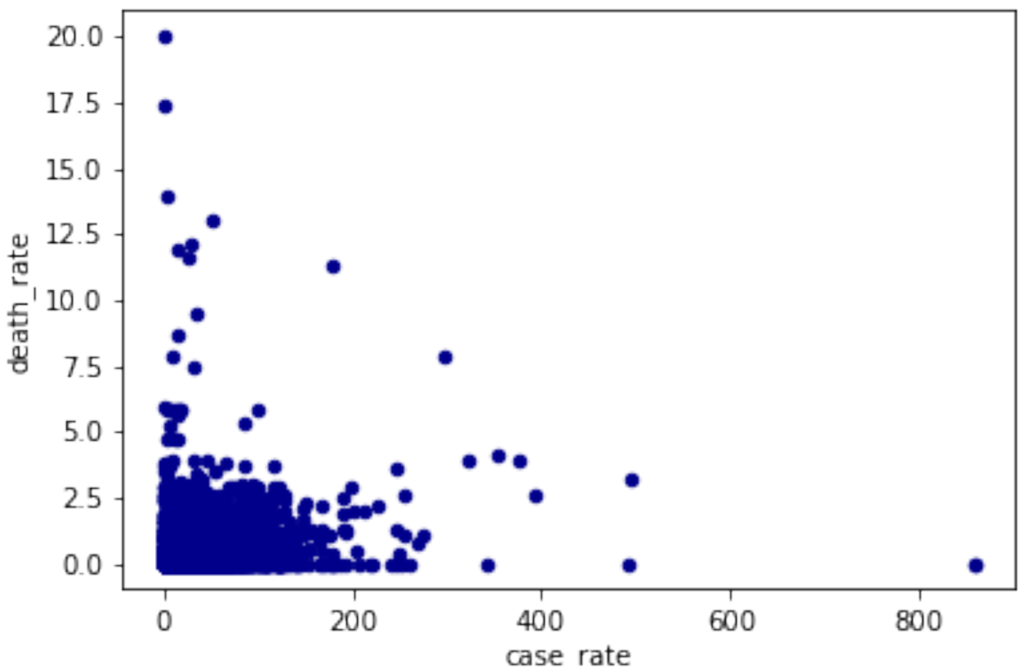
[11]:
```
clean_covid_19_df.describe()
```

[11]:
```
                day         month          year          cases        deaths  \
count  61753.000000  61753.000000  61753.000000   61753.000000  61753.000000
mean      15.629945      7.069227   2019.998931    1158.071689     26.083607
std        8.841257      2.950149      0.032675    6786.916211    130.238403
min        1.000000      1.000000   2019.000000       0.000000      0.000000
25%        8.000000      5.000000   2020.000000       0.000000      0.000000
50%       15.000000      7.000000   2020.000000      16.000000      0.000000
75%       23.000000     10.000000   2020.000000     276.000000      4.000000
max       31.000000     12.000000   2020.000000  234633.000000   4928.000000

        popData2019  \
count  6.175300e+04
mean   4.099461e+07
std    1.531581e+08
min    8.150000e+02
25%    1.293120e+06
50%    7.169456e+06
75%    2.851583e+07
max    1.433784e+09

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000  \
count                                       58997.000000
mean                                           66.329444
std                                           162.354715
min                                          -147.419587
25%                                             0.757526
50%                                             6.724045
75%                                            52.559960
max                                          1900.836210

         death_rate     case_rate  log_case_rate  log_death_rate
count  61753.000000  61753.000000   61753.000000    61753.000000
mean       0.081945      4.847870      -0.601635       -2.889706
std        0.312673     14.933306       2.841068        0.849840
min        0.000000      0.000000      -3.321928       -3.321928
```

```
25%       0.000000      0.000000     -3.321928     -3.321928
50%       0.000000      0.260533     -1.471798     -3.321928
75%       0.034235      3.124746      1.689186     -2.897167
max      20.036065    858.895706      9.746507      4.331710
```

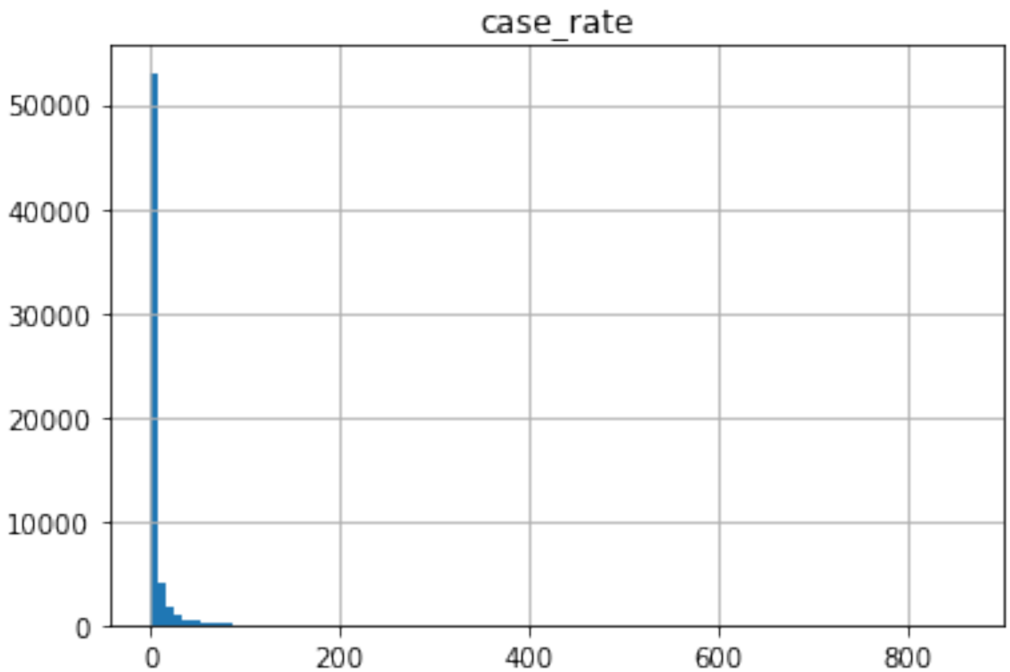
Generate a scatter plot of the updated COVID-19 data frame where the case rate is on the x-axis and the death rate is on the y-axis

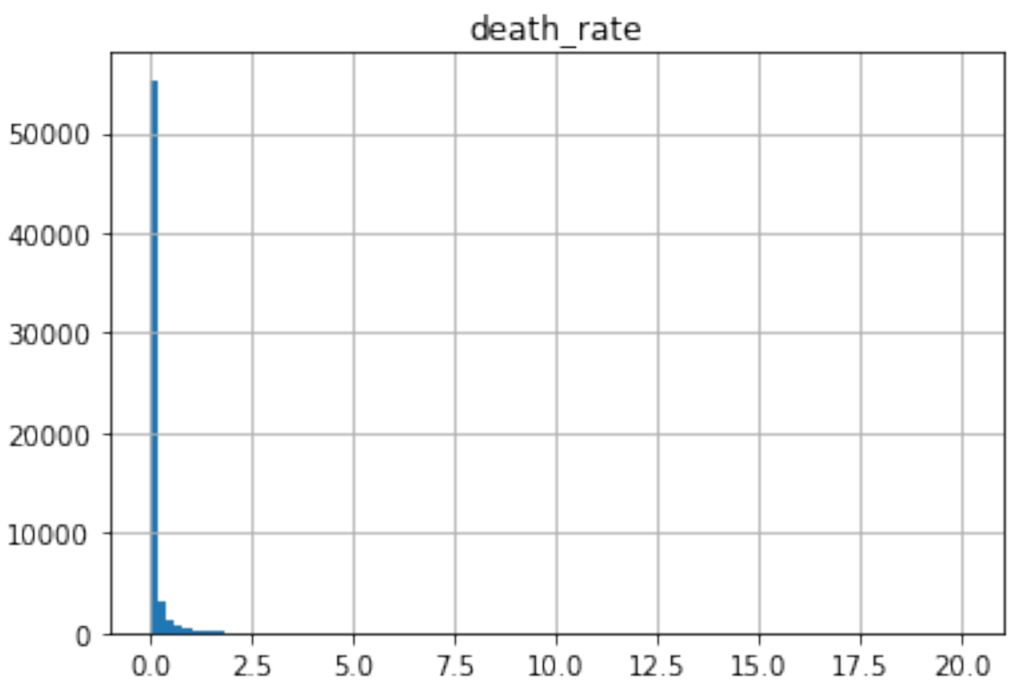
[12]:
```
covid_19_df_plot = clean_covid_19_df.plot.scatter(x = 'case_rate', y =␣
↪'death_rate', c='DarkBlue')
```




Generate histograms for the case rate and death rate.

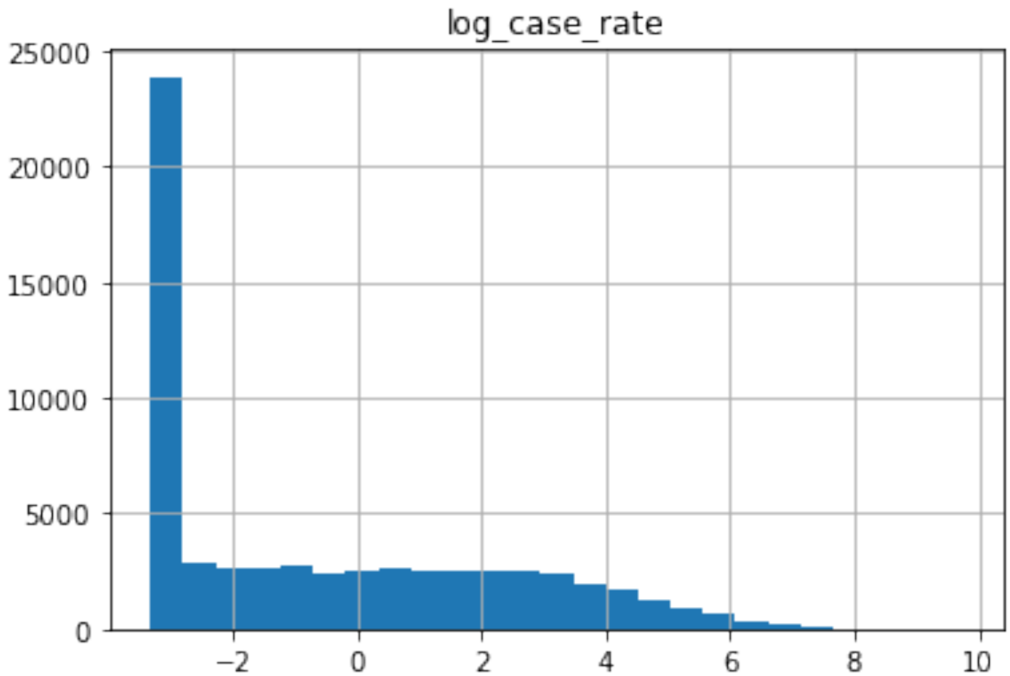
[13]:
```
covid_19_df_hist = clean_covid_19_df.hist('case_rate', bins = 100)
covid_19_df_hist = clean_covid_19_df.hist('death_rate', bins = 100)
```

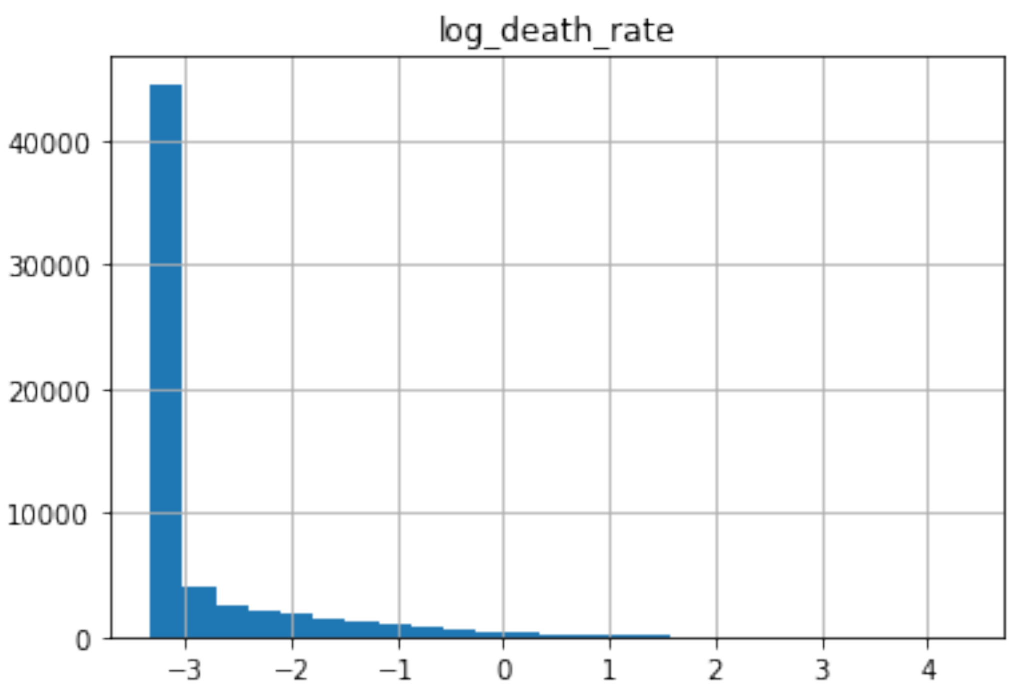



Generate histograms for the log of the case and death rates.

[14]:
```
covid_19_df_hist = clean_covid_19_df.hist('log_case_rate', bins = 25)
covid_19_df_hist = clean_covid_19_df.hist('log_death_rate', bins = 25)
```

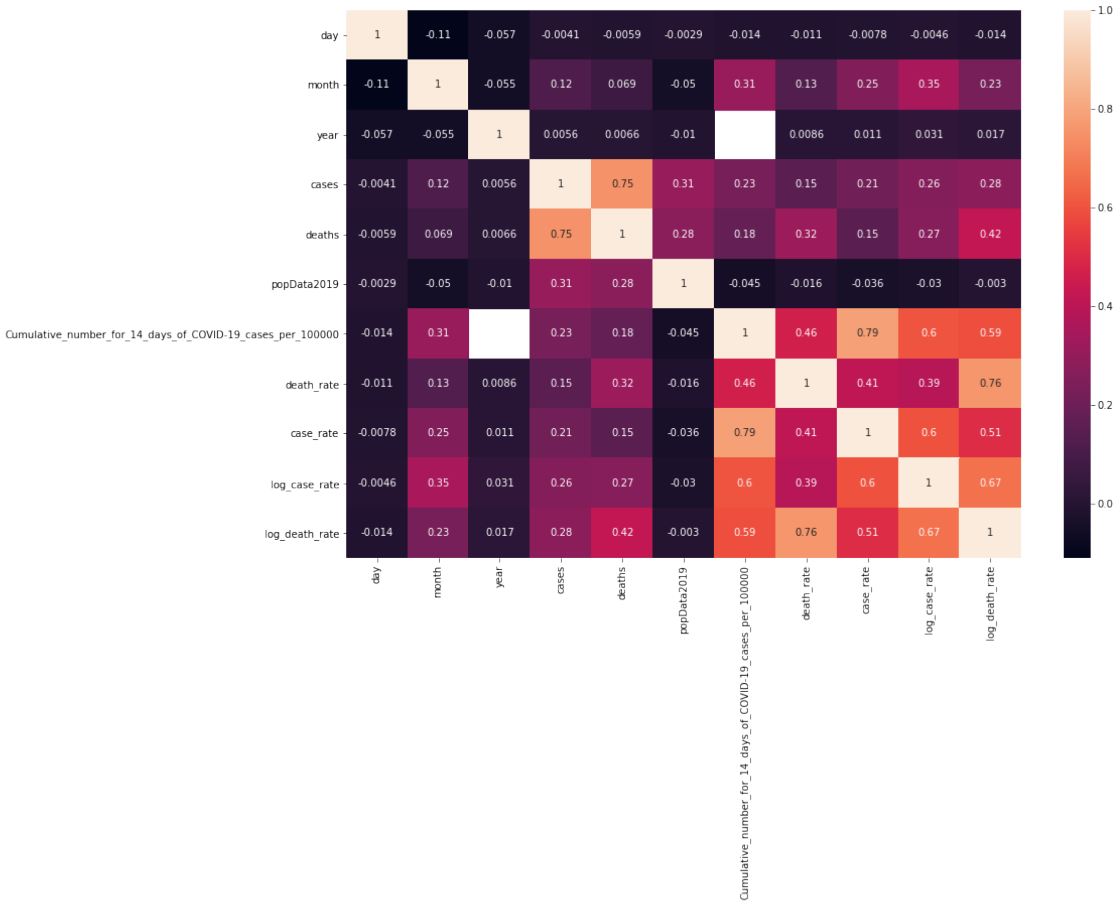


Generate a correlation matrix for the updated COVID-19 data frame.

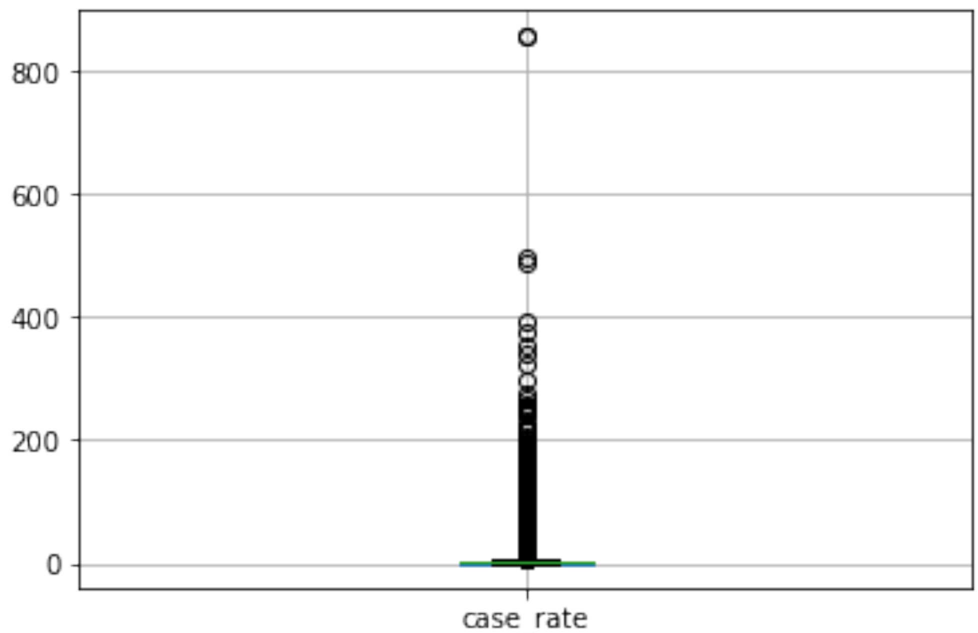
[15]:
```
plt.figure(figsize=(15, 10))

sns.heatmap(clean_covid_19_df.corr(), annot=True)
```

[15]: <AxesSubplot:>

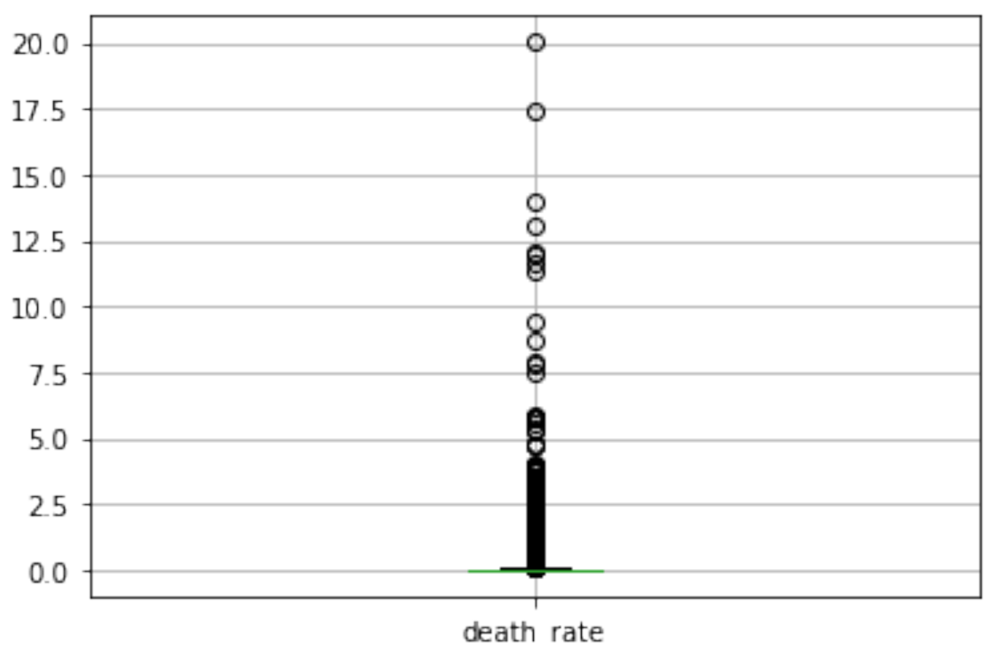


Generate the boxplots for the case rate and the death rate

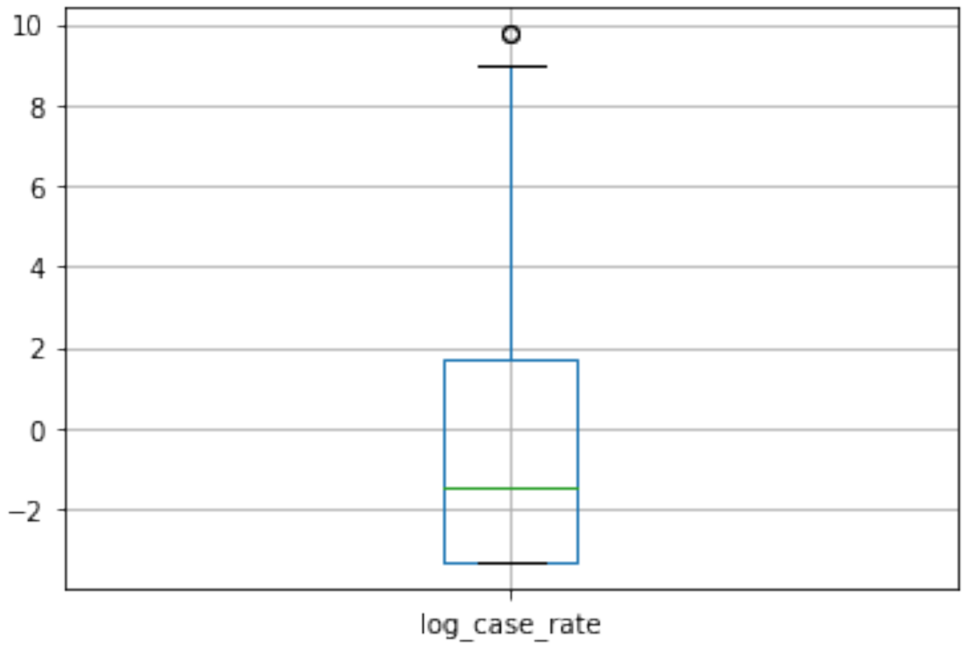
[16]:
```
covid_19_df_box_plot_case_rate = clean_covid_19_df.boxplot('case_rate')
```



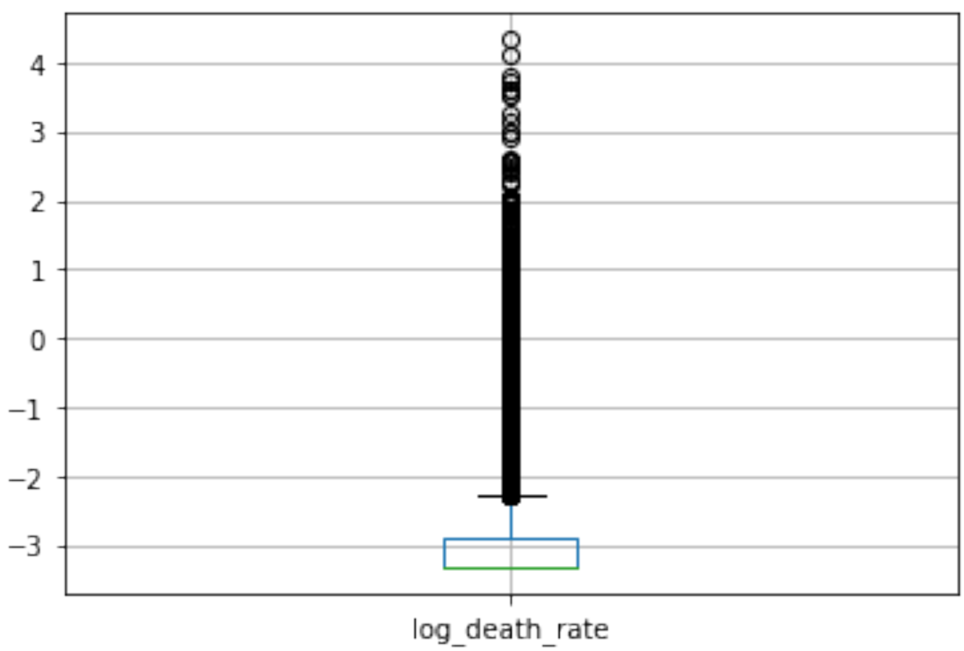

[17]:
```
covid_19_df_box_plot_death_rate = clean_covid_19_df.boxplot('death_rate')
```




[18]:
```
covid_19_df_box_plot_log_case_rate = clean_covid_19_df.boxplot('log_case_rate')
```




[19]:
```
covid_19_df_box_plot_log_death_rate = clean_covid_19_df.
 ↪boxplot('log_death_rate')
```



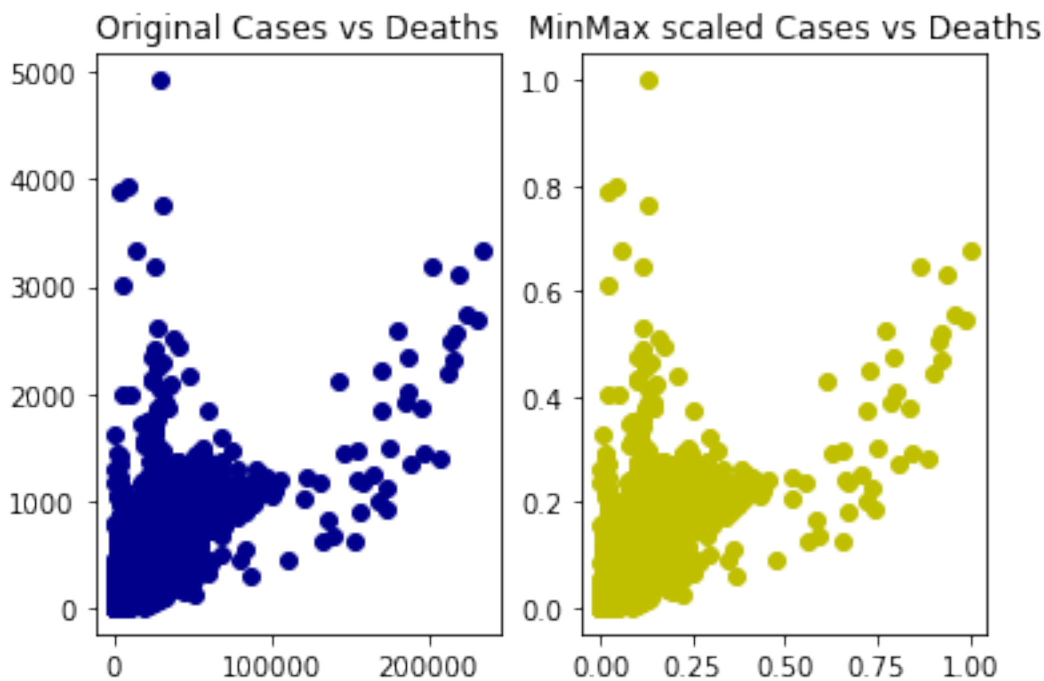
# 2 Min-max normalization and standard scaling for COVID-19 data

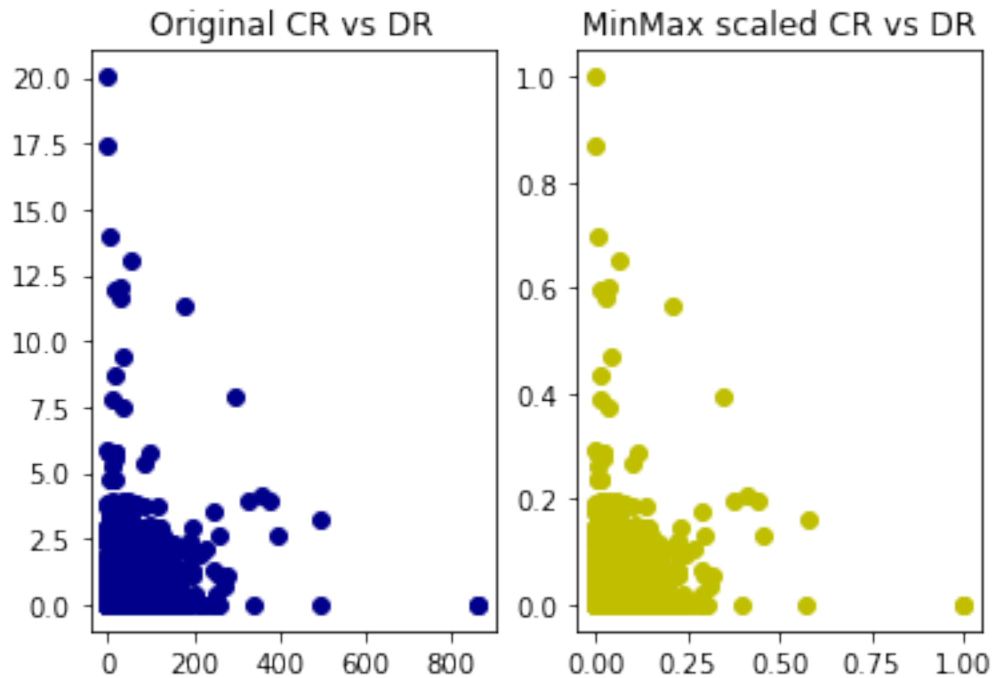
[20]:
```
# Import MinMaxScaler from sklearn
from sklearn.preprocessing import MinMaxScaler

# build scalar model
scaler = MinMaxScaler()

# Perform min-max scaling
clean_covid_19_df_min_max = scaler.fit_transform(clean_covid_19_df[['cases',␣
↪'deaths', 'death_rate', 'case_rate',
                                                                              ␣
↪'log_death_rate', 'log_case_rate']])

# Verify min and max value
print(clean_covid_19_df_min_max.min(axis = 0))
print(clean_covid_19_df_min_max.max(axis = 0))
```

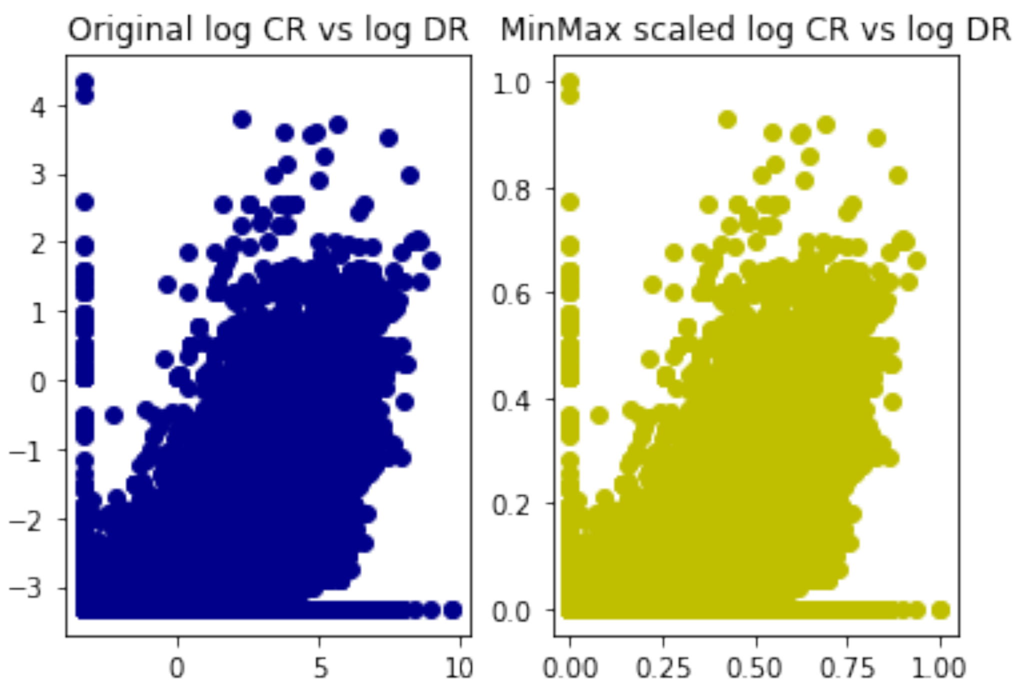
```
[0. 0. 0. 0. 0. 0.]
[1. 1. 1. 1. 1. 1.]
```

[21]:
```
# Effect of Min-Max normalization in a visual example

fig, axes = plt.subplots(1,2)

axes[0].scatter(clean_covid_19_df['cases'], clean_covid_19_df['deaths'],␣
↪c='darkblue')
axes[0].set_title("Original Cases vs Deaths")

axes[1].scatter(clean_covid_19_df_min_max[:,0], clean_covid_19_df_min_max[:,1],␣
↪c='y')
axes[1].set_title("MinMax scaled Cases vs Deaths")

plt.show()

fig, axes = plt.subplots(1,2)

axes[0].scatter(clean_covid_19_df['case_rate'],␣
↪clean_covid_19_df['death_rate'], c='darkblue')
axes[0].set_title("Original CR vs DR")

axes[1].scatter(clean_covid_19_df_min_max[:,3], clean_covid_19_df_min_max[:,2],␣
↪c='y')
axes[1].set_title("MinMax scaled CR vs DR")

plt.show()
```

```
fig, axes = plt.subplots(1,2)

axes[0].scatter(clean_covid_19_df['log_case_rate'],␣
 ↪clean_covid_19_df['log_death_rate'], c='darkblue')
axes[0].set_title("Original log CR vs log DR")

axes[1].scatter(clean_covid_19_df_min_max[:,5], clean_covid_19_df_min_max[:,4],␣
 ↪c='y')
axes[1].set_title("MinMax scaled log CR vs log DR")

plt.show()
```




Original CR vs DR
MinMax scaled CR vs DR
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
200
400
600
800
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.25
0.50
0.75
1.00

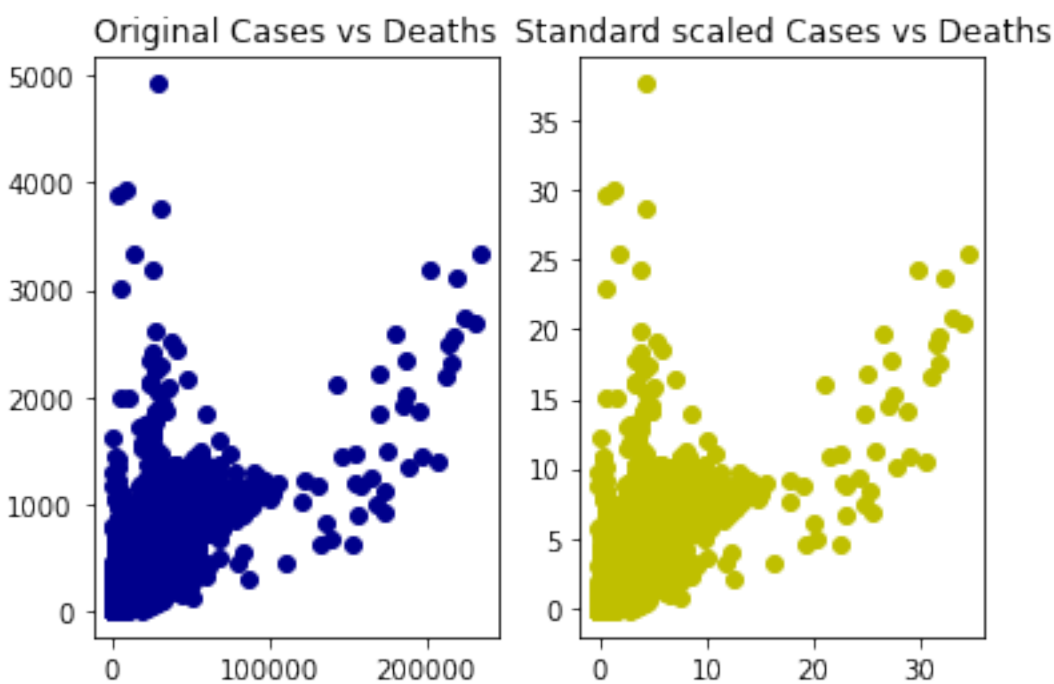

Original log CR vs log DR
MinMax scaled log CR vs log DR
4
3
2
1
0
−1
−2
−3
0
5
10
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.25
0.50
0.75
1.00

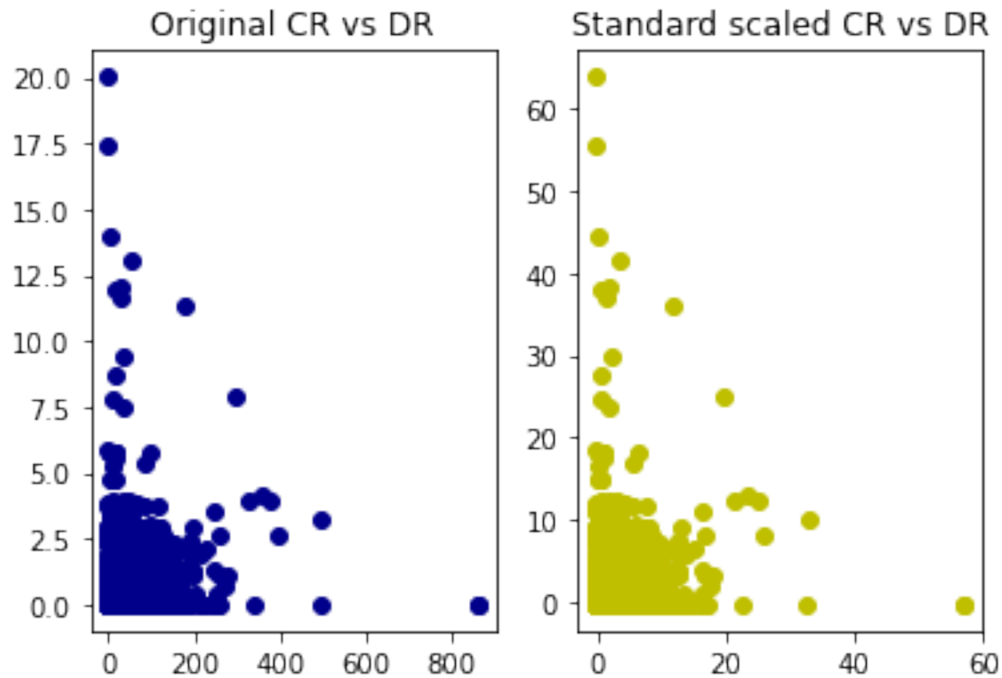
[22]:
```
# Import StandardScaler from sklearn
from sklearn.preprocessing import StandardScaler

# build scalar model
scaler = StandardScaler()

std_scaled = scaler.fit_transform(clean_covid_19_df[['cases', 'deaths',␣
↪'death_rate', 'case_rate',
                                                              'log_death_rate',␣
↪'log_case_rate']])

print(std_scaled)

print("\n")

# Verify min and max value
print(std_scaled.min(axis = 0))
print(std_scaled.max(axis = 0))
```

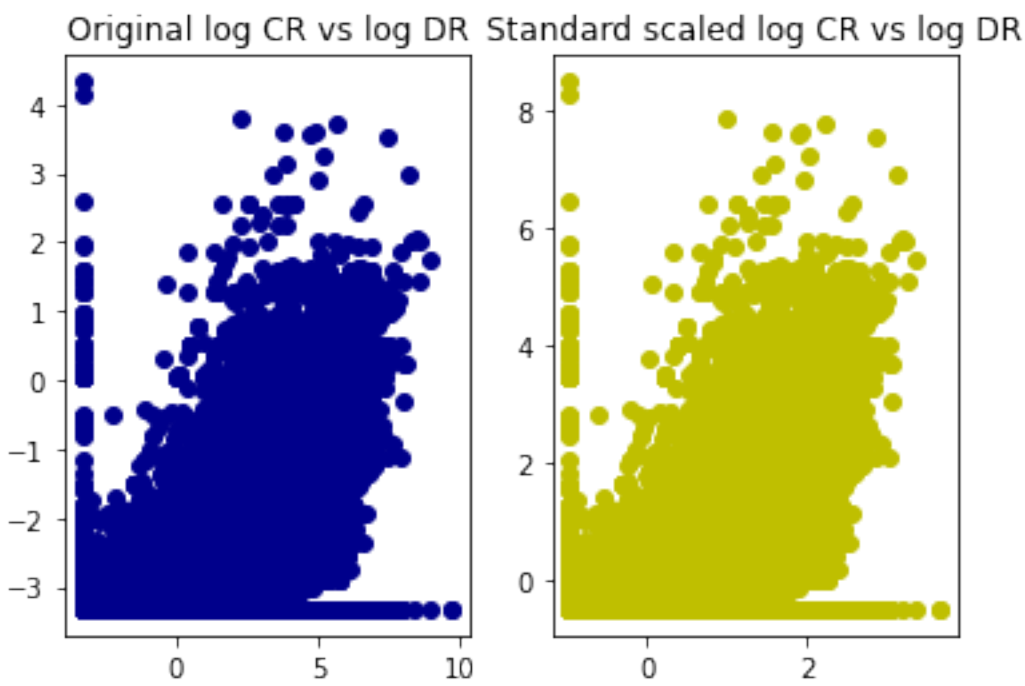
```
[[-0.06071609 -0.15420774 -0.21163944 -0.19331889 -0.25997379  0.57900605]
 [-0.12672599 -0.13117287 -0.18641775 -0.27218038 -0.14810481  0.14878049]
 [-0.15398454 -0.1158163  -0.16960329 -0.30474596 -0.07742182 -0.25730011]
 …
 [-0.17063435 -0.20027747 -0.26208282 -0.32463736 -0.50859702 -0.95749739]
 [-0.17048701 -0.20027747 -0.26208282 -0.32418012 -0.50859702 -0.92395674]
 [-0.17048701 -0.20027747 -0.26208282 -0.32418012 -0.50859702 -0.92395674]]


[-0.17063435 -0.20027747 -0.26208282 -0.32463736 -0.50859702 -0.95749739]
[34.40101682 37.63833061 63.8184198  57.19127212  8.49745551  3.6423725 ]
```

[23]:
```
# Effect of Standard Scaling in a visual example

fig, axes = plt.subplots(1,2)

axes[0].scatter(clean_covid_19_df['cases'], clean_covid_19_df['deaths'],␣
↪c='darkblue')
axes[0].set_title("Original Cases vs Deaths")

axes[1].scatter(std_scaled[:,0], std_scaled[:,1], c='y')
axes[1].set_title("Standard scaled Cases vs Deaths")

plt.show()

fig, axes = plt.subplots(1,2)
```

```
axes[0].scatter(clean_covid_19_df['case_rate'],␣
 ↪clean_covid_19_df['death_rate'], c='darkblue')
axes[0].set_title("Original CR vs DR")

axes[1].scatter(std_scaled[:,3], std_scaled[:,2], c='y')
axes[1].set_title("Standard scaled CR vs DR")

plt.show()

fig, axes = plt.subplots(1,2)

axes[0].scatter(clean_covid_19_df['log_case_rate'],␣
 ↪clean_covid_19_df['log_death_rate'], c='darkblue')
axes[0].set_title("Original log CR vs log DR")

axes[1].scatter(std_scaled[:,5], std_scaled[:,4], c='y')
axes[1].set_title("Standard scaled log CR vs log DR")

plt.show()
```




Original CR vs DR
Standard scaled CR vs DR
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
200
400
600
800
60
50
40
30
20
10
0
0
20
40
60


Original log CR vs log DR
Standard scaled log CR vs log DR
4
3
2
1
0
−1
−2
−3
0
5
10
8
6
4
2
0
0
2

# 3 Time Series for updated Covid-19 data frame

[24]:
```python
# Create datetime index for time series

clean_covid_19_df.index = pd.to_datetime(clean_covid_19_df['dateRep'], dayfirst␣
↪= True)
clean_covid_19_df
```

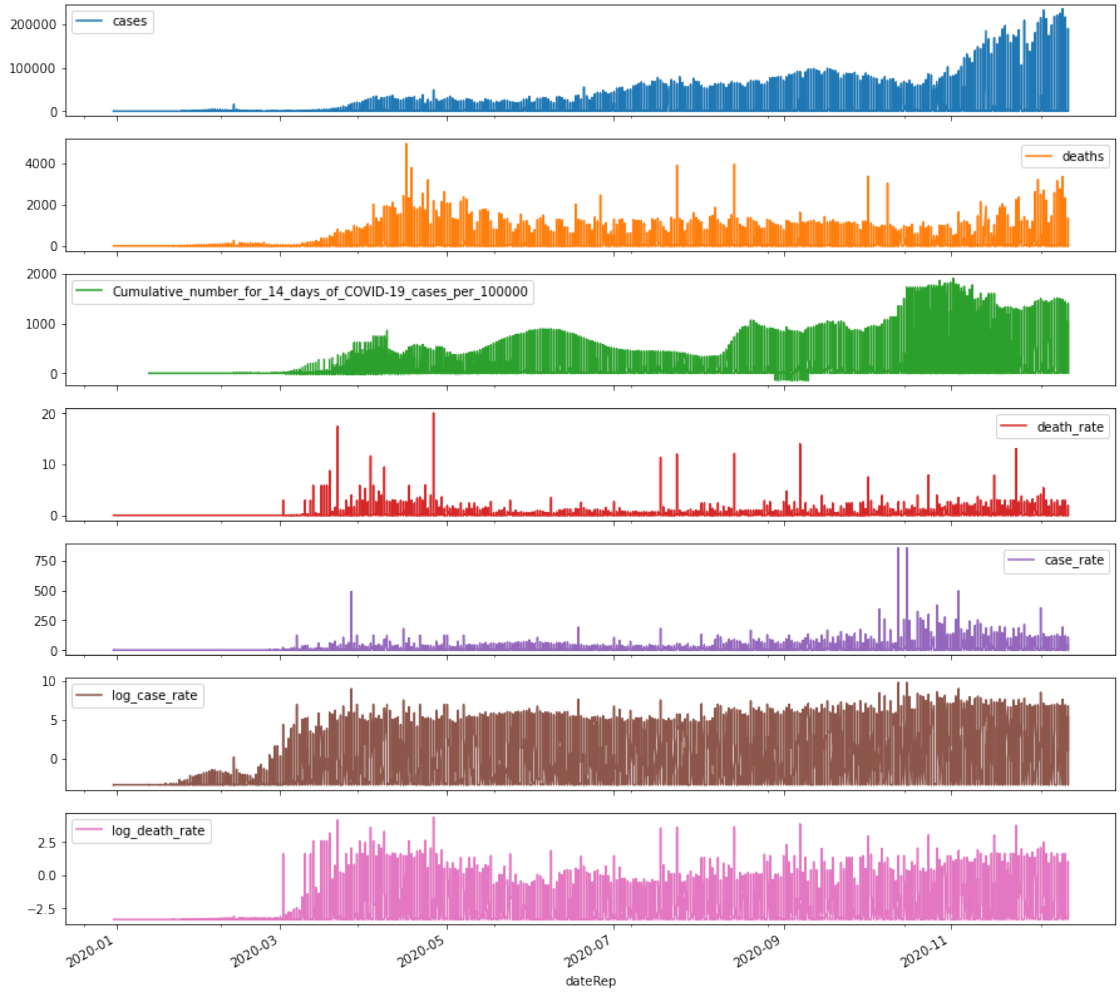
[24]:
```
                dateRep  day  month  year  cases  deaths  \
dateRep
2020-12-14  14/12/2020   14     12  2020    746       6
2020-12-13  13/12/2020   13     12  2020    298       9
2020-12-12    12/12/20   12     12  2020    113      11
2020-12-11    11/12/20   11     12  2020     63      10
2020-12-10    10/12/20   10     12  2020    202      16
…                 …  …    …   …     …     …
2019-12-31  31/12/2019   31     12  2019      0       0
2020-03-24  24/03/2020   24      3  2020      0       1
2019-12-31  31/12/2019   31     12  2019      0       0
2020-03-22  22/03/2020   22      3  2020      1       0
2020-03-21  21/03/2020   21      3  2020      1       0

              countriesAndTerritories geoId countryterritoryCode  popData2019  \
dateRep
2020-12-14                Afghanistan    AF                  AFG   38041757.0
2020-12-13                Afghanistan    AF                  AFG   38041757.0
2020-12-12                Afghanistan    AF                  AFG   38041757.0
2020-12-11                Afghanistan    AF                  AFG   38041757.0
2020-12-10                Afghanistan    AF                  AFG   38041757.0
…                               …   …                  …          …
2019-12-31  United_States_of_America    US                  USA  329064917.0
2020-03-24                   Zimbabwe    ZW                  ZWE   14645473.0
2019-12-31                    Vietnam    VN                  VNM   96462108.0
2020-03-22                   Zimbabwe    ZW                  ZWE   14645473.0
2020-03-21                   Zimbabwe    ZW                  ZWE   14645473.0

           continentExp  \
dateRep
2020-12-14         Asia
2020-12-13         Asia
2020-12-12         Asia
2020-12-11         Asia
2020-12-10         Asia
…                  …
2019-12-31      America
2020-03-24       Africa
2019-12-31         Asia
```

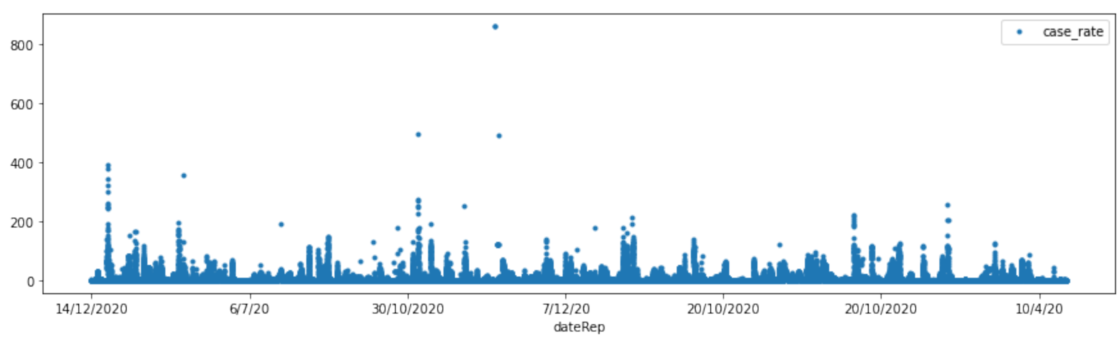
```
2020-03-22       Africa
2020-03-21       Africa

            Cumulative_number_for_14_days_of_COVID-19_cases_per_100000  \
dateRep
2020-12-14                                           9.013779
2020-12-13                                           7.052776
2020-12-12                                           6.868768
2020-12-11                                           7.134266
2020-12-10                                           6.968658
…                                                       …
2019-12-31                                                NaN
2020-03-24                                                NaN
2019-12-31                                                NaN
2020-03-22                                                NaN
2020-03-21                                                NaN

            death_rate  case_rate  log_case_rate  log_death_rate
dateRep
2020-12-14    0.015772   1.961003       1.043347       -3.110640
2020-12-13    0.023658   0.783350      -0.178943       -3.015570
2020-12-12    0.028916   0.297042      -1.332636       -2.955501
2020-12-11    0.026287   0.165607      -1.912632       -2.985223
2020-12-10    0.042059   0.530995      -0.664298       -2.815437
…                  …          …            …             …
2019-12-31    0.000000   0.000000      -3.321928       -3.321928
2020-03-24    0.006828   0.000000      -3.321928       -3.226638
2019-12-31    0.000000   0.000000      -3.321928       -3.321928
2020-03-22    0.000000   0.006828      -3.226638       -3.321928
2020-03-21    0.000000   0.006828      -3.226638       -3.321928

[61753 rows x 16 columns]
```

Generate time series for the cases, deaths, Cumulative_number_for_14_days_of_COVID-19_cases_per_100000, death_rate, case_rate, log_case_rate, and log_death_rate columns. In addtion, generate a time series modeling the case rate over time

[25]:
```python
time_series_covid_19_df = clean_covid_19_df.drop(columns = ["popData2019",␣
 ↪"day", "month", "year"])

time_series_covid_19_df.plot(subplots=True, figsize=(15,15))

time_series_covid_19_df.plot(x = 'dateRep', y = 'case_rate', style='.',␣
 ↪figsize=(15,4))
```

[25]:
```
<AxesSubplot:xlabel='dateRep'>
```




[26]:
```
time_series_covid_19_df
```

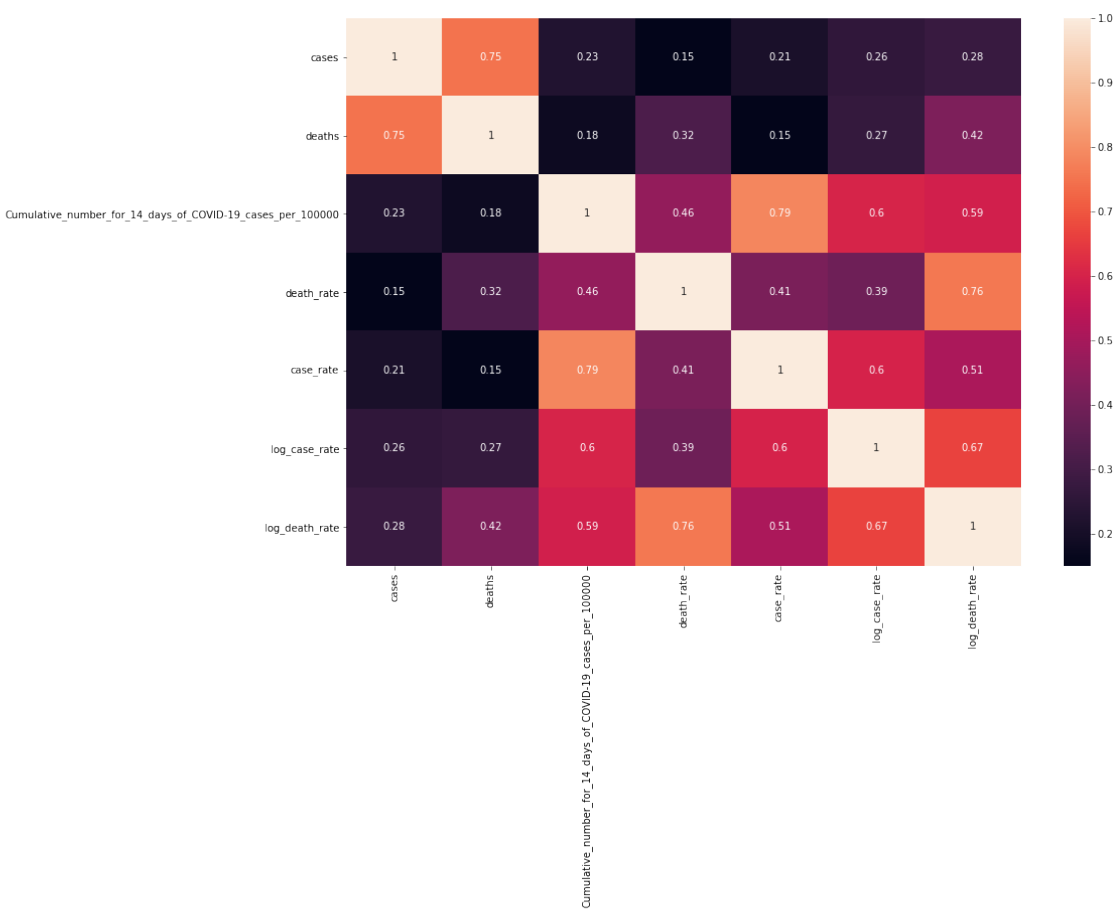
[26]:
```
                 dateRep  cases  deaths  countriesAndTerritories geoId  \
dateRep
2020-12-14  14/12/2020    746       6              Afghanistan    AF
2020-12-13  13/12/2020    298       9              Afghanistan    AF
2020-12-12    12/12/20    113      11              Afghanistan    AF
2020-12-11    11/12/20     63      10              Afghanistan    AF
2020-12-10    10/12/20    202      16              Afghanistan    AF
…                  …     …      …                          …     …
2019-12-31  31/12/2019      0       0 United_States_of_America    US
2020-03-24  24/03/2020      0       1                 Zimbabwe    ZW
2019-12-31  31/12/2019      0       0                  Vietnam    VN
2020-03-22  22/03/2020      1       0                 Zimbabwe    ZW
2020-03-21  21/03/2020      1       0                 Zimbabwe    ZW

           countryterritoryCode continentExp  \
dateRep
2020-12-14                  AFG         Asia
2020-12-13                  AFG         Asia
2020-12-12                  AFG         Asia
2020-12-11                  AFG         Asia
2020-12-10                  AFG         Asia
…                             …          …
2019-12-31                  USA      America
2020-03-24                  ZWE       Africa
2019-12-31                  VNM         Asia
2020-03-22                  ZWE       Africa
2020-03-21                  ZWE       Africa

            Cumulative_number_for_14_days_of_COVID-19_cases_per_100000  \
dateRep
2020-12-14                                           9.013779
2020-12-13                                           7.052776
2020-12-12                                           6.868768
2020-12-11                                           7.134266
2020-12-10                                           6.968658
…                                                          …
2019-12-31                                                NaN
2020-03-24                                                NaN
2019-12-31                                                NaN
2020-03-22                                                NaN
2020-03-21                                                NaN

            death_rate  case_rate  log_case_rate  log_death_rate
dateRep
2020-12-14    0.015772   1.961003       1.043347       -3.110640
2020-12-13    0.023658   0.783350      -0.178943       -3.015570
2020-12-12    0.028916   0.297042      -1.332636       -2.955501
```

```
2020-12-11     0.026287    0.165607      -1.912632        -2.985223
2020-12-10     0.042059    0.530995      -0.664298        -2.815437
…                  …          …          …                …
2019-12-31     0.000000    0.000000      -3.321928        -3.321928
2020-03-24     0.006828    0.000000      -3.321928        -3.226638
2019-12-31     0.000000    0.000000      -3.321928        -3.321928
2020-03-22     0.000000    0.006828      -3.226638        -3.321928
2020-03-21     0.000000    0.006828      -3.226638        -3.321928

[61753 rows x 12 columns]
```

Correlation Matrix for Time Series of COVID-19 data

[27]:
```python
plt.figure(figsize=(15, 10))

sns.heatmap(time_series_covid_19_df.corr(), annot=True)
```

[27]: <AxesSubplot:>



[28]:

```
# Get descriptive statistics of the time series
time_series_covid_19_df.describe()
```

[28]:

```
               cases        deaths  \
count   61753.000000  61753.000000
mean     1158.071689     26.083607
std      6786.916211    130.238403
min         0.000000      0.000000
25%         0.000000      0.000000
50%        16.000000      0.000000
75%       276.000000      4.000000
max    234633.000000   4928.000000

       Cumulative_number_for_14_days_of_COVID-19_cases_per_100000  \
count                                       58997.000000
mean                                           66.329444
std                                           162.354715
min                                          -147.419587
25%                                             0.757526
50%                                             6.724045
75%                                            52.559960
max                                          1900.836210

        death_rate     case_rate  log_case_rate  log_death_rate
count  61753.000000  61753.000000   61753.000000    61753.000000
mean       0.081945      4.847870      -0.601635       -2.889706
std        0.312673     14.933306       2.841068        0.849840
min        0.000000      0.000000      -3.321928       -3.321928
25%        0.000000      0.000000      -3.321928       -3.321928
50%        0.000000      0.260533      -1.471798       -3.321928
75%        0.034235      3.124746       1.689186       -2.897167
max       20.036065    858.895706       9.746507        4.331710
```

[ ]: