Deemal Patel
MSDS 422

Assignment 1

COVID has been an on-going problem since the beginning of 2020 and with the vaccine in sight it's slowly coming to an end. Thanks to the European Centre for Disease Prevention and Control (ECDC) , we have a dataset to analyze how COVID has impacted the world through the number of cases and number of deaths. With these numbers we can understand the actions of the world to combat the pandemic.

Once given the data the first action was to prepare the data for exploration. The dataset contains weekly updates of the number of COVID cases and deaths per country that dates all the way back to the first week of 2020. To fully prepare the data we needed to remove any null values from our set which can cause errors or inaccuracy in data calculations. The initial dataset contained 9790 rows and once the null rows were removed it dropped to 9513. This is probably because some of the data was either not entered properly or there wasn't enough information to fill out the column. One of the columns was a 'notification rate per 100,000' which had 235 missing values. This is probably because collecting the data for a rate of this was difficult to based on the data that was available for the country.

We begin exploring the data to see what sort of information our dataset contains such as data types and basic statistical information within our columns. Some of the statistical information that is provided is the mean, standard deviation, and quarterly percentiles. The average weekly deaths came out to about 204 deaths. With the rapid spread of the disease and the fact that older people were more susceptible to getting it, this is probably why the death average is high.

We wanted to see if there are any correlations between the columns so we developed a heatmap chart that provides us information on whether any of the columns had a negative or positive correlation. Our heatmap showed that weekly deaths and weekly cases had a correlation of 0.83. This is an extremely positive correlation meaning these two definitely went hand in hand. If there was a high amount of cases then that means there was going to be a high amount of deaths, relative to the cases. There is also a small correlation of 0.28 between the population and cases/deaths per week. This makes sense because places with a denser population means there are more people to spread the disease to. In the charts developed, we can see that in USA cases and deaths had a slow start and began spiking around week 12. Week 12 would be the beginning of March and that's when US began a nation wide lockdown to slow the spread and from the charts you can definitely see a slight decline for the next 12 weeks but rose again after. In June, the country started opening places back up causing the rise of cases.

Lastly for any machine learning model a best way to use our variables is to scale it. Two common scalers that we used are the MinMax and Standard Scaler. We used the weekly deaths variable when comparing the two using a distribution plot we can see that they match up to the unscaled graph. MinMax scales from 0 to 1 standard has its own scale depending on the data. We want to use scales because in the case of weekly deaths the numbers had a wide range from -831, which must be a user error, to 22852. Normally when using the scalers it would be a 2D array but our case we used a 1D array to just get a visualization of how the scaler would work.

APPENDIX:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import StandardScaler
```

```
covidData = pd.read_csv('https://opendata.ecdc.europa.eu/covid19/casedistribution/csv')
```

Viewing the Dataset

```
pd.concat([covidData.head(10),covidData.tail(10)])
```

	dateRep	year_week	cases_weekly	deaths_weekly	countriesAndTerritories	geoId	countryterritoryCode	popData2019	continentExp	notification_rate_pe
0	11/01/2021	2021-01	675	71	Afghanistan	AF	AFG	38041757.0	Asia	
1	04/01/2021	2020-53	902	60	Afghanistan	AF	AFG	38041757.0	Asia	
2	28/12/2020	2020-52	1994	88	Afghanistan	AF	AFG	38041757.0	Asia	
3	21/12/2020	2020-51	740	111	Afghanistan	AF	AFG	38041757.0	Asia	
4	14/12/2020	2020-50	1757	71	Afghanistan	AF	AFG	38041757.0	Asia	
5	07/12/2020	2020-49	1672	137	Afghanistan	AF	AFG	38041757.0	Asia	
6	30/11/2020	2020-48	1073	68	Afghanistan	AF	AFG	38041757.0	Asia	
7	23/11/2020	2020-47	1368	69	Afghanistan	AF	AFG	38041757.0	Asia	
8	16/11/2020	2020-46	1164	61	Afghanistan	AF	AFG	38041757.0	Asia	
9	09/11/2020	2020-45	606	24	Afghanistan	AF	AFG	38041757.0	Asia	
9781	25/05/2020	2020-21	10	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9782	18/05/2020	2020-20	10	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9783	11/05/2020	2020-19	2	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9784	04/05/2020	2020-18	3	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9785	27/04/2020	2020-17	6	1	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9786	20/04/2020	2020-16	11	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9787	13/04/2020	2020-15	5	2	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9788	06/04/2020	2020-14	2	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9789	30/03/2020	2020-13	5	1	Zimbabwe	ZW	ZWE	14645473.0	Africa	
9790	23/03/2020	2020-12	2	0	Zimbabwe	ZW	ZWE	14645473.0	Africa	

```
covidData.shape
```

```
(9791, 10)
```

Preparing the Data

```
In [5]: covidData.isnull().sum()
```

```
Out[5]: dateRep                                      0
        year_week                                    0
        cases_weekly                                 0
        deaths_weekly                                0
        countriesAndTerritories                      0
        geoId                                       44
        countryterritoryCode                        22
        popData2019                                 22
        continentExp                                 0
        notification_rate_per_100000_population_14-days    235
        dtype: int64
```

```
In [6]: covidData.dropna(inplace=True)
```

```
In [7]: covidData.shape
```

```
Out[7]: (9513, 10)
```

Exploring the Data

```
covidData.info()
```

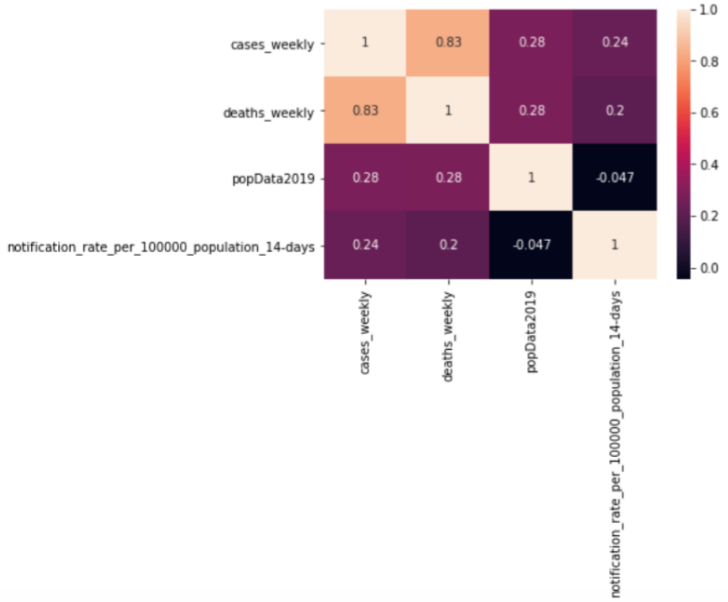
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9513 entries, 0 to 9789
Data columns (total 10 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   dateRep                                     9513 non-null   object 
 1   year_week                                   9513 non-null   object 
 2   cases_weekly                                9513 non-null   int64  
 3   deaths_weekly                               9513 non-null   int64  
 4   countriesAndTerritories                     9513 non-null   object 
 5   geoId                                       9513 non-null   object 
 6   countryterritoryCode                        9513 non-null   object 
 7   popData2019                                 9513 non-null   float64
 8   continentExp                                9513 non-null   object 
 9   notification_rate_per_100000_population_14-days  9513 non-null   float64
dtypes: float64(2), int64(2), object(6)
memory usage: 817.5+ KB
```

```
covidData.describe()
```

	cases_weekly	deaths_weekly	popData2019	notification_rate_per_100000_population_14-days
count	9.513000e+03	9513.000000	9.513000e+03	9513.000000
mean	9.436698e+03	203.956796	4.070112e+07	77.969015
std	5.640675e+04	939.511802	1.520623e+08	189.652189
min	-3.864000e+03	-875.000000	8.150000e+02	-132.600000
25%	1.000000e+01	0.000000	1.293120e+06	0.810000
50%	1.900000e+02	2.000000	7.813207e+06	7.660000
75%	2.523000e+03	40.000000	2.851583e+07	61.950000
max	1.782792e+06	22852.000000	1.433784e+09	4343.440000

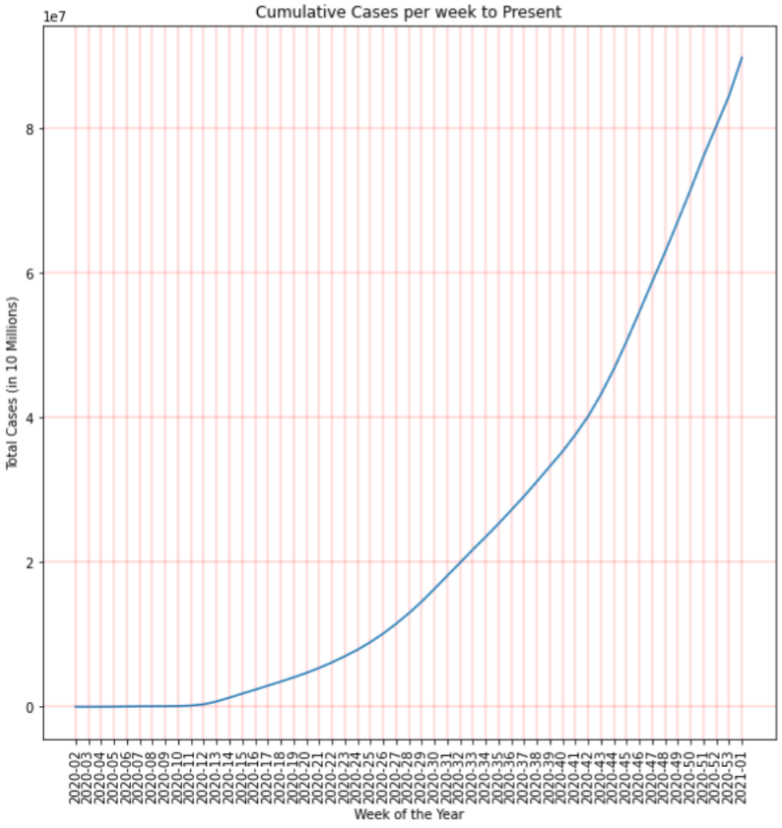
```
sns.heatmap(covidData.corr(),annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fcc3bed2790>
```

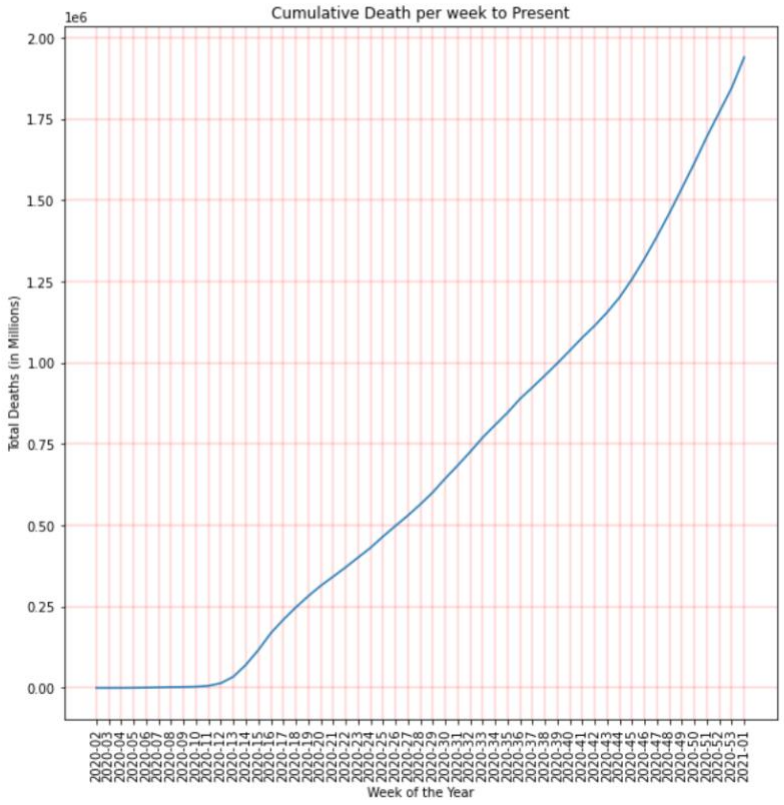


Tracking the Disease

```
covidDatarev = covidData.iloc[::-1]
df_agg = covidDatarev.groupby(['year_week']).agg({'cases_weekly':sum})
fig, ax = plt.subplots()
plt.xticks(rotation=90)
ax.plot(df_agg.index, df_agg['cases_weekly'].cumsum())
plt.xlabel('Week of the Year')
plt.ylabel('Total Cases (in 10 Millions)')
plt.title('Cumulative Cases per week to Present')
plt.grid(color='r', linestyle='-', linewidth=0.25)
fig.set_size_inches([10,10])
```

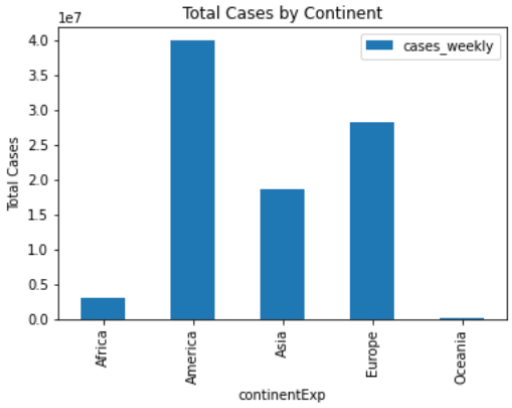


```
covidDatarev = covidData.iloc[::-1]
df_agg = covidDatarev.groupby(['year_week']).agg({'deaths_weekly':sum})
fig, ax = plt.subplots()
plt.xticks(rotation=90)
ax.plot(df_agg.index, df_agg['deaths_weekly'].cumsum())
plt.xlabel('Week of the Year')
plt.ylabel('Total Deaths (in Millions)')
plt.title('Cumulative Death per week to Present')
plt.grid(color='r', linestyle='-', linewidth=0.25)
fig.set_size_inches([10,10])
```




```
continentdf= covidData.groupby(['continentExp']).agg({'cases_weekly':sum})
continentdf.plot(kind='bar')
plt.ylabel('Total Cases')
plt.title('Total Cases by Continent')
```

```
Text(0.5, 1.0, 'Total Cases by Continent')
```

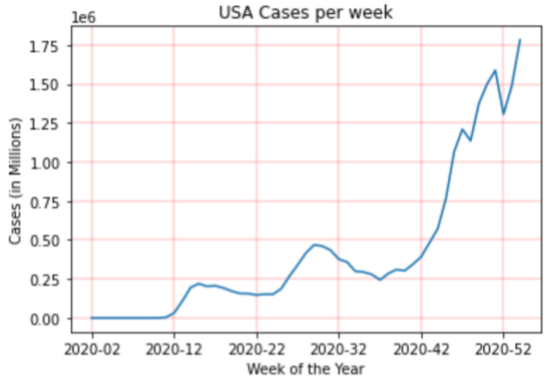


```
covidData['infectionRatePerWeekPerMillion'] = (covidData['cases_weekly'] / covidData['popData2019']) * 1000000
```

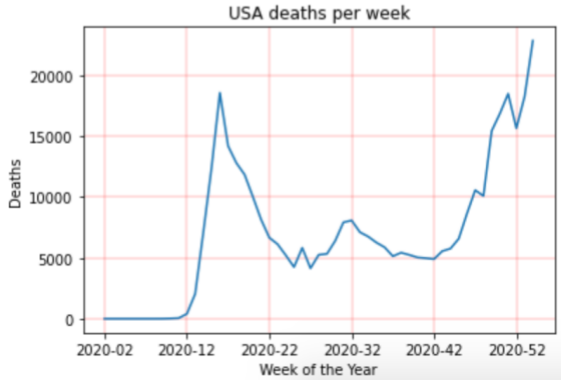
```
covidData
```

eekly	countriesAndTerritories	geoId	countryterritoryCode	popData2019	continentExp	notification_rate_per_100000_population_14-days	infectionRatePerWeekPerMillion
71	Afghanistan	AF	AFG	38041757.0	Asia	4.15	17.743660
60	Afghanistan	AF	AFG	38041757.0	Asia	7.61	23.710787
88	Afghanistan	AF	AFG	38041757.0	Asia	7.19	52.416086
111	Afghanistan	AF	AFG	38041757.0	Asia	6.56	19.452309
71	Afghanistan	AF	AFG	38041757.0	Asia	9.01	46.186090
...	...	...	...	...	...	...	...
1	Zimbabwe	ZW	ZWE	14645473.0	Africa	0.12	0.409683
0	Zimbabwe	ZW	ZWE	14645473.0	Africa	0.11	0.751085
2	Zimbabwe	ZW	ZWE	14645473.0	Africa	0.05	0.341402
0	Zimbabwe	ZW	ZWE	14645473.0	Africa	0.05	0.136561
1	Zimbabwe	ZW	ZWE	14645473.0	Africa	0.05	0.341402

```
usaDF = covidData[covidData['countryterritoryCode']=='USA'].set_index('year_week')
usaDF=usaDF.iloc[::-1]
usaDF['cases_weekly'].plot()
plt.xlabel('Week of the Year')
plt.ylabel('Cases (in Millions)')
plt.title('USA Cases per week')
plt.grid(color='r', linestyle='-', linewidth=0.25)
fig.set_size_inches([10,10])
```




```
usaDF['deaths_weekly'].plot()
plt.xlabel('Week of the Year')
plt.ylabel('Deaths')
plt.title('USA deaths per week')
plt.grid(color='r', linestyle='-', linewidth=0.25)
fig.set_size_inches([10,10])
```



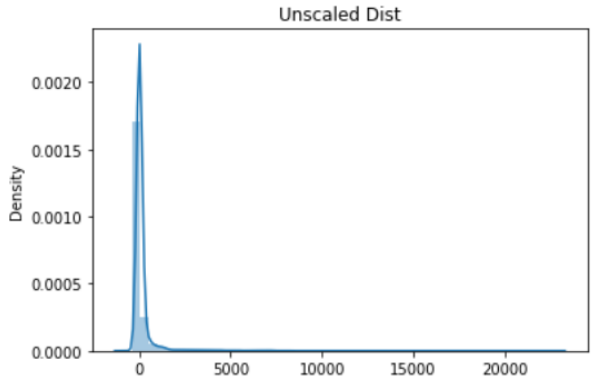
Scaling the 'deaths_weekly' variable

```
X = pd.DataFrame(covidData['deaths_weekly'])
```

```
sns.distplot(X).set_title('Unscaled Dist')
print(X.values.reshape(-1,1))
```

```
/Users/deemalpatel/opt/anaconda3/lib/python3.7/site-packages/seaborn/
is a deprecated function and will be removed in a future version. Ple
igure-level function with similar flexibility) or `histplot` (an axes
  warnings.warn(msg, FutureWarning)
```

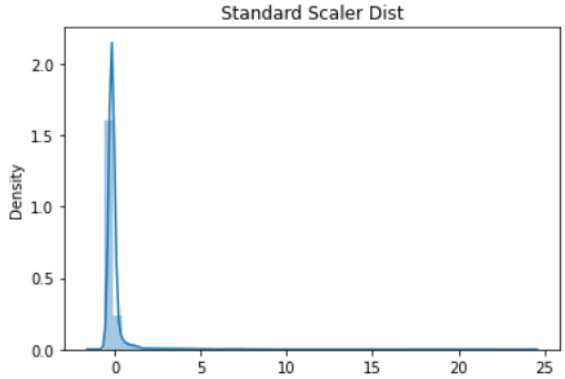
```
[[71]
 [60]
 [88]
 ...
 [ 2]
 [ 0]
 [ 1]]
```




```
sns.distplot(StandardScaler().fit_transform(X.values.reshape(-1,1))).
print(StandardScaler().fit_transform(X.values.reshape(-1,1)))
```

```
/Users/deemalpatel/opt/anaconda3/lib/python3.7/site-packages/seaborn/
is a deprecated function and will be removed in a future version. Ple
igure-level function with similar flexibility) or `histplot` (an axes
  warnings.warn(msg, FutureWarning)
```

```
[[-0.14152434]
 [-0.15323316]
 [-0.12342888]
 ...
 [-0.2149706 ]
 [-0.21709947]
 [-0.21603503]]
```



```
sns.distplot(MinMaxScaler().fit_transform(X.values.reshape(-1,1)))
print(MinMaxScaler().fit_transform(X.values.reshape(-1,1)))
```

```
/Users/deemalpatel/opt/anaconda3/lib/python3.7/site-packages/seabo
is a deprecated function and will be removed in a future version.
igure-level function with similar flexibility) or `histplot` (an a
  warnings.warn(msg, FutureWarning)
```

```
[[0.03987019]
 [0.03940658]
 [0.04058667]
 ...
 [0.03696211]
 [0.03687782]
 [0.03691996]]
```