

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from datetime import datetime
import plotly.graph_objects as go
import plotly.io as pio
```

```
In [2]: Data = pd.read_csv('owid-covid-data.csv',index_col=False)
```

```
In [3]: Data
```

```
Out[3]:
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0		NaN
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0		NaN
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0		NaN
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0		NaN
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0		NaN
...
139377	ZWE	Africa	Zimbabwe	2021-12-07	141601.0	2555.0		996.571
139378	ZWE	Africa	Zimbabwe	2021-12-08	150628.0	9027.0		2184.429
139379	ZWE	Africa	Zimbabwe	2021-12-09	155817.0	5189.0		2776.857
139380	ZWE	Africa	Zimbabwe	2021-12-10	155817.0	0.0		2625.143
139381	ZWE	Africa	Zimbabwe	2021-12-11	165002.0	9185.0		3782.714

139382 rows × 67 columns

```
In [4]: DataGermany=Data[Data['location']=='Germany']
```

```
In [5]: DataGermany = DataGermany.reset_index(drop=True)
```

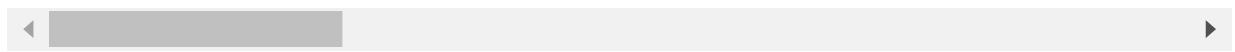
```
In [6]: DataGermany
```

```
Out[6]:
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_death
0	DEU	Europe	Germany	2020-01-27	1.0	1.0		NaN
1	DEU	Europe	Germany	2020-01-28	4.0	3.0		NaN

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_death
2	DEU	Europe	Germany	2020-01-29	4.0	0.0	NaN	NaN
3	DEU	Europe	Germany	2020-01-30	4.0	0.0	NaN	NaN
4	DEU	Europe	Germany	2020-01-31	5.0	1.0	NaN	NaN
...
680	DEU	Europe	Germany	2021-12-07	6312346.0	93087.0	55540.286	104051.0
681	DEU	Europe	Germany	2021-12-08	6339828.0	27482.0	48686.857	104201.0
682	DEU	Europe	Germany	2021-12-09	6442846.0	103018.0	52935.286	105000.0
683	DEU	Europe	Germany	2021-12-10	6496142.0	53296.0	51664.286	105510.0
684	DEU	Europe	Germany	2021-12-11	6528894.0	32752.0	50128.857	105642.0

685 rows × 67 columns



In [7]: DataGermany.columns

```
Out[7]: Index(['iso_code', 'continent', 'location', 'date', 'total_cases', 'new_cases',
       'new_cases_smoothed', 'total_deaths', 'new_deaths',
       'new_deaths_smoothed', 'total_cases_per_million',
       'new_cases_per_million', 'new_cases_smoothed_per_million',
       'total_deaths_per_million', 'new_deaths_per_million',
       'new_deaths_smoothed_per_million', 'reproduction_rate', 'icu_patients',
       'icu_patients_per_million', 'hosp_patients',
       'hosp_patients_per_million', 'weekly_icu_admissions',
       'weekly_icu_admissions_per_million', 'weekly_hosp_admissions',
       'weekly_hosp_admissions_per_million', 'new_tests', 'total_tests',
       'total_tests_per_thousand', 'new_tests_per_thousand',
       'new_tests_smoothed', 'new_tests_smoothed_per_thousand',
       'positive_rate', 'tests_per_case', 'tests_units', 'total_vaccinations',
       'people_vaccinated', 'people_fully_vaccinated', 'total_boosters',
       'new_vaccinations', 'new_vaccinations_smoothed',
       'total_vaccinations_per_hundred', 'people_vaccinated_per_hundred',
       'people_fully_vaccinated_per_hundred', 'total_boosters_per_hundred',
       'new_vaccinations_smoothed_per_million',
       'new_people_vaccinated_smoothed',
       'new_people_vaccinated_smoothed_per_hundred', 'stringency_index',
       'population', 'population_density', 'median_age', 'aged_65_older',
       'aged_70_older', 'gdp_per_capita', 'extreme_poverty',
       'cardiovasc_death_rate', 'diabetes_prevalence', 'female_smokers',
       'male_smokers', 'handwashing_facilities', 'hospital_beds_per_thousand',
       'life_expectancy', 'human_development_index',
       'excess_mortality_cumulative_absolute', 'excess_mortality_cumulative',
       'excess_mortality', 'excess_mortality_cumulative_per_million'],
      dtype='object')
```

Checking null and nan

In [8]: TempNull=DataGermany.isnull().sum().values
DataGermany.isnull().sum().values

```
Out[8]: array([ 0, 0, 0, 0, 0, 0, 5, 42, 42, 5, 0, 0, 5,
    42, 42, 5, 38, 53, 53, 685, 685, 667, 667, 592, 592, 685,
    594, 594, 685, 61, 61, 594, 594, 54, 336, 336, 336, 338, 337,
    337, 336, 336, 336, 338, 337, 337, 337, 6, 0, 0, 0, 0,
    0, 0, 685, 0, 0, 0, 685, 0, 0, 0, 590, 590,
    590, 590], dtype=int64)
```

```
In [9]: TempNa = DataGermany.isna().sum().values
DataGermany.isna().sum().values
```

```
Out[9]: array([ 0, 0, 0, 0, 0, 0, 5, 42, 42, 5, 0, 0, 5,
    42, 42, 5, 38, 53, 53, 685, 685, 667, 667, 592, 592, 685,
    594, 594, 685, 61, 61, 594, 594, 54, 336, 336, 336, 338, 337,
    337, 336, 336, 336, 338, 337, 337, 337, 6, 0, 0, 0, 0,
    0, 0, 685, 0, 0, 0, 685, 0, 0, 0, 590, 590,
    590, 590], dtype=int64)
```

```
In [10]: SelectedData = DataGermany.loc[:,TempNa<61]
```

```
In [11]: SelectedData
```

```
Out[11]:
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_death
0	DEU	Europe	Germany	2020-01-27	1.0	1.0	NaN	NaN
1	DEU	Europe	Germany	2020-01-28	4.0	3.0	NaN	NaN
2	DEU	Europe	Germany	2020-01-29	4.0	0.0	NaN	NaN
3	DEU	Europe	Germany	2020-01-30	4.0	0.0	NaN	NaN
4	DEU	Europe	Germany	2020-01-31	5.0	1.0	NaN	NaN
...
680	DEU	Europe	Germany	2021-12-07	6312346.0	93087.0	55540.286	104051.0
681	DEU	Europe	Germany	2021-12-08	6339828.0	27482.0	48686.857	104201.0
682	DEU	Europe	Germany	2021-12-09	6442846.0	103018.0	52935.286	105000.0
683	DEU	Europe	Germany	2021-12-10	6496142.0	53296.0	51664.286	105510.0
684	DEU	Europe	Germany	2021-12-11	6528894.0	32752.0	50128.857	105642.0

685 rows × 34 columns



```
In [12]: SelectedData.columns
```

```
Out[12]: Index(['iso_code', 'continent', 'location', 'date', 'total_cases', 'new_cases',
    'new_cases_smoothed', 'total_deaths', 'new_deaths',
    'new_deaths_smoothed', 'total_cases_per_million',
    'new_cases_per_million', 'new_cases_smoothed_per_million',
    'total_deaths_per_million', 'new_deaths_per_million',
    'new_deaths_smoothed_per_million', 'reproduction_rate', 'icu_patients',
```

```
'icu_patients_per_million', 'tests_units', 'stringency_index',
'population', 'population_density', 'median_age', 'aged_65_older',
'aged_70_older', 'gdp_per_capita', 'cardiovasc_death_rate',
'diabetes_prevalence', 'female_smokers', 'male_smokers',
'hospital_beds_per_thousand', 'life_expectancy',
'human_development_index'],
dtype='object')
```

```
In [13]: x=SelectedData.loc[:,['date', 'total_cases', 'new_cases',
                           'new_cases_smoothed', 'total_deaths', 'new_deaths',
                           'new_deaths_smoothed','reproduction_rate', 'icu_patients','stringency_index']]
```

```
In [14]: x.isna().sum()
```

```
Out[14]: date          0
total_cases      0
new_cases        0
new_cases_smoothed 5
total_deaths     42
new_deaths       42
new_deaths_smoothed 5
reproduction_rate 38
icu_patients     53
stringency_index  6
dtype: int64
```

```
In [15]: x[x.new_cases_smoothed!=x.new_cases_smoothed]
```

```
Out[15]:   date  total_cases  new_cases  new_cases_smoothed  total_deaths  new_deaths  new_deaths_smoothed
0  2020-01-27      1.0      1.0            NaN            NaN            NaN            NaN
1  2020-01-28      4.0      3.0            NaN            NaN            NaN            NaN
2  2020-01-29      4.0      0.0            NaN            NaN            NaN            NaN
3  2020-01-30      4.0      0.0            NaN            NaN            NaN            NaN
4  2020-01-31      5.0      1.0            NaN            NaN            NaN            NaN
```

The new cases smoothed are NaN they need data to get the smoothed value. Will replace it with zero value.

```
In [16]: x.loc[0:4,['new_cases_smoothed']] = 0
```

```
In [17]: x
```

```
Out[17]:   date  total_cases  new_cases  new_cases_smoothed  total_deaths  new_deaths  new_deaths_smoothed
0  2020-01-27      1.0      1.0         0.000            NaN            NaN            NaN
1  2020-01-28      4.0      3.0         0.000            NaN            NaN            NaN
2  2020-01-29      4.0      0.0         0.000            NaN            NaN            NaN
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
3	2020-01-30	4.0	0.0	0.000	NaN	NaN	
4	2020-01-31	5.0	1.0	0.000	NaN	NaN	
...
680	2021-12-07	6312346.0	93087.0	55540.286	104051.0	811.0	3
681	2021-12-08	6339828.0	27482.0	48686.857	104201.0	150.0	2
682	2021-12-09	6442846.0	103018.0	52935.286	105000.0	799.0	3
683	2021-12-10	6496142.0	53296.0	51664.286	105510.0	510.0	3
684	2021-12-11	6528894.0	32752.0	50128.857	105642.0	132.0	3

685 rows × 10 columns



In [18]: `x[x.total_deaths != x.total_deaths]`

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	NaN	NaN	
1	2020-01-28	4.0	3.0	0.000	NaN	NaN	
2	2020-01-29	4.0	0.0	0.000	NaN	NaN	
3	2020-01-30	4.0	0.0	0.000	NaN	NaN	
4	2020-01-31	5.0	1.0	0.000	NaN	NaN	
5	2020-02-01	8.0	3.0	1.143	NaN	NaN	
6	2020-02-02	10.0	2.0	1.429	NaN	NaN	
7	2020-02-03	12.0	2.0	1.571	NaN	NaN	
8	2020-02-04	12.0	0.0	1.143	NaN	NaN	
9	2020-02-05	12.0	0.0	1.143	NaN	NaN	
10	2020-02-06	12.0	0.0	1.143	NaN	NaN	
11	2020-02-07	13.0	1.0	1.143	NaN	NaN	

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoo
12	2020-02-08	13.0	0.0	0.714	NaN	NaN	
13	2020-02-09	14.0	1.0	0.571	NaN	NaN	
14	2020-02-10	14.0	0.0	0.286	NaN	NaN	
15	2020-02-11	16.0	2.0	0.571	NaN	NaN	
16	2020-02-12	16.0	0.0	0.571	NaN	NaN	
17	2020-02-13	16.0	0.0	0.571	NaN	NaN	
18	2020-02-14	16.0	0.0	0.429	NaN	NaN	
19	2020-02-15	16.0	0.0	0.429	NaN	NaN	
20	2020-02-16	16.0	0.0	0.286	NaN	NaN	
21	2020-02-17	16.0	0.0	0.286	NaN	NaN	
22	2020-02-18	16.0	0.0	0.000	NaN	NaN	
23	2020-02-19	16.0	0.0	0.000	NaN	NaN	
24	2020-02-20	16.0	0.0	0.000	NaN	NaN	
25	2020-02-21	16.0	0.0	0.000	NaN	NaN	
26	2020-02-22	16.0	0.0	0.000	NaN	NaN	
27	2020-02-23	16.0	0.0	0.000	NaN	NaN	
28	2020-02-24	16.0	0.0	0.000	NaN	NaN	
29	2020-02-25	17.0	1.0	0.143	NaN	NaN	
30	2020-02-26	27.0	10.0	1.571	NaN	NaN	
31	2020-02-27	46.0	19.0	4.286	NaN	NaN	
32	2020-02-28	48.0	2.0	4.571	NaN	NaN	
33	2020-02-29	79.0	31.0	9.000	NaN	NaN	
34	2020-03-01	130.0	51.0	16.286	NaN	NaN	

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
35	2020-03-02	159.0	29.0	20.429	NaN	NaN	NaN
36	2020-03-03	196.0	37.0	25.571	NaN	NaN	NaN
37	2020-03-04	262.0	66.0	33.571	NaN	NaN	NaN
38	2020-03-05	482.0	220.0	62.286	NaN	NaN	NaN
39	2020-03-06	670.0	188.0	88.857	NaN	NaN	NaN
40	2020-03-07	799.0	129.0	102.857	NaN	NaN	NaN
41	2020-03-08	1040.0	241.0	130.000	NaN	NaN	NaN

◀ ▶

```
In [19]: x.loc[0:41,['total_deaths','new_deaths']] = 0
x.loc[0:4,'new_deaths_smoothed'] = 0
```

```
In [20]: x[x.reproduction_rate!=x.reproduction_rate]
```

Out[20]:

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
5	2020-02-01	8.0	3.0	1.143	0.0	0.0	0.0
6	2020-02-02	10.0	2.0	1.429	0.0	0.0	0.0
7	2020-02-03	12.0	2.0	1.571	0.0	0.0	0.0
8	2020-02-04	12.0	0.0	1.143	0.0	0.0	0.0
9	2020-02-05	12.0	0.0	1.143	0.0	0.0	0.0
10	2020-02-06	12.0	0.0	1.143	0.0	0.0	0.0
11	2020-02-07	13.0	1.0	1.143	0.0	0.0	0.0

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
12	2020-02-08	13.0	0.0	0.714	0.0	0.0	
13	2020-02-09	14.0	1.0	0.571	0.0	0.0	
14	2020-02-10	14.0	0.0	0.286	0.0	0.0	
15	2020-02-11	16.0	2.0	0.571	0.0	0.0	
16	2020-02-12	16.0	0.0	0.571	0.0	0.0	
17	2020-02-13	16.0	0.0	0.571	0.0	0.0	
18	2020-02-14	16.0	0.0	0.429	0.0	0.0	
19	2020-02-15	16.0	0.0	0.429	0.0	0.0	
20	2020-02-16	16.0	0.0	0.286	0.0	0.0	
21	2020-02-17	16.0	0.0	0.286	0.0	0.0	
22	2020-02-18	16.0	0.0	0.000	0.0	0.0	
23	2020-02-19	16.0	0.0	0.000	0.0	0.0	
24	2020-02-20	16.0	0.0	0.000	0.0	0.0	
25	2020-02-21	16.0	0.0	0.000	0.0	0.0	
26	2020-02-22	16.0	0.0	0.000	0.0	0.0	
27	2020-02-23	16.0	0.0	0.000	0.0	0.0	
28	2020-02-24	16.0	0.0	0.000	0.0	0.0	
29	2020-02-25	17.0	1.0	0.143	0.0	0.0	
30	2020-02-26	27.0	10.0	1.571	0.0	0.0	
31	2020-02-27	46.0	19.0	4.286	0.0	0.0	
32	2020-02-28	48.0	2.0	4.571	0.0	0.0	
33	2020-02-29	79.0	31.0	9.000	0.0	0.0	
34	2020-03-01	130.0	51.0	16.286	0.0	0.0	

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
682	2021-12-09	6442846.0	103018.0	52935.286	105000.0	799.0	3
683	2021-12-10	6496142.0	53296.0	51664.286	105510.0	510.0	3
684	2021-12-11	6528894.0	32752.0	50128.857	105642.0	132.0	3



In [21]: `x[x.reproduction_rate!=x.reproduction_rate]`

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
5	2020-02-01	8.0	3.0	1.143	0.0	0.0	0.0
6	2020-02-02	10.0	2.0	1.429	0.0	0.0	0.0
7	2020-02-03	12.0	2.0	1.571	0.0	0.0	0.0
8	2020-02-04	12.0	0.0	1.143	0.0	0.0	0.0
9	2020-02-05	12.0	0.0	1.143	0.0	0.0	0.0
10	2020-02-06	12.0	0.0	1.143	0.0	0.0	0.0
11	2020-02-07	13.0	1.0	1.143	0.0	0.0	0.0
12	2020-02-08	13.0	0.0	0.714	0.0	0.0	0.0
13	2020-02-09	14.0	1.0	0.571	0.0	0.0	0.0
14	2020-02-10	14.0	0.0	0.286	0.0	0.0	0.0
15	2020-02-11	16.0	2.0	0.571	0.0	0.0	0.0
16	2020-02-12	16.0	0.0	0.571	0.0	0.0	0.0

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
17	2020-02-13	16.0	0.0	0.571	0.0	0.0	
18	2020-02-14	16.0	0.0	0.429	0.0	0.0	
19	2020-02-15	16.0	0.0	0.429	0.0	0.0	
20	2020-02-16	16.0	0.0	0.286	0.0	0.0	
21	2020-02-17	16.0	0.0	0.286	0.0	0.0	
22	2020-02-18	16.0	0.0	0.000	0.0	0.0	
23	2020-02-19	16.0	0.0	0.000	0.0	0.0	
24	2020-02-20	16.0	0.0	0.000	0.0	0.0	
25	2020-02-21	16.0	0.0	0.000	0.0	0.0	
26	2020-02-22	16.0	0.0	0.000	0.0	0.0	
27	2020-02-23	16.0	0.0	0.000	0.0	0.0	
28	2020-02-24	16.0	0.0	0.000	0.0	0.0	
29	2020-02-25	17.0	1.0	0.143	0.0	0.0	
30	2020-02-26	27.0	10.0	1.571	0.0	0.0	
31	2020-02-27	46.0	19.0	4.286	0.0	0.0	
32	2020-02-28	48.0	2.0	4.571	0.0	0.0	
33	2020-02-29	79.0	31.0	9.000	0.0	0.0	
34	2020-03-01	130.0	51.0	16.286	0.0	0.0	
682	2021-12-09	6442846.0	103018.0	52935.286	105000.0	799.0	3
683	2021-12-10	6496142.0	53296.0	51664.286	105510.0	510.0	3
684	2021-12-11	6528894.0	32752.0	50128.857	105642.0	132.0	3



In [22]: `x[x.icu_patients!=x.icu_patients]`

Out[22]:

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
5	2020-02-01	8.0	3.0	1.143	0.0	0.0	0.0
6	2020-02-02	10.0	2.0	1.429	0.0	0.0	0.0
7	2020-02-03	12.0	2.0	1.571	0.0	0.0	0.0
8	2020-02-04	12.0	0.0	1.143	0.0	0.0	0.0
9	2020-02-05	12.0	0.0	1.143	0.0	0.0	0.0
10	2020-02-06	12.0	0.0	1.143	0.0	0.0	0.0
11	2020-02-07	13.0	1.0	1.143	0.0	0.0	0.0
12	2020-02-08	13.0	0.0	0.714	0.0	0.0	0.0
13	2020-02-09	14.0	1.0	0.571	0.0	0.0	0.0
14	2020-02-10	14.0	0.0	0.286	0.0	0.0	0.0
15	2020-02-11	16.0	2.0	0.571	0.0	0.0	0.0
16	2020-02-12	16.0	0.0	0.571	0.0	0.0	0.0
17	2020-02-13	16.0	0.0	0.571	0.0	0.0	0.0
18	2020-02-14	16.0	0.0	0.429	0.0	0.0	0.0
19	2020-02-15	16.0	0.0	0.429	0.0	0.0	0.0
20	2020-02-16	16.0	0.0	0.286	0.0	0.0	0.0
21	2020-02-17	16.0	0.0	0.286	0.0	0.0	0.0
22	2020-02-18	16.0	0.0	0.000	0.0	0.0	0.0

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoo
23	2020-02-19	16.0	0.0	0.000	0.0	0.0	-
24	2020-02-20	16.0	0.0	0.000	0.0	0.0	-
25	2020-02-21	16.0	0.0	0.000	0.0	0.0	-
26	2020-02-22	16.0	0.0	0.000	0.0	0.0	-
27	2020-02-23	16.0	0.0	0.000	0.0	0.0	-
28	2020-02-24	16.0	0.0	0.000	0.0	0.0	-
29	2020-02-25	17.0	1.0	0.143	0.0	0.0	-
30	2020-02-26	27.0	10.0	1.571	0.0	0.0	-
31	2020-02-27	46.0	19.0	4.286	0.0	0.0	-
32	2020-02-28	48.0	2.0	4.571	0.0	0.0	-
33	2020-02-29	79.0	31.0	9.000	0.0	0.0	-
34	2020-03-01	130.0	51.0	16.286	0.0	0.0	-
35	2020-03-02	159.0	29.0	20.429	0.0	0.0	-
36	2020-03-03	196.0	37.0	25.571	0.0	0.0	-
37	2020-03-04	262.0	66.0	33.571	0.0	0.0	-
38	2020-03-05	482.0	220.0	62.286	0.0	0.0	-
39	2020-03-06	670.0	188.0	88.857	0.0	0.0	-
40	2020-03-07	799.0	129.0	102.857	0.0	0.0	-
41	2020-03-08	1040.0	241.0	130.000	0.0	0.0	-
42	2020-03-09	1176.0	136.0	145.286	2.0	2.0	-
43	2020-03-10	1457.0	281.0	180.143	2.0	0.0	-
44	2020-03-11	1908.0	451.0	235.143	3.0	1.0	-
45	2020-03-12	2078.0	170.0	228.000	3.0	0.0	-

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoo
46	2020-03-13	3675.0	1597.0	429.286	7.0	4.0	
47	2020-03-14	4585.0	910.0	540.857	9.0	2.0	
48	2020-03-15	5795.0	1210.0	679.286	11.0	2.0	
49	2020-03-16	7272.0	1477.0	870.857	17.0	6.0	
50	2020-03-17	9257.0	1985.0	1114.286	24.0	7.0	
51	2020-03-18	12327.0	3070.0	1488.429	28.0	4.0	
52	2020-03-19	15320.0	2993.0	1891.714	44.0	16.0	

◀ ▶

In [23]: `x.loc[35:52,'icu_patients']=x.loc[35:52,'total_deaths']*3`

In [24]: `x.loc[35:52,:]`

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoo
35	2020-03-02	159.0	29.0	20.429	0.0	0.0	
36	2020-03-03	196.0	37.0	25.571	0.0	0.0	
37	2020-03-04	262.0	66.0	33.571	0.0	0.0	
38	2020-03-05	482.0	220.0	62.286	0.0	0.0	
39	2020-03-06	670.0	188.0	88.857	0.0	0.0	
40	2020-03-07	799.0	129.0	102.857	0.0	0.0	
41	2020-03-08	1040.0	241.0	130.000	0.0	0.0	
42	2020-03-09	1176.0	136.0	145.286	2.0	2.0	
43	2020-03-10	1457.0	281.0	180.143	2.0	0.0	
44	2020-03-11	1908.0	451.0	235.143	3.0	1.0	
45	2020-03-12	2078.0	170.0	228.000	3.0	0.0	
46	2020-03-13	3675.0	1597.0	429.286	7.0	4.0	

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
47	2020-03-14	4585.0	910.0	540.857	9.0	2.0	
48	2020-03-15	5795.0	1210.0	679.286	11.0	2.0	
49	2020-03-16	7272.0	1477.0	870.857	17.0	6.0	
50	2020-03-17	9257.0	1985.0	1114.286	24.0	7.0	
51	2020-03-18	12327.0	3070.0	1488.429	28.0	4.0	
52	2020-03-19	15320.0	2993.0	1891.714	44.0	16.0	

```
In [25]: x.loc[0:34, 'reproduction_rate' ]=1  
x.loc[0:34, 'icu_patients' ]=0
```

```
In [26]: x.isna().sum()
```

```
Out[26]: date          0  
total_cases      0  
new_cases        0  
new_cases_smoothed 0  
total_deaths     0  
new_deaths       0  
new_deaths_smoothed 0  
reproduction_rate 3  
icu_patients     0  
stringency_index 6  
dtype: int64
```

```
In [27]: Recovdata = pd.read_csv('time series covid19 recovered global.csv')
```

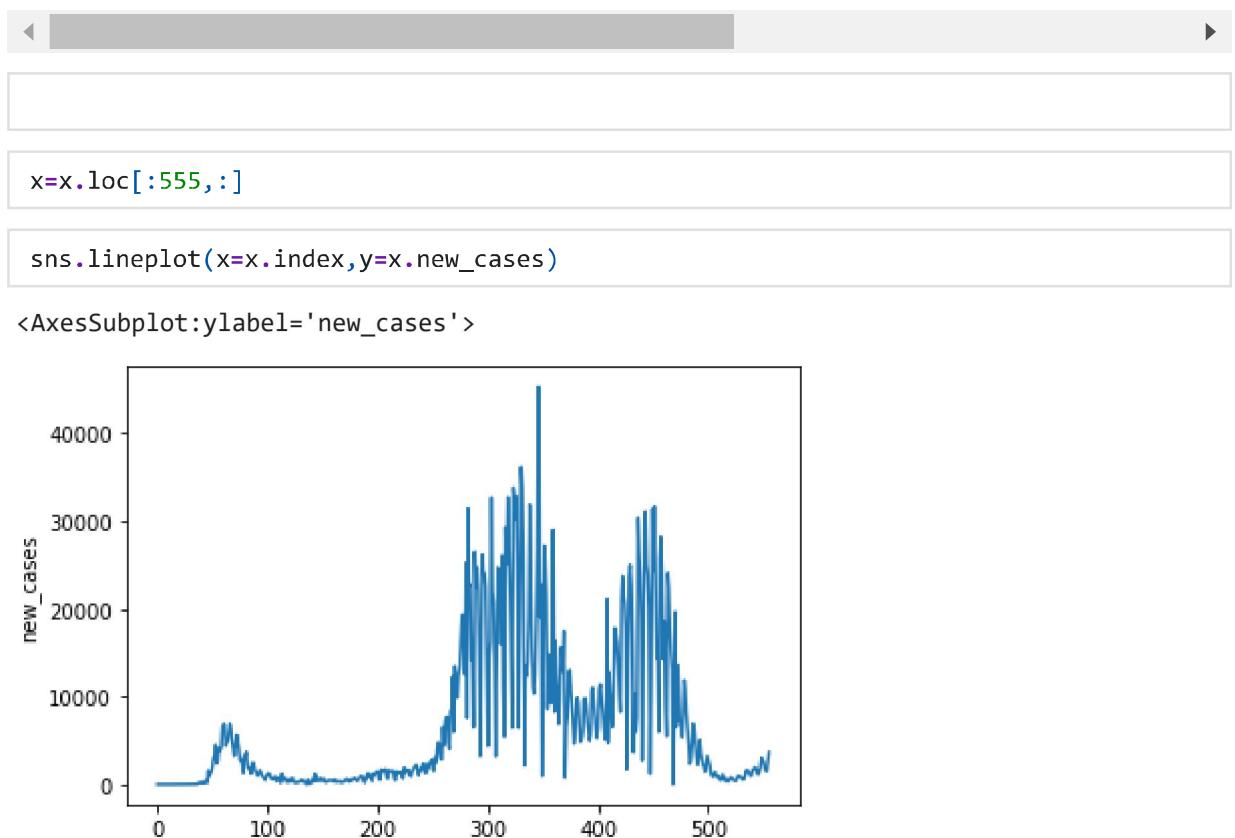
```
In [28]: x['recovered'] = Recovdata[Recovdata['Country/Region']=='Germany'].values[0][9:]
```

```
In [29]: x['recovered']=x['recovered'].astype(str).astype(int)
```

```
In [30]: x.loc[555:,:]
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
680	2021-12-07	6312346.0	93087.0	55540.286	104051.0	811.0	3
681	2021-12-08	6339828.0	27482.0	48686.857	104201.0	150.0	2
682	2021-12-09	6442846.0	103018.0	52935.286	105000.0	799.0	3
683	2021-12-10	6496142.0	53296.0	51664.286	105510.0	510.0	3
684	2021-12-11	6528894.0	32752.0	50128.857	105642.0	132.0	3

130 rows × 11 columns



In [33]:

X

Out[33]:

	Date	Open	Close	Min	Max	Volume
0	2020-01-27	1.0	1.0	0.000	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0
...

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	

556 rows × 11 columns



```
In [34]: for i in range(len(x)):
    if i ==0:
        x.loc[i,'new_recovered']=0
    else:
        x.loc[i,'new_recovered']=x.loc[i,'recovered']-x.loc[i-1,'recovered']
```

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexing.py:1596: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
self.obj[key] = _infer_fill_value(value)

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexing.py:1765: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

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

```
In [35]: x['new_recovered_smooth']=0
```

<ipython-input-35-39b21c5292e6>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
x['new_recovered_smooth']=0

```
In [36]: x
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	

556 rows × 13 columns



```
In [37]: for i in range(len(x)):
    if i <6:
        x.loc[i,'new_recovered_smooth']=x.loc[:i,'new_recovered'].values.sum()//7
    else:
        x.loc[i,'new_recovered_smooth']=x.loc[i-6:i,'new_recovered'].values.sum()//7
```

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexing.py:1765: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

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

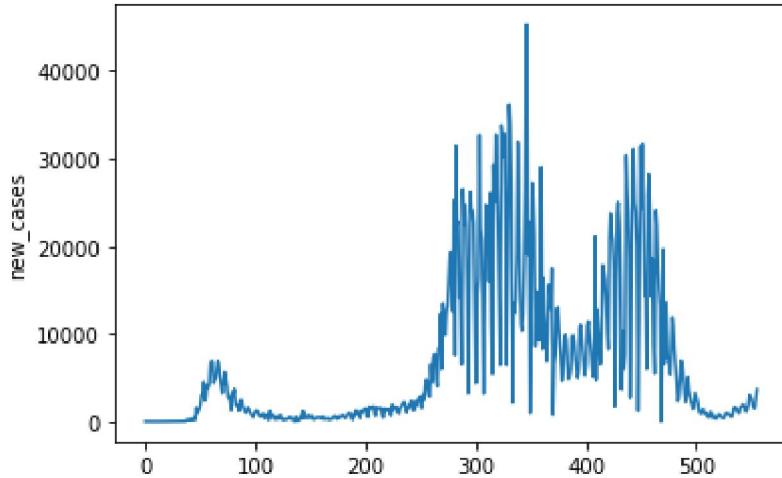
```
In [38]: x['active']=x.total_cases-x.recovered-x.total_deaths
```

<ipython-input-38-48260af4fa77>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
x['active']=x.total_cases-x.recovered-x.total_deaths

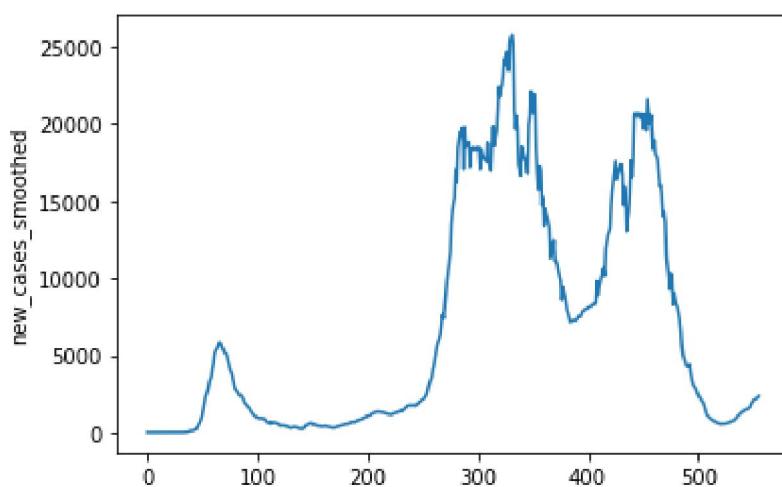
```
In [39]: sns.lineplot(x=x.index,y=x.new_cases)
```

```
Out[39]: <AxesSubplot:ylabel='new_cases'>
```



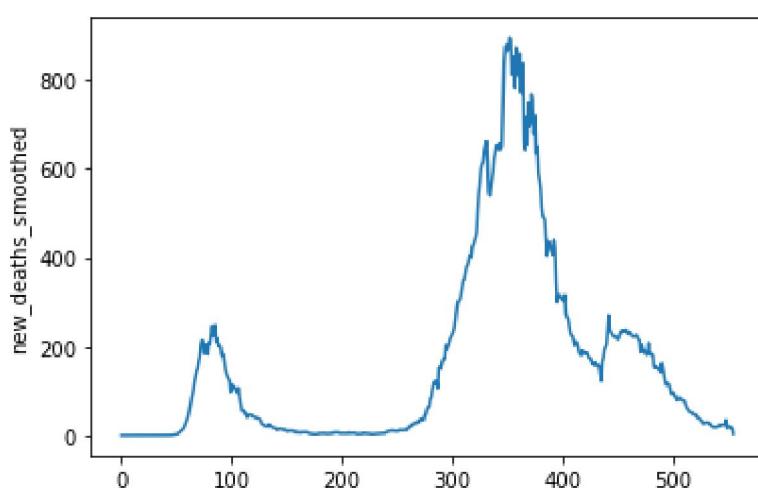
```
In [40]: sns.lineplot(x=x.index,y=x.new_cases_smoothed)
```

```
Out[40]: <AxesSubplot:ylabel='new_cases_smoothed'>
```



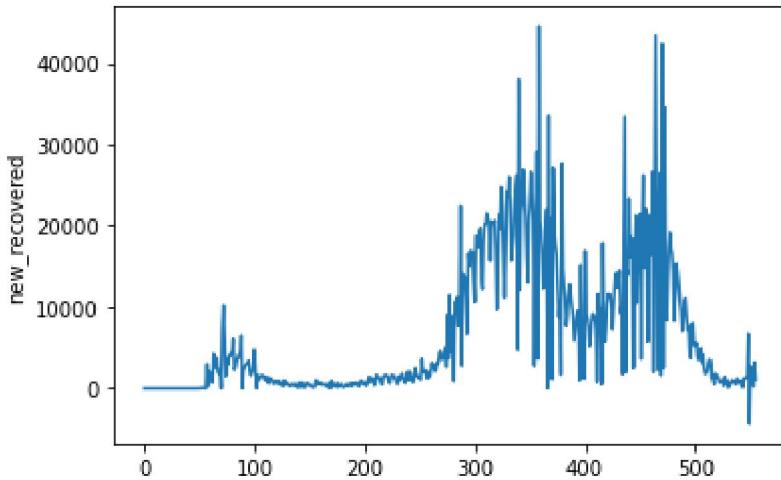
```
In [41]: sns.lineplot(x=x.index,y=x.new_deaths_smoothed)
```

```
Out[41]: <AxesSubplot:ylabel='new_deaths_smoothed'>
```



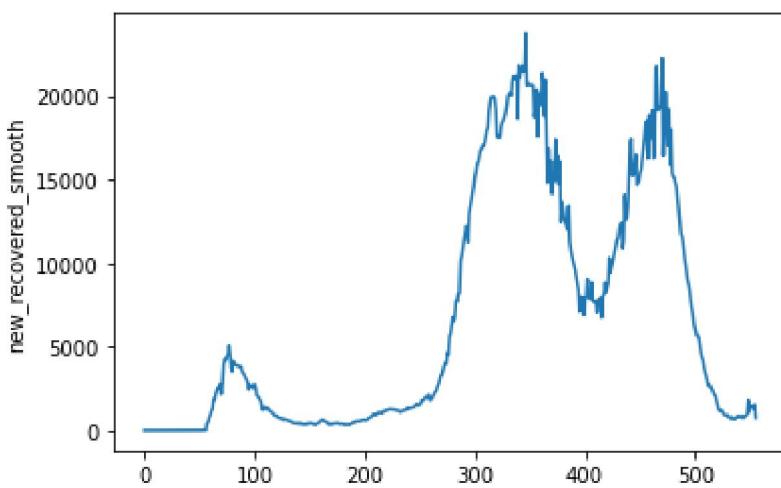
```
In [42]: sns.lineplot(x=x.index,y=x.new_recovered)
```

```
Out[42]: <AxesSubplot:ylabel='new_recovered'>
```



```
In [43]: sns.lineplot(x=x.index,y=x.new_recovered_smooth)
```

```
Out[43]: <AxesSubplot:ylabel='new_recovered_smooth'>
```



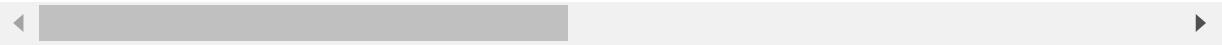
```
In [44]: x
```

```
Out[44]:
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	3.0
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	0.0
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	19.0

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
554	2021-08-03	3782344.0	2547.0		2263.286	91710.0	25.0
555	2021-08-04	3786003.0	3659.0		2350.143	91736.0	26.0

556 rows × 14 columns



```
In [45]: x['Total_RecovOrDead']=x['total_deaths'] + x['recovered']
```

```
<ipython-input-45-79eaf69e53f4>:1: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
`x['Total_RecovOrDead']=x['total_deaths'] + x['recovered']`

```
In [46]: x.loc[200,:]
```

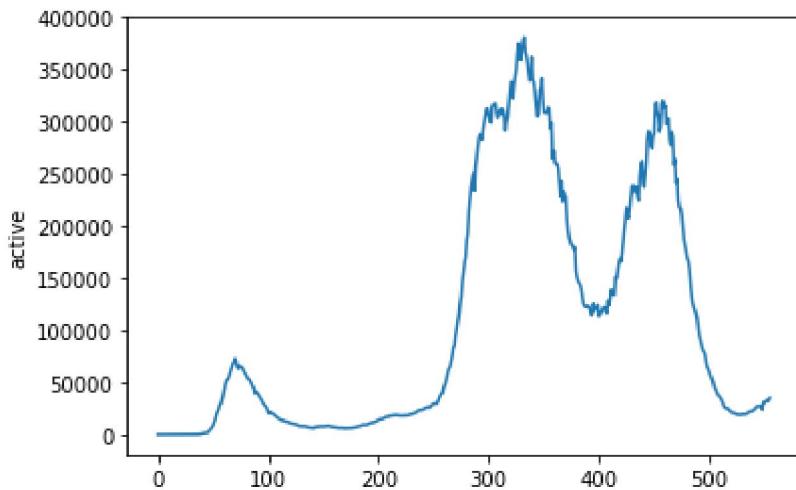
```
Out[46]: date                2020-08-14  
total_cases          223791  
new_cases             1510  
new_cases_smoothed    1085  
total_deaths          9230  
new_deaths              13  
new_deaths_smoothed      5  
reproduction_rate       1.23  
icu_patients            230  
stringency_index        56.94  
recovered           200440  
new_recovered           786  
new_recovered_smooth    643.571  
active                 14121  
Total_RecovOrDead       209670  
Name: 200, dtype: object
```

```
In [47]: x.dtypes
```

```
Out[47]: date                  object  
total_cases        float64  
new_cases         float64  
new_cases_smoothed float64  
total_deaths       float64  
new_deaths         float64  
new_deaths_smoothed float64  
reproduction_rate float64  
icu_patients       float64  
stringency_index    float64  
recovered          int32  
new_recovered       float64  
new_recovered_smooth float64  
active              float64  
Total_RecovOrDead    float64  
dtype: object
```

```
In [48]: sns.lineplot(x=x.index,y=x.active)
```

```
Out[48]: <AxesSubplot:ylabel='active'>
```



Vaccine Data

```
In [49]: DataVaccine = pd.read_csv('covid_de_vaccines.csv', index_col=False)
```

```
In [50]: DataVaccine
```

	date	doses	doses_first	doses_second	pfizer_cumul	moderna_cumul	astrazeneca_cumul	per
0	27-12-2020	24355	24344	11	24349	6	0	
1	28-12-2020	18106	18042	64	42455	6	0	
2	29-12-2020	50732	50079	652	93186	6	1	
3	30-12-2020	64133	64101	32	157319	6	1	
4	31-12-2020	50014	49927	87	207333	6	1	
...
322	14-11-2021	47005	10754	8208	89060578	9892224	12712087	
323	15-11-2021	235550	40983	36422	89283522	9900961	12712632	
324	16-11-2021	459895	61398	63310	89727092	9912770	12713091	
325	17-11-2021	526663	65076	63820	90234821	9927593	12713164	

	date	doses	doses_first	doses_second	pfizer_cumul	moderna_cumul	astrazeneca_cumul	pe
5	18- 11- 2021	522619	64549	65315	90737811	9942772	12713611	

327 rows × 10 columns

◀ ▶

```
In [51]: x[x.date == '2020-12-27']
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
335	2020-12-27	1658639.0	12399.0	20525.286	30297.0	351.0	5

◀ ▶

Out[52]:		date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
	551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	
	552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	
	553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	
	554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	
	555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	

◀ ▶

```
Out[53]: 0          11
         1          75
         2         727
         3         759
         4         846
         ...
217      43743110
218      43978363
219      44364634
220      44841022
221      45274515
Name: persons_full_cumul, Length: 222, dtype: int64
```

```
In [54]: temp = np.zeros(336)
```

```
In [55]: np.append(temp,DataVaccine.persons_full_cumul[:220])
```


0.0000000e+00, 0.0000000e+00, 0.0000000e+00, 0.0000000e+00,
 1.1000000e+01, 7.5000000e+01, 7.2700000e+02, 7.5900000e+02,
 8.4600000e+02, 8.5700000e+02, 9.3500000e+02, 9.4000000e+02,
 1.0560000e+03, 1.1430000e+03, 1.2290000e+03, 1.2500000e+03,
 1.3530000e+03, 1.4980000e+03, 1.6110000e+03, 1.6650000e+03,
 1.7280000e+03, 1.9290000e+03, 2.1420000e+03, 3.0620000e+03,
 4.2840000e+03, 2.1295000e+04, 3.6908000e+04, 7.2957000e+04,
 1.2672400e+05, 1.6764600e+05, 2.0123200e+05, 2.4777600e+05,
 2.6991200e+05, 3.0864900e+05, 3.5638100e+05, 4.1486400e+05,
 4.6058200e+05, 5.1847200e+05, 5.7197000e+05, 6.0343800e+05,
 6.6532200e+05, 7.3260600e+05, 8.3100600e+05, 9.0351400e+05,
 9.8170100e+05, 1.0302520e+06, 1.0555320e+06, 1.1119860e+06,
 1.1863460e+06, 1.2640810e+06, 1.3330860e+06, 1.4090600e+06,
 1.4548650e+06, 1.4807010e+06, 1.5363890e+06, 1.5928590e+06,
 1.6493080e+06, 1.7026490e+06, 1.7585820e+06, 1.7949120e+06,
 1.8232170e+06, 1.8816010e+06, 1.9421880e+06, 2.0016340e+06,
 2.0609940e+06, 2.1286350e+06, 2.1684490e+06, 2.1926470e+06,
 2.2426710e+06, 2.2949310e+06, 2.3657250e+06, 2.4290730e+06,
 2.4965260e+06, 2.5452670e+06, 2.5787950e+06, 2.6306640e+06,
 2.6856300e+06, 2.7530960e+06, 2.8140630e+06, 2.8851870e+06,
 2.9355820e+06, 2.9712810e+06, 3.0305950e+06, 3.0989220e+06,
 3.1801030e+06, 3.2559120e+06, 3.3405800e+06, 3.3939820e+06,
 3.4323080e+06, 3.5090330e+06, 3.5934260e+06, 3.6831260e+06,
 3.7644980e+06, 3.8535600e+06, 3.9160460e+06, 3.9693870e+06,
 4.0583680e+06, 4.1513120e+06, 4.2532520e+06, 4.3445340e+06,
 4.4126520e+06, 4.4773330e+06, 4.5279240e+06, 4.5983450e+06,
 4.6932460e+06, 4.7897910e+06, 4.8777000e+06, 4.9668560e+06,
 5.0320100e+06, 5.0824500e+06, 5.1589500e+06, 5.2327630e+06,
 5.3161400e+06, 5.3864830e+06, 5.4608670e+06, 5.5175730e+06,
 5.5566580e+06, 5.6182200e+06, 5.6811570e+06, 5.7545860e+06,
 5.8219920e+06, 5.8912350e+06, 5.9503960e+06, 5.9995910e+06,
 6.0737900e+06, 6.1659730e+06, 6.2933680e+06, 6.4273720e+06,
 6.5650020e+06, 6.6414190e+06, 6.7088980e+06, 6.8256250e+06,
 6.9876810e+06, 7.2209030e+06, 7.4455810e+06, 7.6846900e+06,
 7.8390610e+06, 7.9417390e+06, 8.1357490e+06, 8.4245300e+06,
 8.7922490e+06, 8.9533110e+06, 9.1840580e+06, 9.3498220e+06,
 9.4715530e+06, 9.6778500e+06, 1.0032869e+07, 1.0567797e+07,
 1.1061569e+07, 1.1507592e+07, 1.1766791e+07, 1.1927272e+07,
 1.2112276e+07, 1.2496371e+07, 1.3206952e+07, 1.3844658e+07,
 1.4393165e+07, 1.4651996e+07, 1.4838087e+07, 1.5207090e+07,
 1.5796244e+07, 1.6528840e+07, 1.6941730e+07, 1.7461939e+07,
 1.7778778e+07, 1.8011739e+07, 1.8433154e+07, 1.9146023e+07,
 2.0121545e+07, 2.0908449e+07, 2.1606893e+07, 2.1918825e+07,
 2.2136959e+07, 2.2577788e+07, 2.3277812e+07, 2.4194298e+07,
 2.4942290e+07, 2.5589585e+07, 2.5897379e+07, 2.6124491e+07,
 2.6546267e+07, 2.7247852e+07, 2.8094755e+07, 2.8641436e+07,
 2.9181746e+07, 2.9454853e+07, 2.9656908e+07, 3.0042507e+07,
 3.0586575e+07, 3.1158290e+07, 3.1688295e+07, 3.2155417e+07,
 3.2383572e+07, 3.2563757e+07, 3.2903845e+07, 3.3412525e+07,
 3.4085685e+07, 3.4671727e+07, 3.5210636e+07, 3.5458238e+07,
 3.5624926e+07, 3.5942464e+07, 3.6525808e+07, 3.7299238e+07,
 3.7897522e+07, 3.8407645e+07, 3.8608813e+07, 3.8748934e+07,
 3.9039548e+07, 3.9522563e+07, 4.0125231e+07, 4.0627799e+07,
 4.1086564e+07, 4.1245368e+07, 4.1337566e+07, 4.1604900e+07,
 4.2061459e+07, 4.2631466e+07, 4.3105591e+07, 4.3509498e+07,
 4.3656939e+07, 4.3743110e+07, 4.3978363e+07, 4.4364634e+07]

```
In [56]: x['Full_Vaccine']=np.append(temp,DataVaccine.persons_full_cumul[:220])
```

```
<ipython-input-56-02816c0f7f49>:1: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
x['Full_Vaccine']=np.append(temp,DataVaccine.persons_full_cumul[:220])
```

```
In [57]: x['PerDay_Vaccine']=np.append(temp,DataVaccine.vacc_per_day[:220])
```

```
<ipython-input-57-08e797372ee8>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
x['PerDay_Vaccine']=np.append(temp,DataVaccine.vacc_per_day[:220])
```

```
In [58]: x
```

```
Out[58]:
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	

556 rows × 17 columns

Model

```
In [59]: Pop_Sus=83019213
```

```
In [60]: import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt
from lmfit import minimize, Parameters, Parameter, report_fit
```

```
In [61]: def ode_model(z, t, beta, sigma, gamma):
```

```

"""
Reference https://www.idmod.org/docs/hiv/model-seir.html
"""

S, E, I, R = z
N = S + E + I + R
#print(t)
dSdt = -beta*S*I/N
dEdt = beta*S*I/N - sigma*E
dIdt = sigma*E - gamma*I
dRdt = gamma*I
return [dSdt, dEdt, dIdt, dRdt]

```

```
In [62]: def ode_solver(t, initialconditions, params):
    initE, initI, initR, initN = initialconditions
    beta, sigma, gamma = params['beta'].value, params['sigma'].value, params['gamma']
    initS = initN - (initE + initI + initR)
    res = odeint(ode_model, [initS, initE, initI, initR], t, args=(beta, sigma, gamma))
    return res
```

```
In [63]: def error(params, initialconditions, tspan, data):
    sol = ode_solver(tspan, initialconditions, params)
    return (sol[:, 2:4] - data).ravel()
```

```
In [64]: init_N=83019213
init_E=21121
init_I=223791
init_R=209670
initial_conditions = [init_E, init_I, init_R, init_N]
sigma = 1/5.2
gamma = 1/2.9
R0 = 1
beta = R0 * gamma
```

```
In [65]: params = Parameters()
params.add('beta', value=beta, min=0, max=10)
params.add('sigma', value=sigma, min=0, max=10)
params.add('gamma', value=gamma, min=0, max=10)
```

```
In [66]: params['beta'].value = beta
params['sigma'].value = sigma
params['gamma'].value = gamma
days = 400
tspan = np.arange(200, days, 1)
data = x.loc[200:(days-1), ['total_cases', 'Total_RecovOrDead']].values
#data = x.loc[0:(days-1), ['total_cases', 'Total_RecovOrDead', 'stringency_index']].v
```

```
In [67]: # fit model and find predicted values
result = minimize(error, params, args=(initial_conditions, tspan, data), method='leas
```

```
In [68]: result
```

Out[68]: **Fit Statistics**

fitting method	leastsq
# function evals	119
# data points	400
# variables	3

```

chi-square 1.0950e+13
reduced chi-square 2.7582e+10
Akaike info crit. 9619.16115
Bayesian info crit. 9631.13554

```

Variables

name	value	standard error	relative error	initial value	min	max	vary
beta	0.02432189	2.7123e-04	(1.12%)	0.3448275862068966	0.00000000	10.0000000	True
sigma	9.99998820	54.4192379	(544.19%)	0.1923076923076923	0.00000000	10.0000000	True
gamma	0.011111395	1.6974e-04	(1.53%)	0.3448275862068966	0.00000000	10.0000000	True

Correlations (unreported correlations are < 0.100)

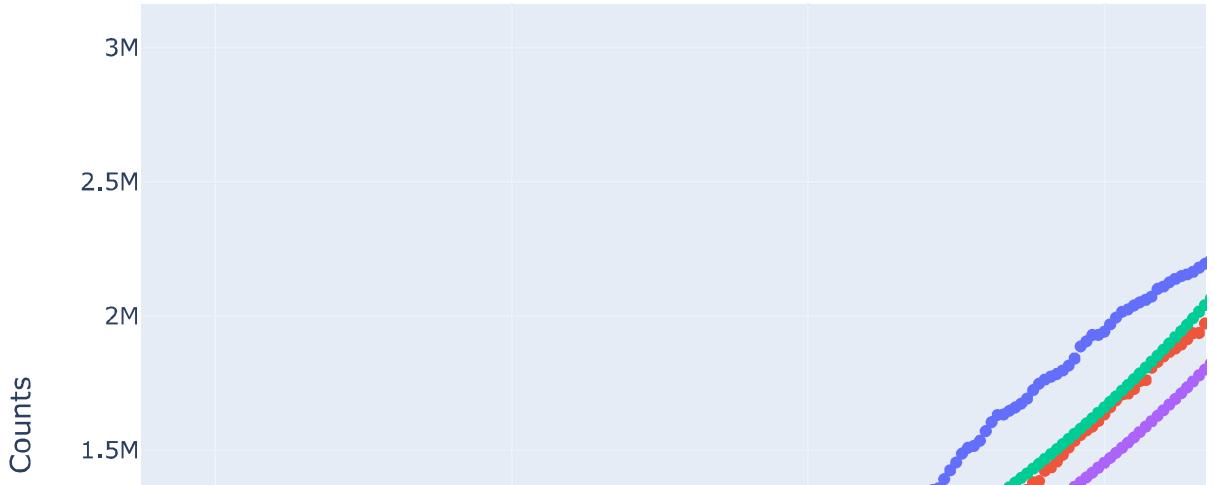
```

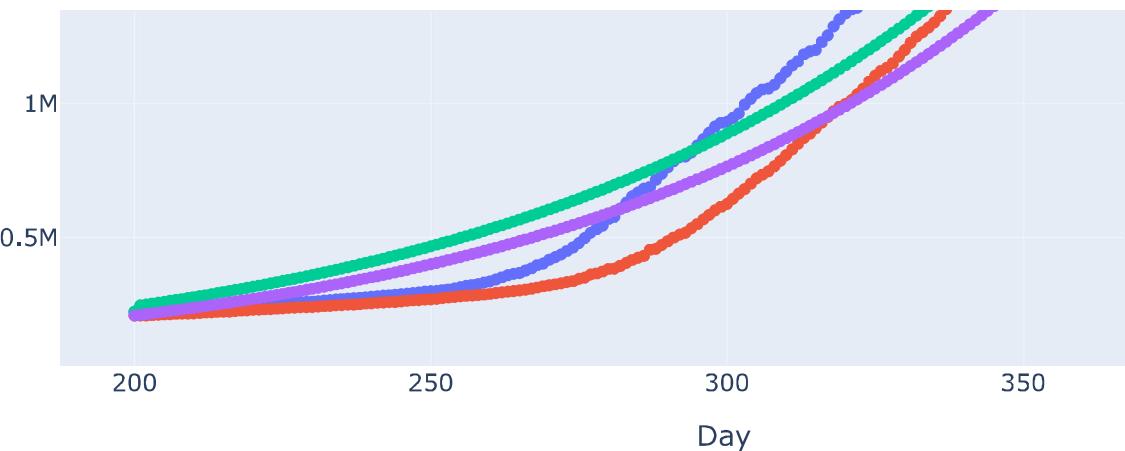
beta   sigma -0.8778
beta   gamma  0.8575
sigma  gamma -0.5586

```

```
In [69]: final = data + result.residual.reshape(data.shape)
fig = go.Figure()
fig.add_trace(go.Scatter(x=tspan, y=data[:, 0], mode='markers', name='Observed Infec')
fig.add_trace(go.Scatter(x=tspan, y=data[:, 1], mode='markers', name='Observed Recov')
fig.add_trace(go.Scatter(x=tspan, y=final[:, 0], mode='lines+markers', name='Fitted')
fig.add_trace(go.Scatter(x=tspan, y=final[:, 1], mode='lines+markers', name='Fitted')
fig.update_layout(title='Observed vs Fitted',
                  xaxis_title='Day',
                  yaxis_title='Counts',
                  title_x=0.5,
                  width=1000, height=600
                 )
```

Observed vs Fitted





In [782...]

x

Out[782...]

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	3.0
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	0.0
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	19.0
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	25.0
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	26.0

556 rows × 18 columns

In [929...]

x['PerDay_Vaccine+100']=x['PerDay_Vaccine']+100000

<ipython-input-929-1d438aaac3c7>:1: SettingWithCopyWarning:

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

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

In [930...]

x

Out[930...]

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	

556 rows × 18 columns

◀ ▶

In [931...]

params

Out[931...]

	name	value	initial value	min	max	vary
	beta	1.50000000	1.5000000000000002	0.00000000	10.0000000	True
	sigma	0.01111111	0.01111111111111112	0.00000000	10.0000000	True
	gamma	0.05000000	0.05	0.00000000	10.0000000	True

In [932...]

```
def ode_model(z, t, stringency, vaccine, beta, sigma, gamma):
    """
    Reference https://www.idmod.org/docs/hiv/model-seir.html
    """
    S, E, I, R = z
    N = S + E + I + R
    temp=((100-int(stringency[min(int(t-200),299)][0]))/100)**2.5
    #print(t,vaccine[0][0])
    #print(t,(100-int(stringency[min(int(t-200),199)][0]))/(100))
    dSdt = -beta*S*I*temp/N - (vaccine[min(int(t-200),299)][0])
    dEdt = beta*S*I*temp/N - sigma*E
    dIdt = sigma*E - gamma*I
```

```
dRdt = gamma*I  
return [dSdt, dEdt, dIdt, dRdt]
```

```
In [933...]  
def ode_model1(z, t, stringency, vaccine, beta, sigma, gamma):  
    """  
        Reference https://www.idmod.org/docs/hiv/model-seir.html  
    """  
    S, E, I, R, V = z  
    N = S + E + I + R + V  
    temp=((100-int(stringency[min(int(t-200),299)][0]))/100)**2.5  
    #print(t,vaccine[0][0])  
    #print(t,(100-int(stringency[min(int(t-200),199)][0]))/(100))  
    dSdt = -beta*S*I*temp/N - (vaccine[min(int(t-200),299)][0])  
    dEdt = beta*S*I*temp/N - sigma*E  
    dIdt = sigma*E - gamma*I  
    dRdt = gamma*I  
    dVdt = (vaccine[min(int(t-200),299)][0])  
    return [dSdt, dEdt, dIdt, dRdt,dVdt]
```

```
In [934...]  
def ode_solver(t, initialconditions, stringency,vaccine, params):  
    initE, initI, initR, initN = initialconditions  
    beta, sigma, gamma = params['beta'].value, params['sigma'].value, params['gamma'].value  
    initS = initN - (initE + initI + initR)  
    res = odeint(ode_model, [initS, initE, initI, initR], t, args=(stringency,vaccine))  
    return res
```

```
In [935...]  
def ode_solver1(t, initialconditions, stringency,vaccine, params):  
    initE, initI, initR, initN, initV = initialconditions  
    beta, sigma, gamma = params['beta'].value, params['sigma'].value, params['gamma'].value  
    initS = initN - (initE + initI + initR + initV)  
    res = odeint(ode_model1, [initS, initE, initI, initR, initV], t, args=(stringency,vaccine))  
    return res
```

```
In [936...]  
def error(params, initialconditions, stringency,vaccine,tspan, data):  
    sol = ode_solver(tspan, initialconditions, stringency,vaccine, params)  
    return (sol[:, 2:4] - data).ravel()
```

```
In [937...]  
def error1(params, initialconditions, stringency,vaccine,tspan, data):  
    sol = ode_solver1(tspan, initialconditions, stringency,vaccine, params)  
    return (sol[:, 2:4] - data).ravel()
```

```
In [938...]  
params = Parameters()  
params.add('beta', value=beta, min=0, max=10)  
params.add('sigma', value=sigma, min=0, max=10)  
params.add('gamma', value=gamma, min=0, max=10)
```

```
In [939...]  
init_N=83019213  
init_E=224000  
init_I=14000  
init_R=209000  
init_V=0  
initial_conditions = [init_E, init_I, init_R, init_N, init_V]  
sigma = 1/90  
gamma = 1/20  
R0 = 2  
beta = 1.5*R0 * gamma*10  
params['beta'].value = beta  
params['sigma'].value = sigma  
params['gamma'].value = gamma  
days = 500  
tspan = np.arange(200, days, 1)
```

```
#data = x.loc[0:(days-1), ['total_cases', 'Total_RecovOrDead']].values  
data = x.loc[200:(days-1), ['active', 'Total_RecovOrDead']].values
```

```
In [940...]: Stringent = x.loc[200:(days-1), ['stringency_index']].values
```

```
In [941...]: Vaccine = x.loc[200:(days-1), ['PerDay_Vaccine']].values
```

fit model and find predicted values

```
result = minimize(error, params, args=(initial_conditions, tspan, data), method='leastsq')
```

```
In [942...]: # fit model and find predicted values  
result = minimize(error1, params, args=(initial_conditions, Stringent, Vaccine, tspan,
```

```
In [943...]: result
```

```
Out[943...]
```

Fit Statistics

```
fitting method      leastsq  
# function evals       77  
# data points        600  
# variables          3  
chi-square    3.9983e+12  
reduced chi-square 6.6974e+09  
Akaike info crit.  13577.9803  
Bayesian info crit. 13591.1711
```

Variables

name	value	standard error	relative error	initial value	min	max	vary	
beta	1.62902878	0.00216977	(0.13%)	1.5000000000000002	0.00000000	10.0000000	True	
sigma	0.01079713	8.8424e-05	(0.82%)	0.01111111111111112	0.00000000	10.0000000	True	
gamma	0.05215794	5.2234e-04	(1.00%)		0.05	0.00000000	10.0000000	True

Correlations (unreported correlations are < 0.100)

```
sigma gamma  0.8230
```

```
beta sigma -0.8167
```

```
beta gamma -0.3763
```

```
In [944...]: result.params
```

name	value	standard error	relative error	initial value	min	max	vary
------	-------	----------------	----------------	---------------	-----	-----	------

```

beta  1.62902878   0.00216977   (0.13%)    1.5000000000000002  0.00000000  10.0000000  True
sigma 0.01079713   8.8424e-05   (0.82%)    0.01111111111111112  0.00000000  10.0000000  True
gamma 0.05215794   5.2234e-04   (1.00%)          0.05  0.00000000  10.0000000  True

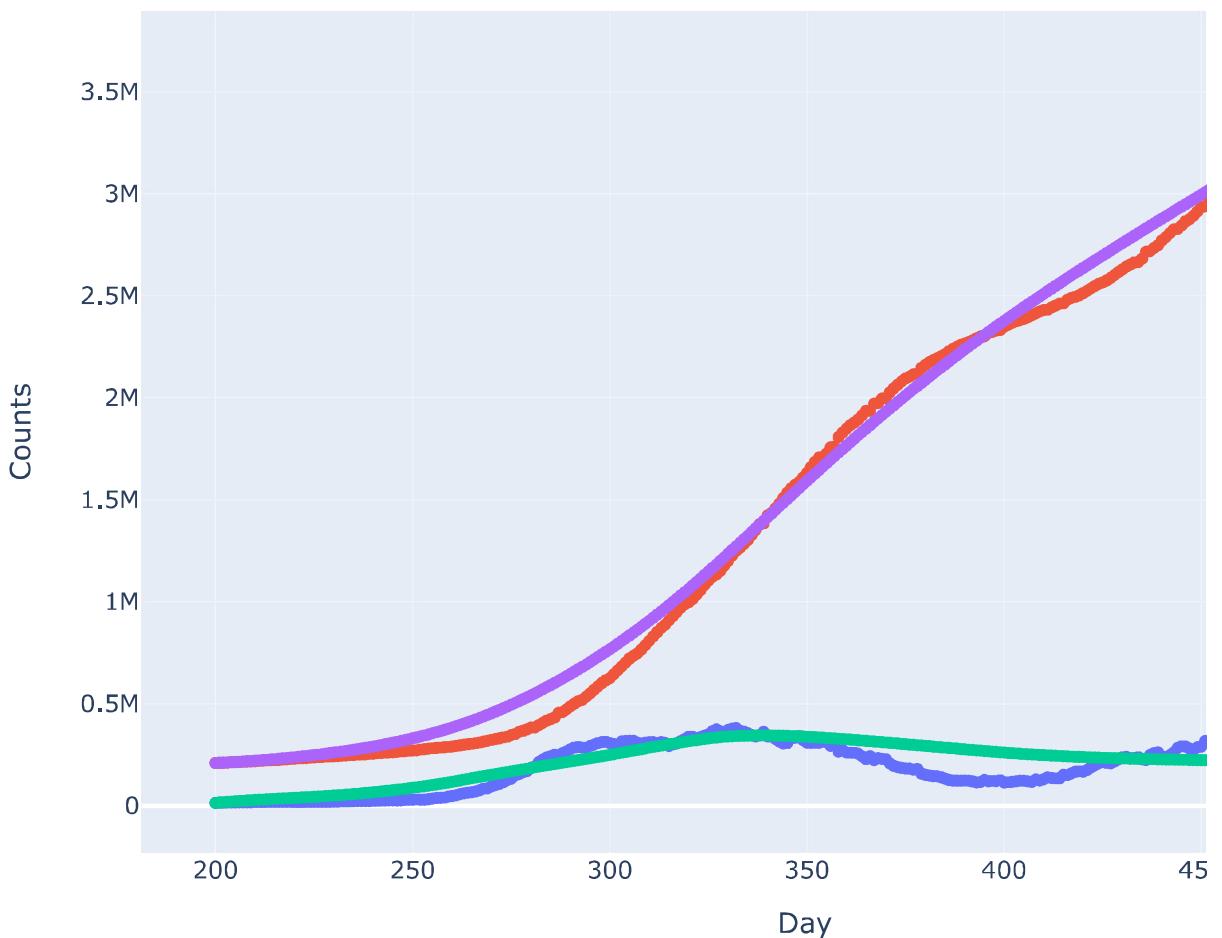
```

```

In [945... final = data + result.residual.reshape(data.shape)
fig = go.Figure()
fig.add_trace(go.Scatter(x=tspan, y=data[:, 0], mode='markers', name='Observed Infec')
fig.add_trace(go.Scatter(x=tspan, y=data[:, 1], mode='markers', name='Observed Recov')
fig.add_trace(go.Scatter(x=tspan, y=final[:, 0], mode='lines+markers', name='Fitted')
fig.add_trace(go.Scatter(x=tspan, y=final[:, 1], mode='lines+markers', name='Fitted')
fig.update_layout(title='Observed vs Fitted',
                  xaxis_title='Day',
                  yaxis_title='Counts',
                  title_x=0.5,
                  width=1000, height=600
)

```

Observed vs Fitted



```

In [946... final = data + result.residual.reshape(data.shape)
fig = go.Figure()
fig.add_trace(go.Scatter(x=tspan, y=data[:, 0], mode='markers', name='Observed Infec')
fig.add_trace(go.Scatter(x=tspan, y=data[:, 1], mode='markers', name='Observed Recov')
fig.add_trace(go.Scatter(x=tspan, y=final[:, 0], mode='lines+markers', name='Fitted')

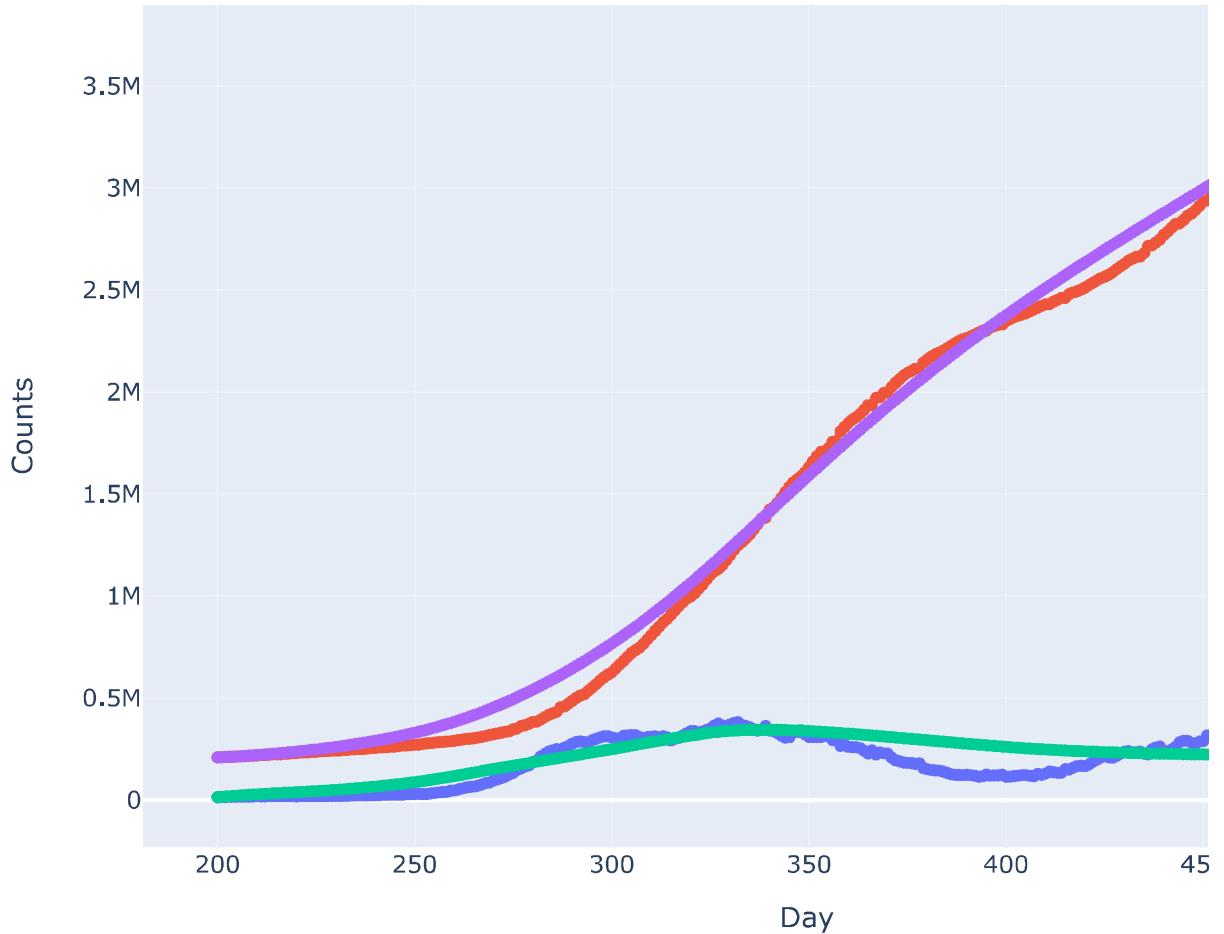
```

```

fig.add_trace(go.Scatter(x=tspan, y=final[:, 1], mode='lines+markers', name='Fitted')
fig.update_layout(title='Observed vs Fitted',
                  xaxis_title='Day',
                  yaxis_title='Counts',
                  title_x=0.5,
                  width=1000, height=600
)

```

Observed vs Fitted



◀

▶

```

In [947...]: Stringent1 = x.loc[200:(days-1), ['stringency_index']].values
In [948...]: Vaccine1 = x.loc[200:(days-1), ['PerDay_Vaccine']].values
In [949...]: result.params.values()
Out[949...]: dict_values([<Parameter 'beta', value=1.629028775350993 +/- 0.00217, bounds=[0:10]>,
<Parameter 'sigma', value=0.01079713488908729 +/- 8.84e-05, bounds=[0:10]>, <Parameter 'gamma', value=0.052157940615192855 +/- 0.000522, bounds=[0:10]>])
In [950...]: params1 = Parameters()
          params1.add('beta', value=beta, min=0, max=10)
          params1.add('sigma', value=sigma, min=0, max=10)
          params1.add('gamma', value=gamma, min=0, max=10)
          params1['beta'].value = 1.62902878

```

```
params1['sigma'].value = 0.01079713
params1['gamma'].value = 0.05215794
```

```
In [951...]: sws = ode_solver1(tspan, initial_conditions, Stringent1, Vaccine1, params1)
```

```
In [952...]: sws.shape
```

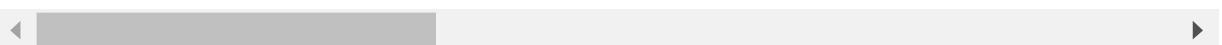
```
Out[952...]: (300, 5)
```

```
In [953...]: x
```

```
Out[953...]:
```

	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smo
0	2020-01-27	1.0	1.0	0.000	0.0	0.0	0.0
1	2020-01-28	4.0	3.0	0.000	0.0	0.0	0.0
2	2020-01-29	4.0	0.0	0.000	0.0	0.0	0.0
3	2020-01-30	4.0	0.0	0.000	0.0	0.0	0.0
4	2020-01-31	5.0	1.0	0.000	0.0	0.0	0.0
...
551	2021-07-31	3776724.0	1806.0	2122.143	91666.0	3.0	3.0
552	2021-08-01	3778277.0	1553.0	2179.857	91666.0	0.0	0.0
553	2021-08-02	3779797.0	1520.0	2193.714	91685.0	19.0	19.0
554	2021-08-03	3782344.0	2547.0	2263.286	91710.0	25.0	25.0
555	2021-08-04	3786003.0	3659.0	2350.143	91736.0	26.0	26.0

556 rows × 18 columns



```
In [954...]: ddd = x.loc[200:(days-1+100), ['active']].values
ffff = x.loc[200:(days-1+100), ['Total_RecovOrDead']].values
```

```
In [955...]: ddd.shape
```

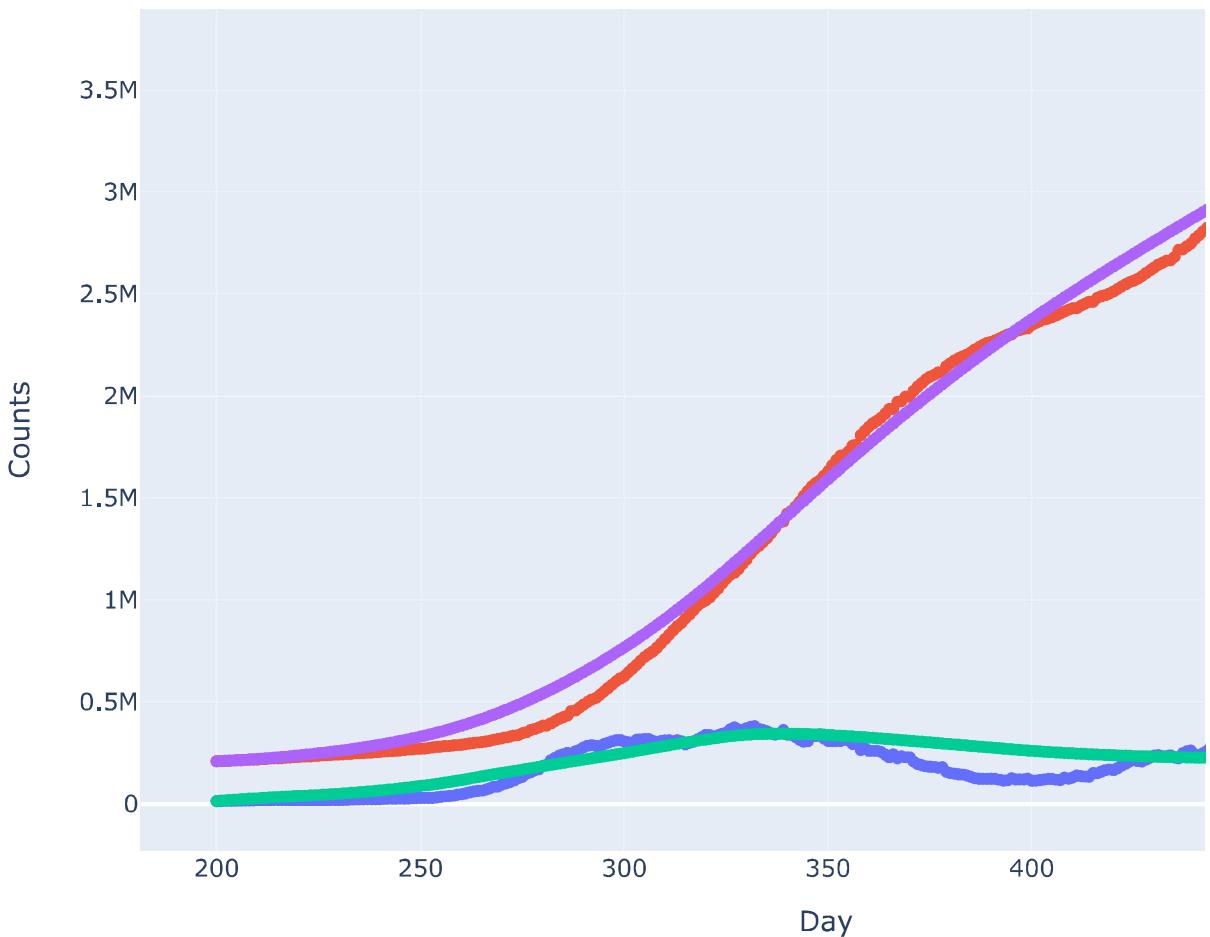
```
Out[955...]: (356, 1)
```

```
In [956...]: tspan1=np.arange(200,500,1)
```

```
In [957...]: fig = go.Figure()
fig.add_trace(go.Scatter(x=tspan, y=ddd[:, 0], mode='markers', name='Observed Infect')
fig.add_trace(go.Scatter(x=tspan, y=ffff[:, 0], mode='markers', name='Observed Recove')
fig.add_trace(go.Scatter(x=tspan, y=sws[:, 2], mode='lines+markers', name='Fitted In')
fig.add_trace(go.Scatter(x=tspan, y=sws[:, 3], mode='lines+markers', name='Fitted Re')
fig.update_layout(title='Observed vs Fitted',
```

```
xaxis_title='Day',
yaxis_title='Counts',
title_x=0.5,
width=1000, height=600
)
```

Observed vs Fitted



```
In [958... Vaccine2 = x.loc[200:(days-1), ['PerDay_Vaccine+100']].values
In [959... sws100 = ode_solver1(tspan1, initial_conditions, Stringent1,Vaccine2, params1)
In [960... sws100
Out[960... array([[8.25722130e+07, 2.24000000e+05, 1.40000000e+04, 2.09000000e+05,
   0.00000000e+00],
   [8.24691291e+07, 2.24662078e+05, 1.56482934e+04, 2.09773539e+05,
   1.00000000e+05],
   [8.23657146e+07, 2.25645793e+05, 1.72215085e+04, 2.10631060e+05,
   2.00000000e+05],
   ...,
   [3.18989864e+07, 5.88090764e+05, 1.32415904e+05, 2.92093880e+06,
   4.74787811e+07],
   [3.15608778e+07, 5.86894309e+05, 1.31867036e+05, 2.92783096e+06,
   4.78117429e+07],
   [3.10344067e+07, 5.85620261e+05, 1.31333088e+05, 2.93469488e+06,
   4.83331581e+07]])
```

```
In [961...]
```

```
sws
```

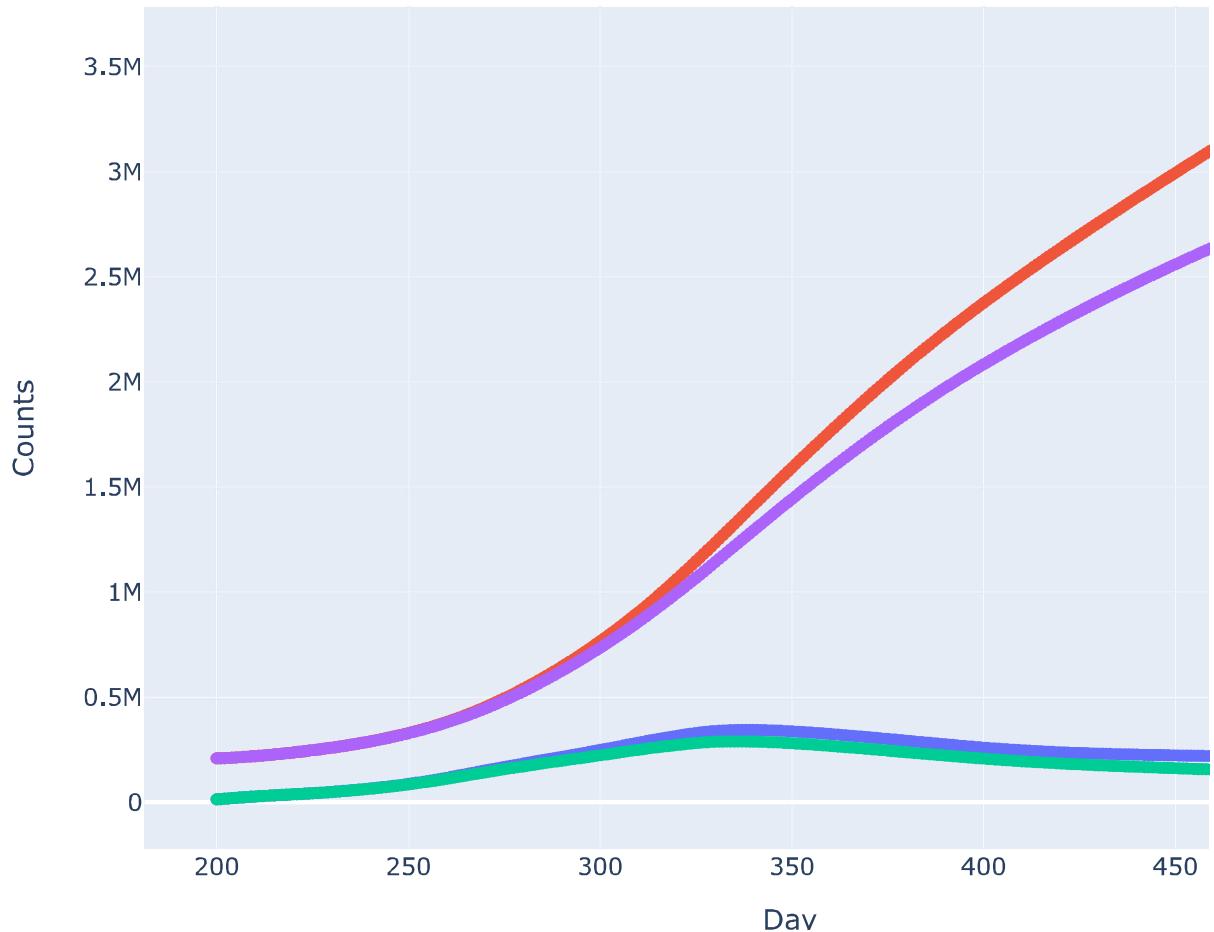
```
Out[961...]: array([[8.25722130e+07, 2.24000000e+05, 1.40000000e+04, 2.09000000e+05,
       0.0000000e+00],
      [8.25691272e+07, 2.24663975e+05, 1.56482997e+04, 2.09773540e+05,
       0.0000000e+00],
      [8.25657065e+07, 2.25653894e+05, 1.72215630e+04, 2.10631062e+05,
       0.0000000e+00],
      ...,
      [6.05051427e+07, 1.00278935e+06, 2.08637342e+05, 3.52386887e+06,
       1.77787748e+07],
      [6.02567189e+07, 1.00739989e+06, 2.08608467e+05, 3.53474999e+06,
       1.80117358e+07],
      [5.98199292e+07, 1.01187332e+06, 2.08628878e+05, 3.54563090e+06,
       1.84331507e+07]])
```

```
In [ ]:
```

```
In [963...]
```

```
fig = go.Figure()
fig.add_trace(go.Scatter(x=tspan, y=sws[:, 2], mode='lines+markers', name='Fitted In')
fig.add_trace(go.Scatter(x=tspan, y=sws[:, 3], mode='lines+markers', name='Fitted Re')
fig.add_trace(go.Scatter(x=tspan, y=sws100[:, 2], mode='lines+markers', name='Fitted In')
fig.add_trace(go.Scatter(x=tspan, y=sws100[:, 3], mode='lines+markers', name='Fitted Re')
fig.update_layout(title='Observed vs Fitted',
                  xaxis_title='Day',
                  yaxis_title='Counts',
                  title_x=0.5,
                  width=1000, height=600
                 )
```

Observed vs Fitted



◀  ▶

In []: