

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn.impute import SimpleImputer
```

```
df = pd.read_csv
```

```
df.head(10)
```

```
-----
-----
NameError                                Traceback (most recent call
last)
```

```
Cell In[5], line 1
----> 1 df.head(10)
```

```
NameError: name 'df' is not defined
```

```
df.drop(['SNo', 'Last Update'], axis=1, inplace=True)
df.rename(columns={'ObservationDate': 'Date', 'Province/State': 'State', '
Country/Region': 'Country'}, inplace=True)
```

```
-----
-----
NameError                                Traceback (most recent call
last)
```

```
Cell In[6], line 1
----> 1 df.drop(['SNo', 'Last Update'], axis=1, inplace=True)
      2
```

```
df.rename(columns={'ObservationDate': 'Date', 'Province/State': 'State', '
Country/Region': 'Country'}, inplace=True)
```

```
NameError: name 'df' is not defined
```

```
df['Date'] = pd.to_datetime(df['Date'])
```

```
-----
-----
NameError                                Traceback (most recent call
last)
```

```
Cell In[7], line 1
----> 1 df['Date'] = pd.to_datetime(df['Date'])
```

```
NameError: name 'df' is not defined
```

```
imputer = SimpleImputer(strategy='constant')
df2 = pd.DataFrame(imputer.fit_transform(df), columns=df.columns)
```

```
df3 = df2.groupby(['Country', 'Date'])
[['Country', 'Date', 'Confirmed', 'Deaths', 'Recovered']].sum().reset_inde
x()
```

```
df3.head(20)
```

	Country	Date	Confirmed	Deaths	Recovered
0	Azerbaijan	2020-02-28	1	0	0
1	('St. Martin',)	2020-03-10	2	0	0
2	Afghanistan	2020-02-24	1	0	0
3	Afghanistan	2020-02-25	1	0	0
4	Afghanistan	2020-02-26	1	0	0
5	Afghanistan	2020-02-27	1	0	0
6	Afghanistan	2020-02-28	1	0	0
7	Afghanistan	2020-02-29	1	0	0
8	Afghanistan	2020-03-01	1	0	0
9	Afghanistan	2020-03-02	1	0	0
10	Afghanistan	2020-03-03	1	0	0
11	Afghanistan	2020-03-04	1	0	0
12	Afghanistan	2020-03-05	1	0	0
13	Afghanistan	2020-03-06	1	0	0
14	Afghanistan	2020-03-07	1	0	0
15	Afghanistan	2020-03-08	4	0	0
16	Afghanistan	2020-03-09	4	0	0
17	Afghanistan	2020-03-10	5	0	0
18	Afghanistan	2020-03-11	7	0	0
19	Afghanistan	2020-03-12	7	0	0

```
countries = df3['Country'].unique()
```

```
len(countries)
```

```
171
```

```
for idx in range(0,len(countries)):
```

```
    C = df3[df3['Country']==countries[idx]].reset_index()
```

```
plt.scatter(np.arange(0,len(C)),C['Confirmed'],color='blue',label='Confirmed')
```

```
plt.scatter(np.arange(0,len(C)),C['Recovered'],color='green',label='Recovered')
```

```
plt.scatter(np.arange(0,len(C)),C['Deaths'],color='red',label='Deaths')
```

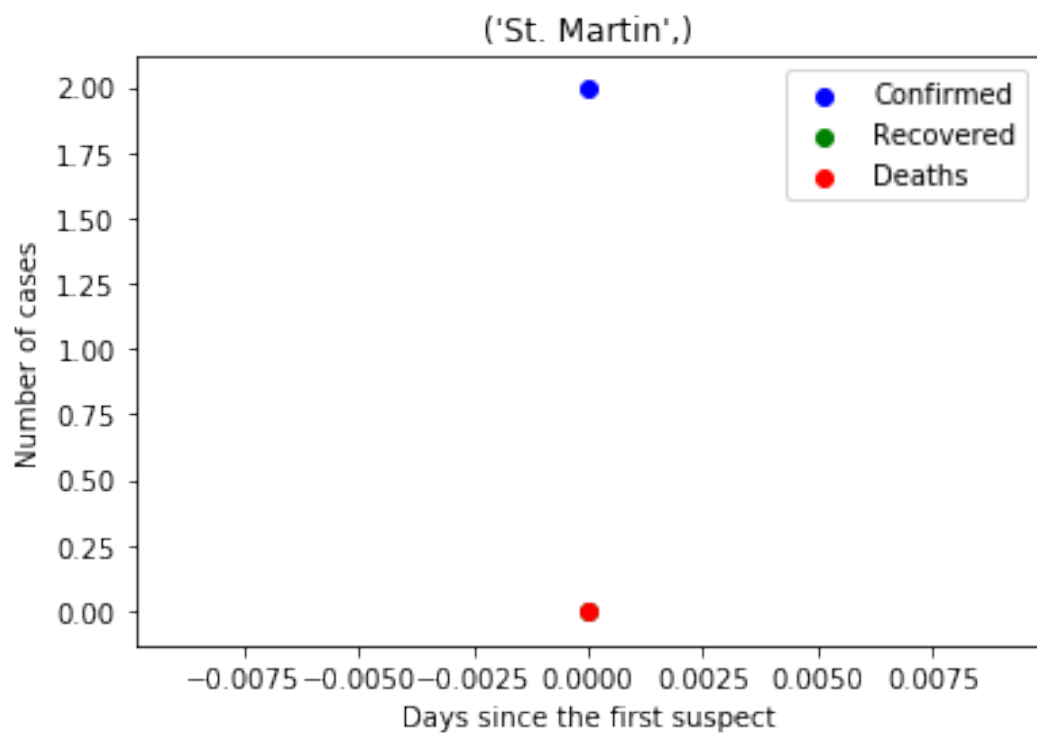
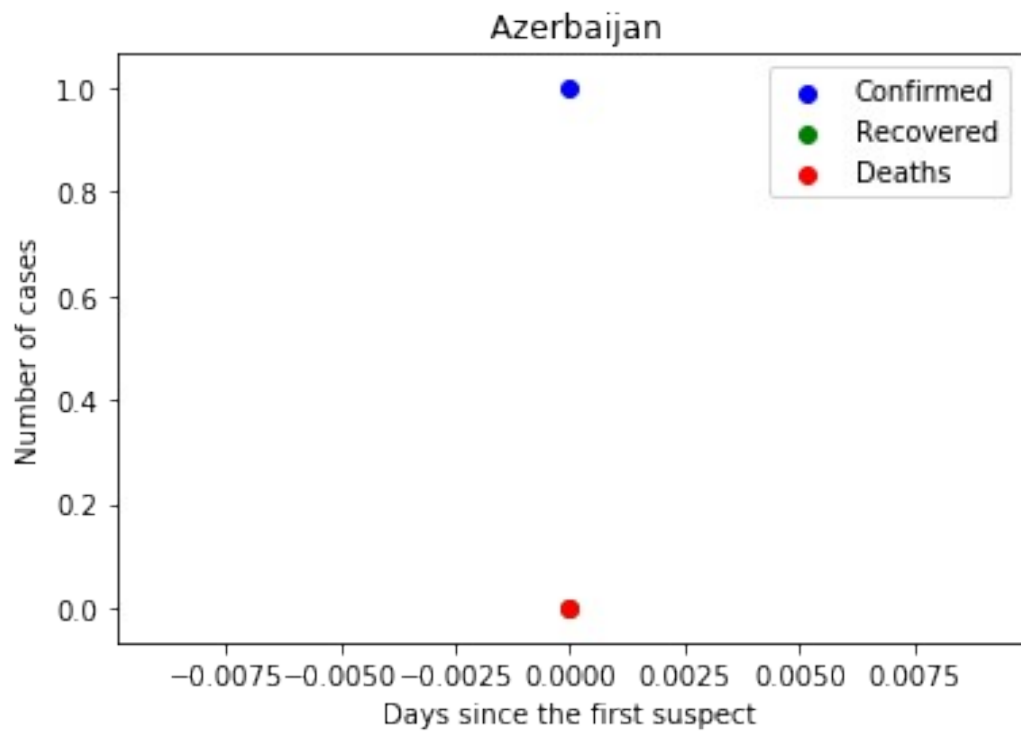
```
    plt.title(countries[idx])
```

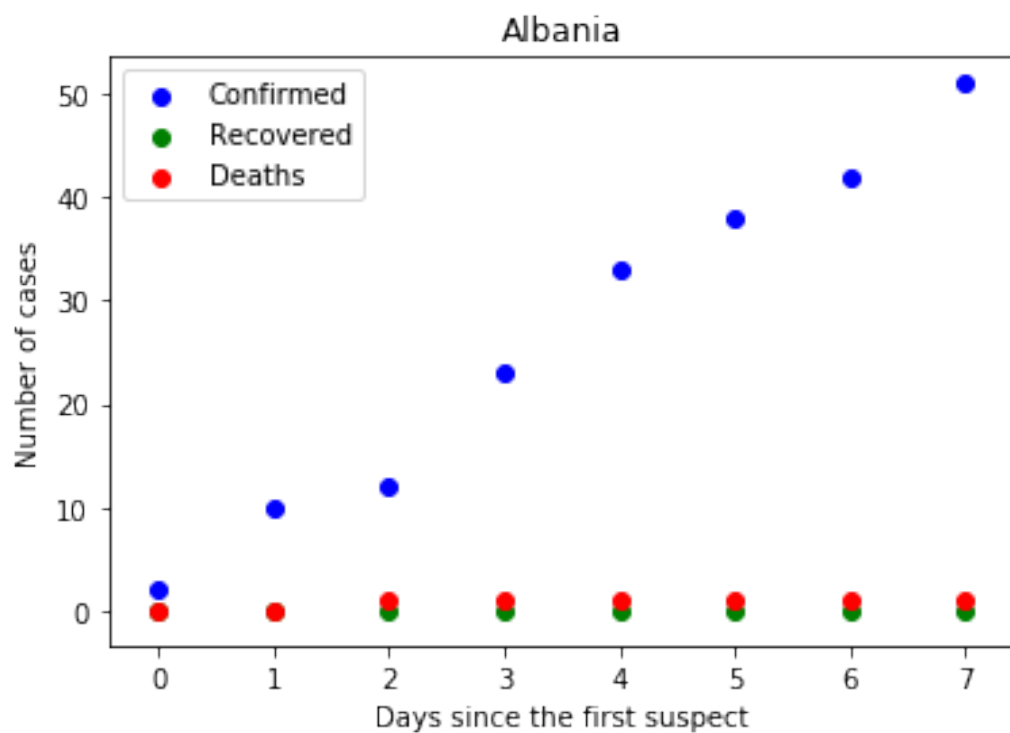
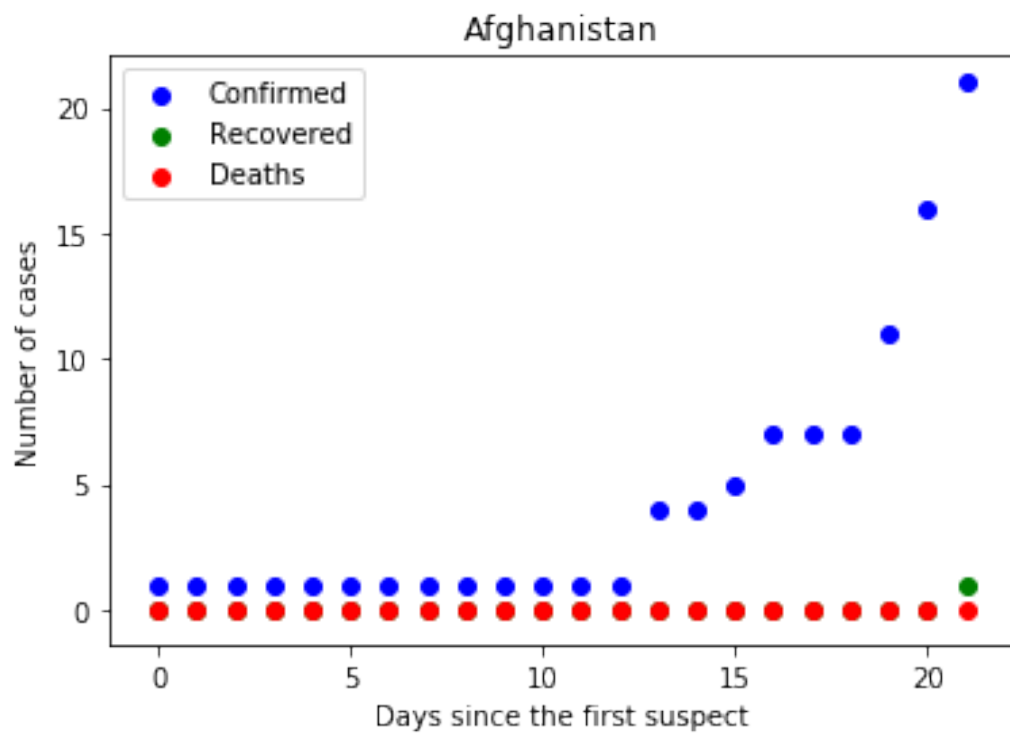
```
    plt.xlabel('Days since the first suspect')
```

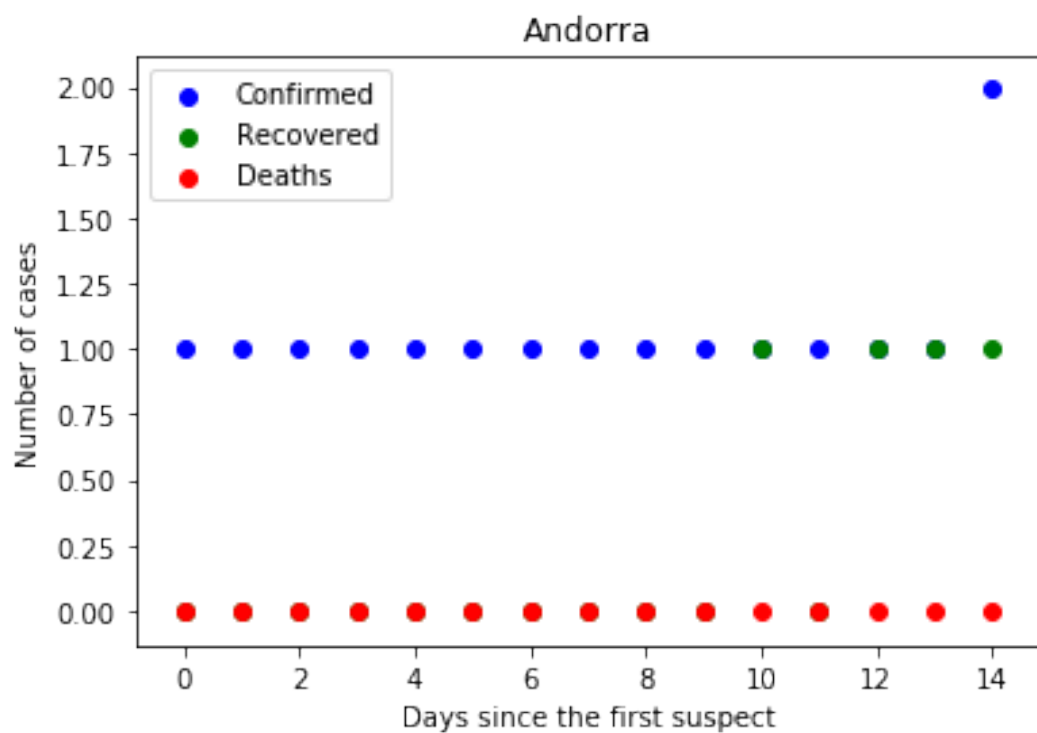
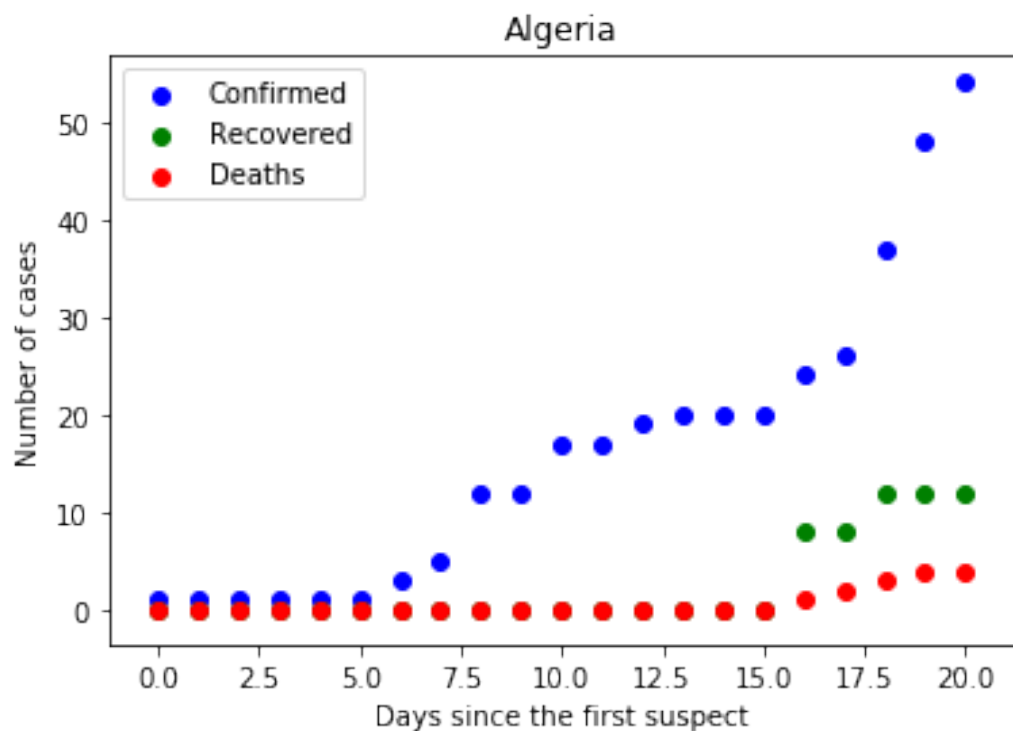
```
    plt.ylabel('Number of cases')
```

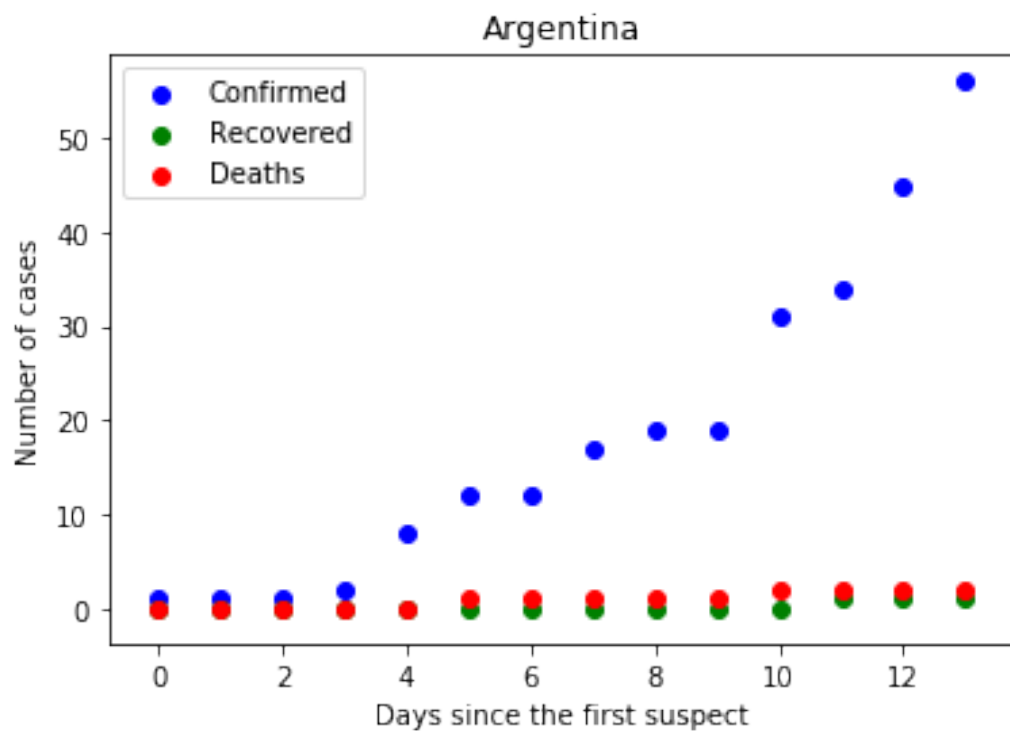
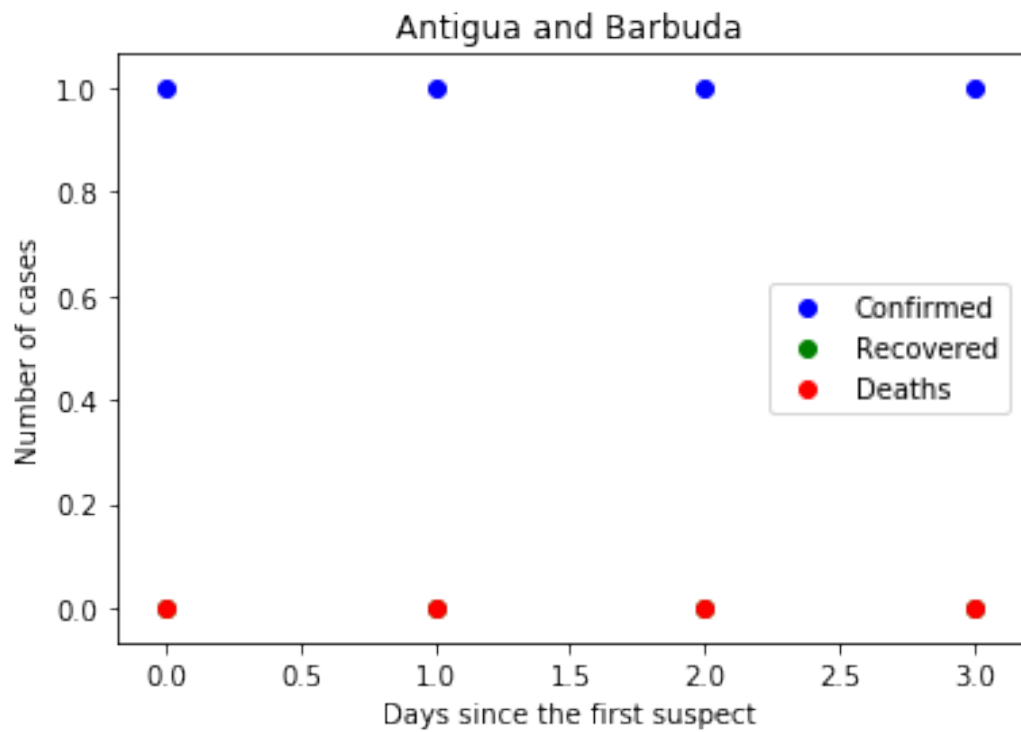
```
    plt.legend()
```

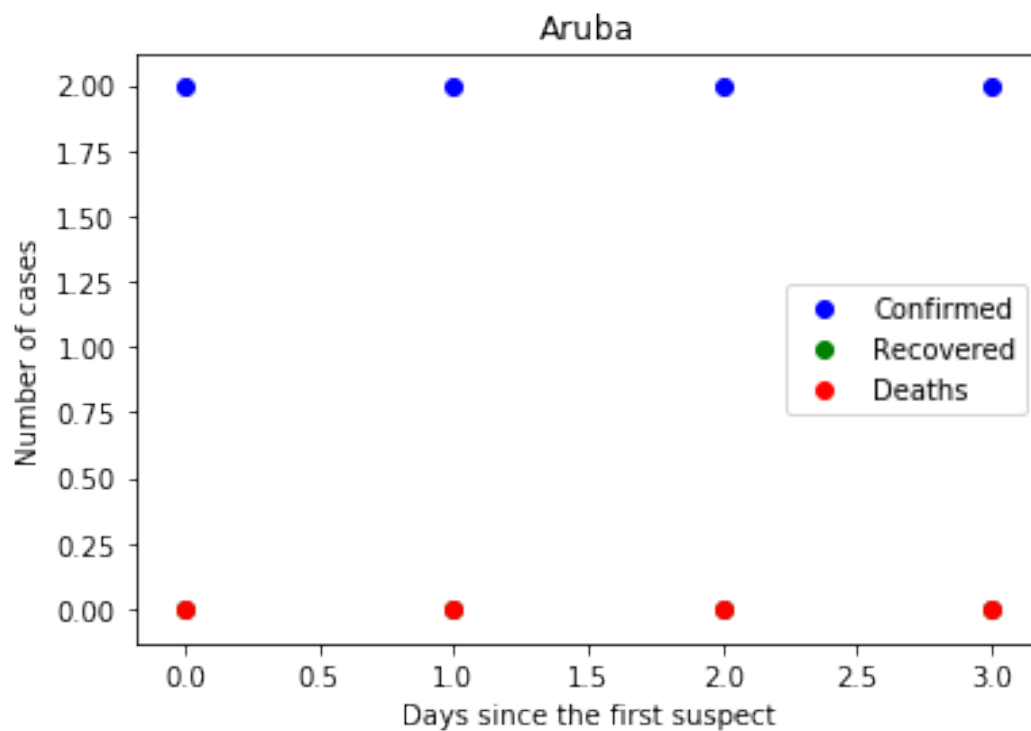
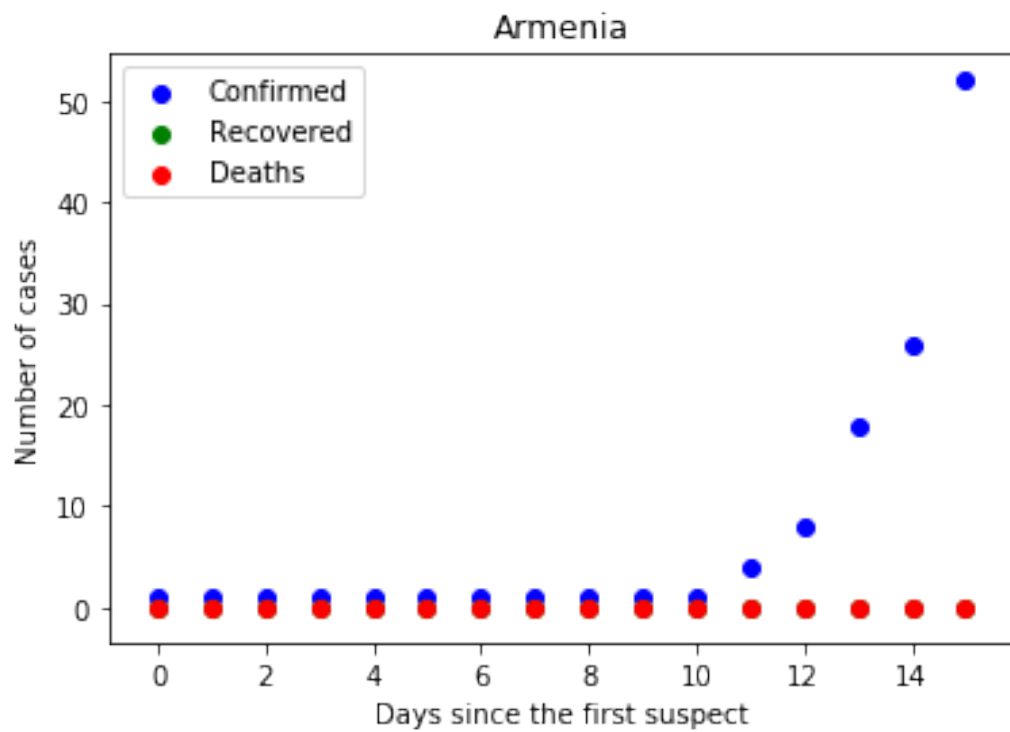
```
    plt.show()
```

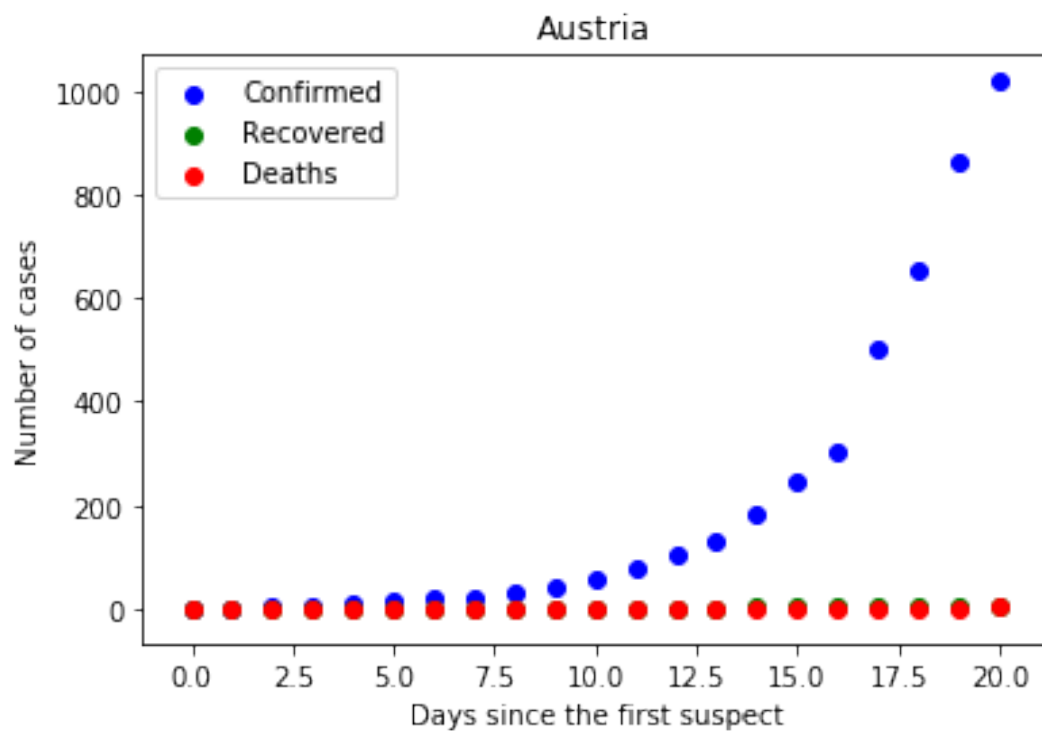
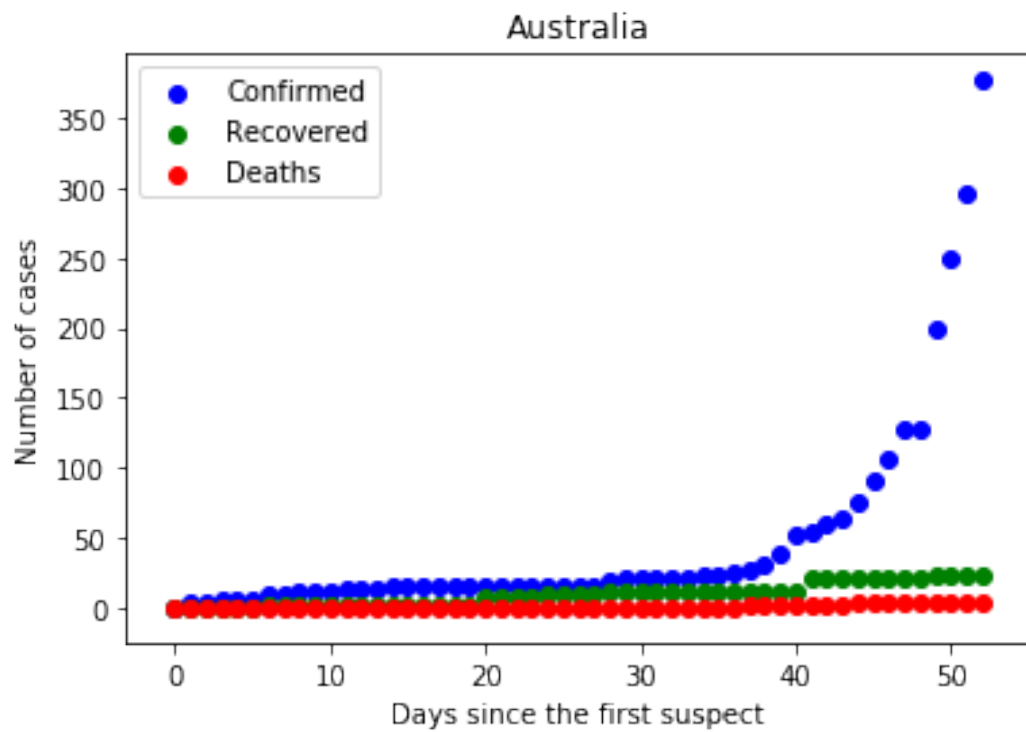




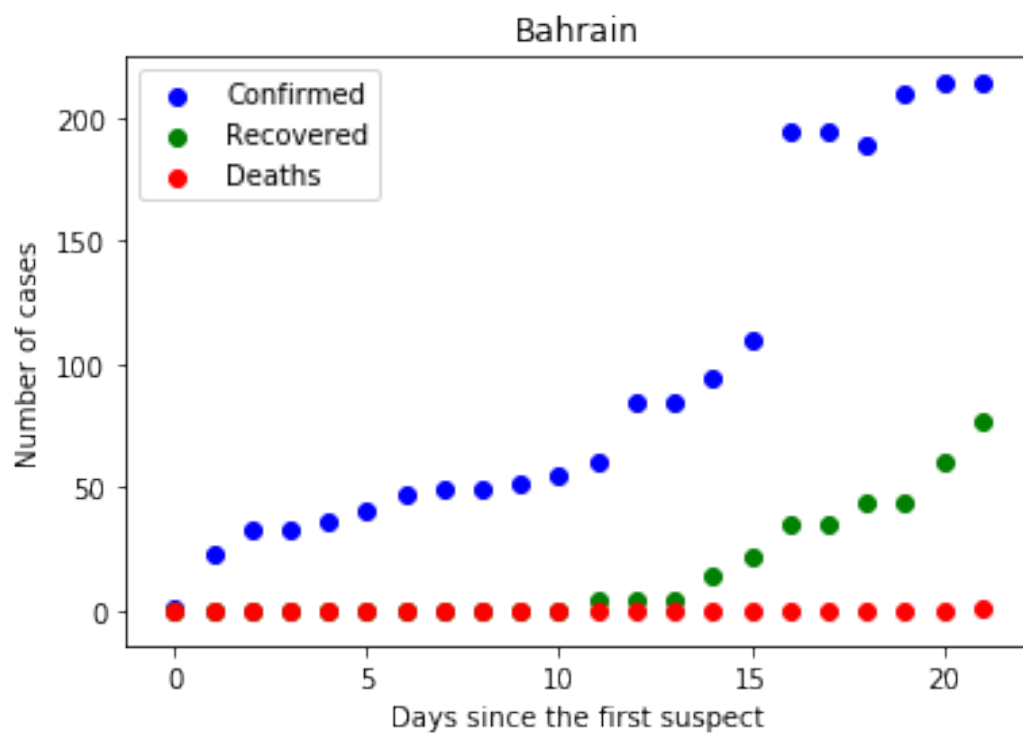
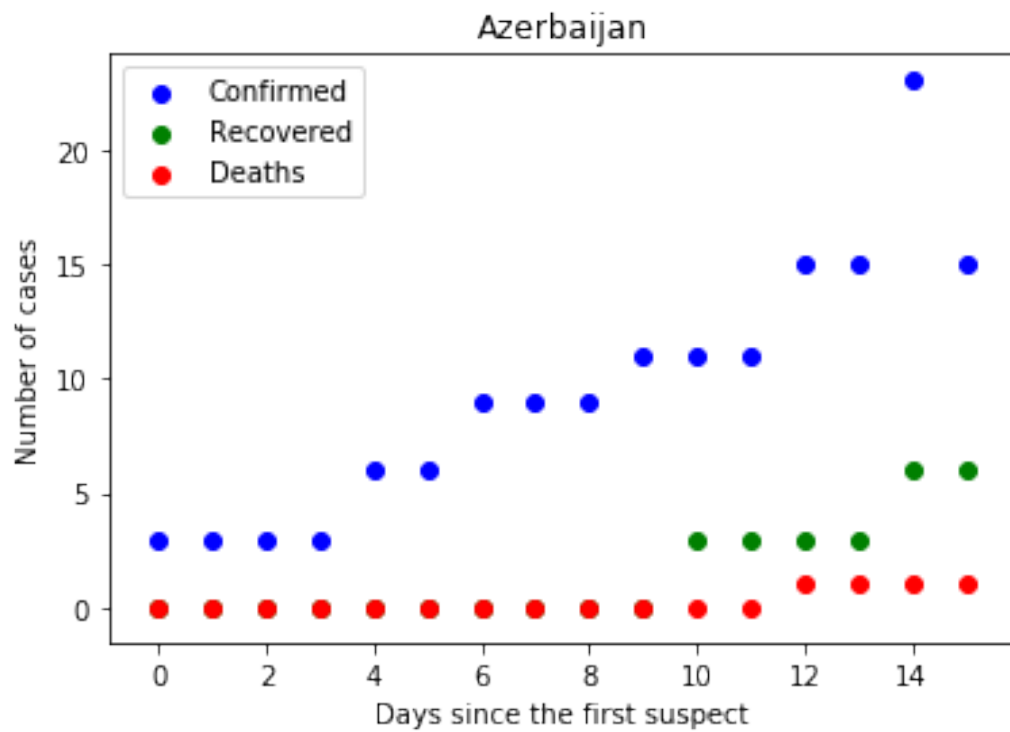


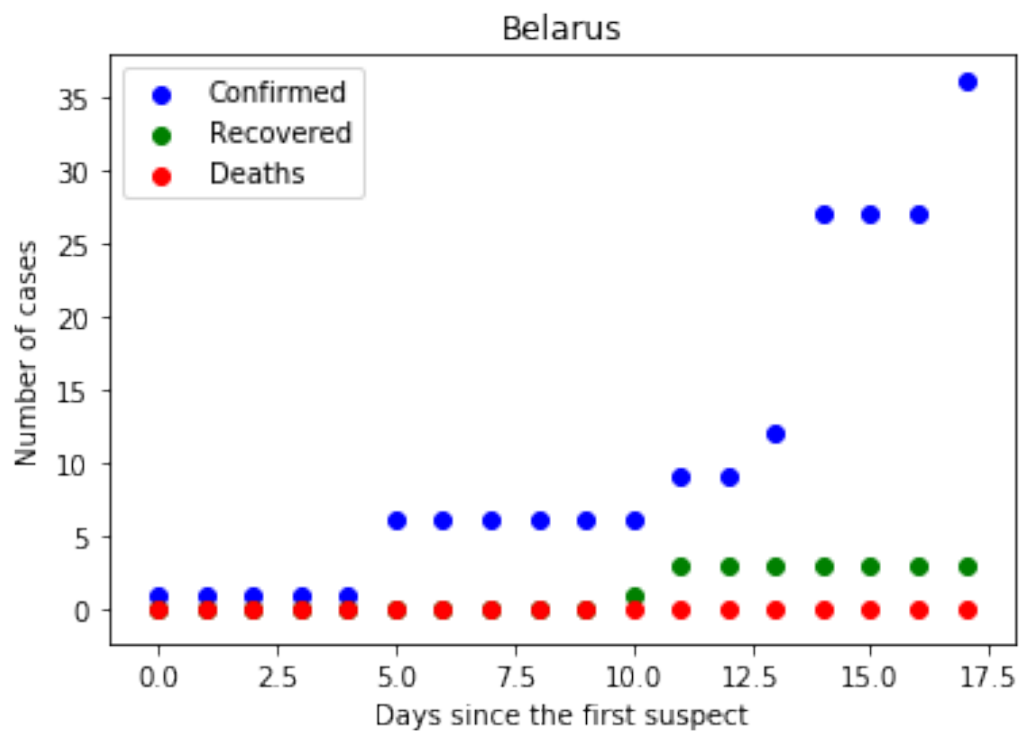
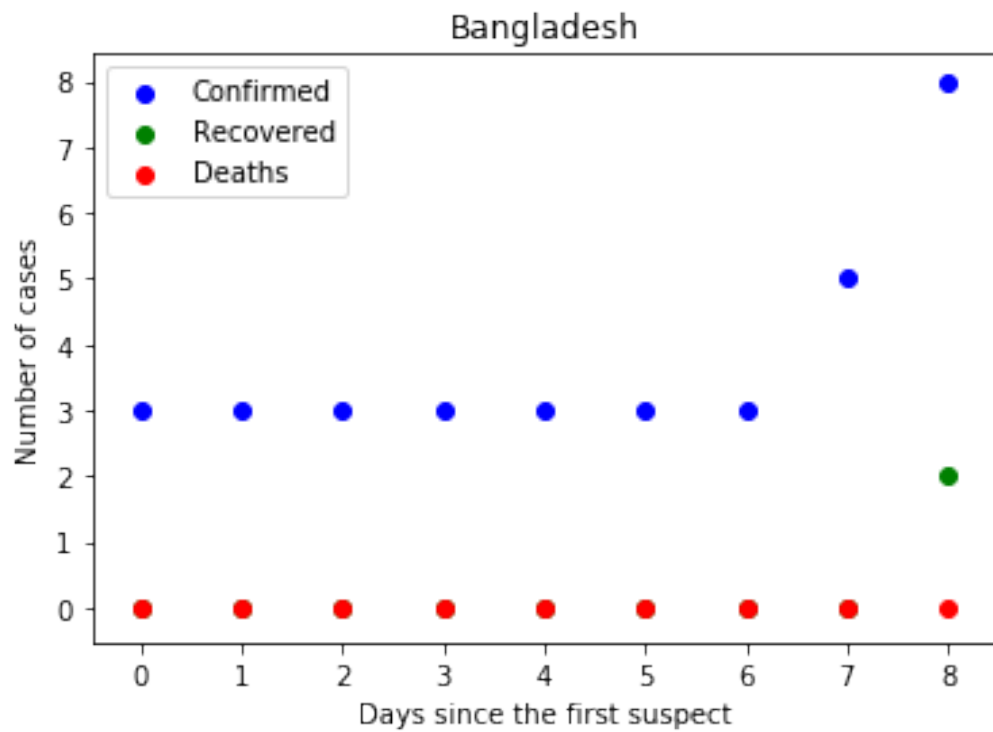


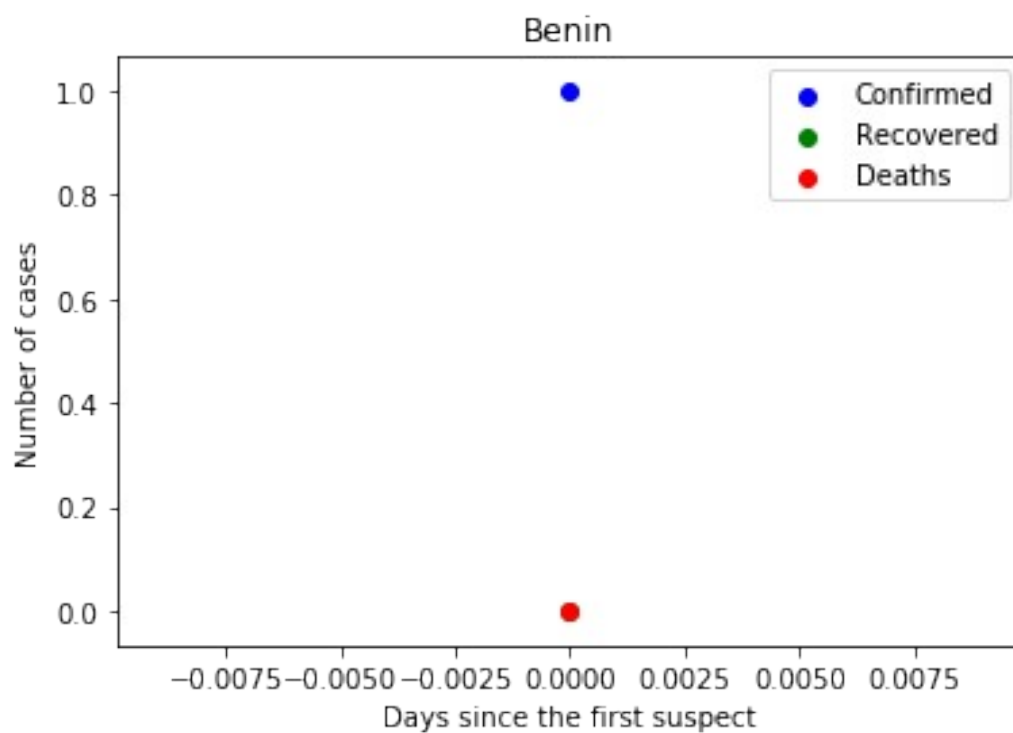
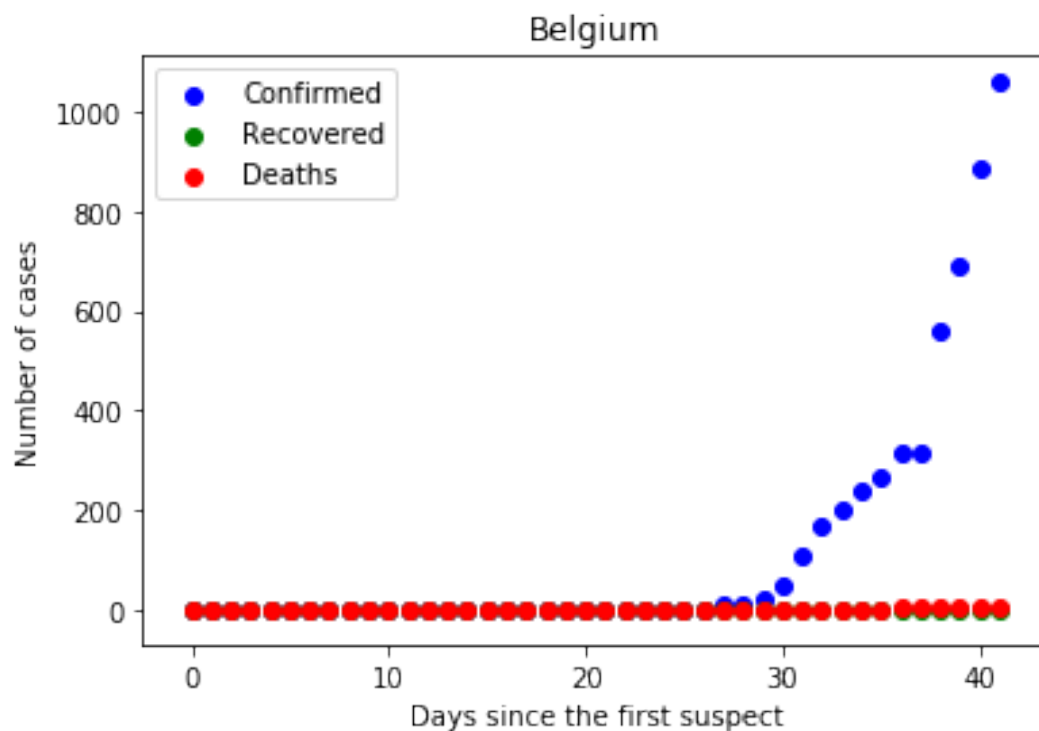


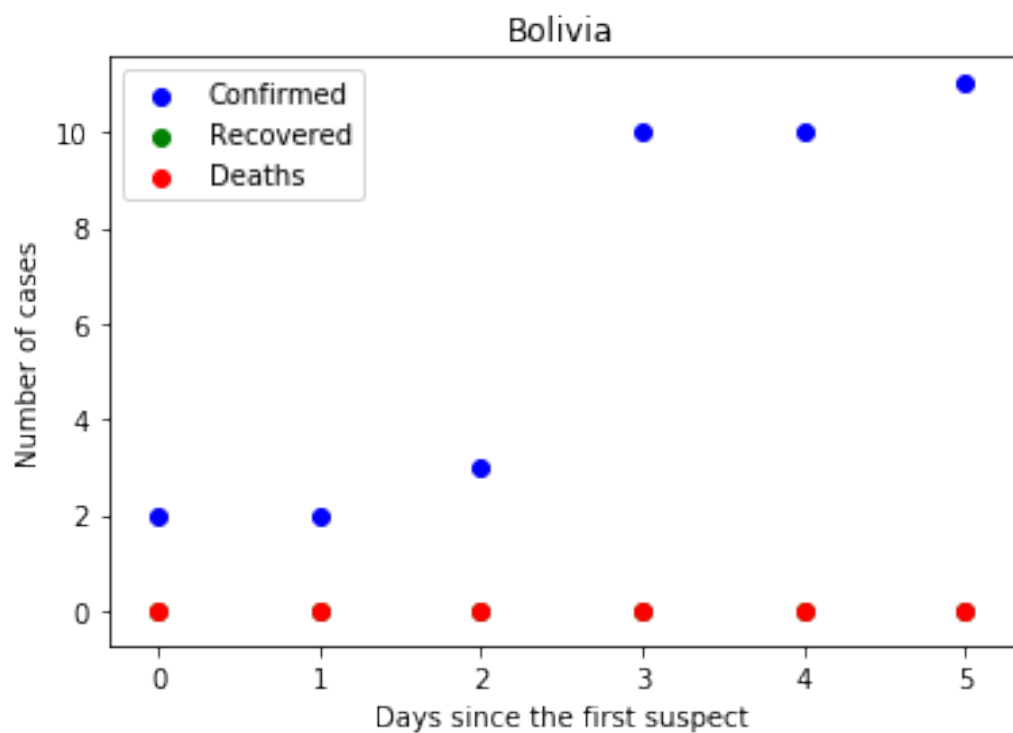
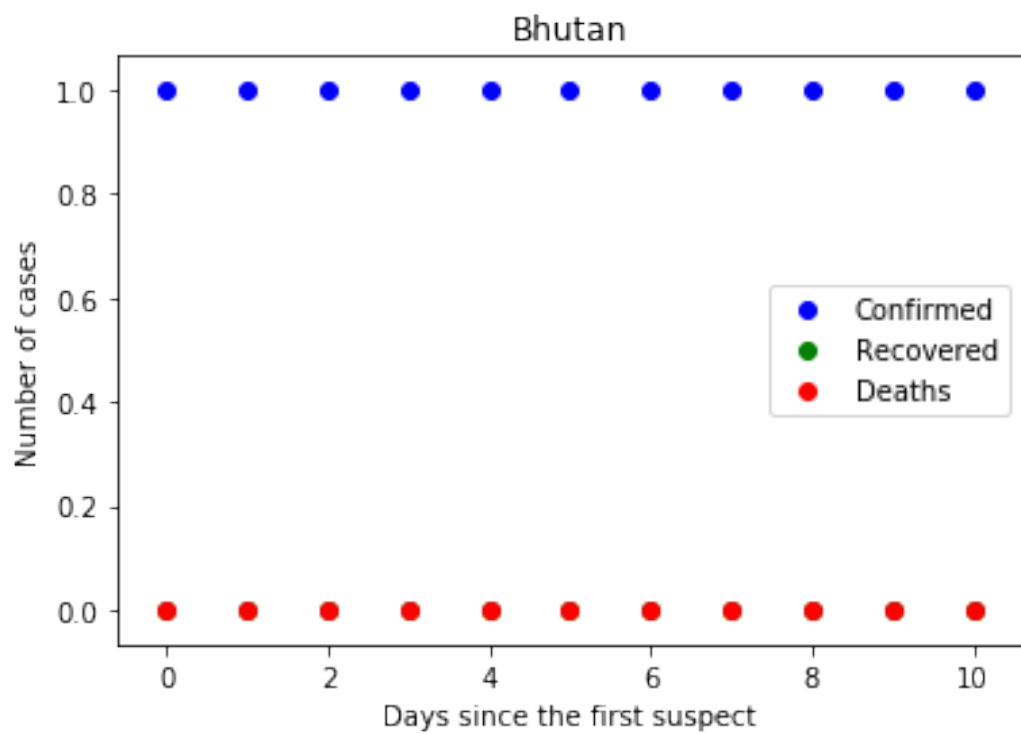


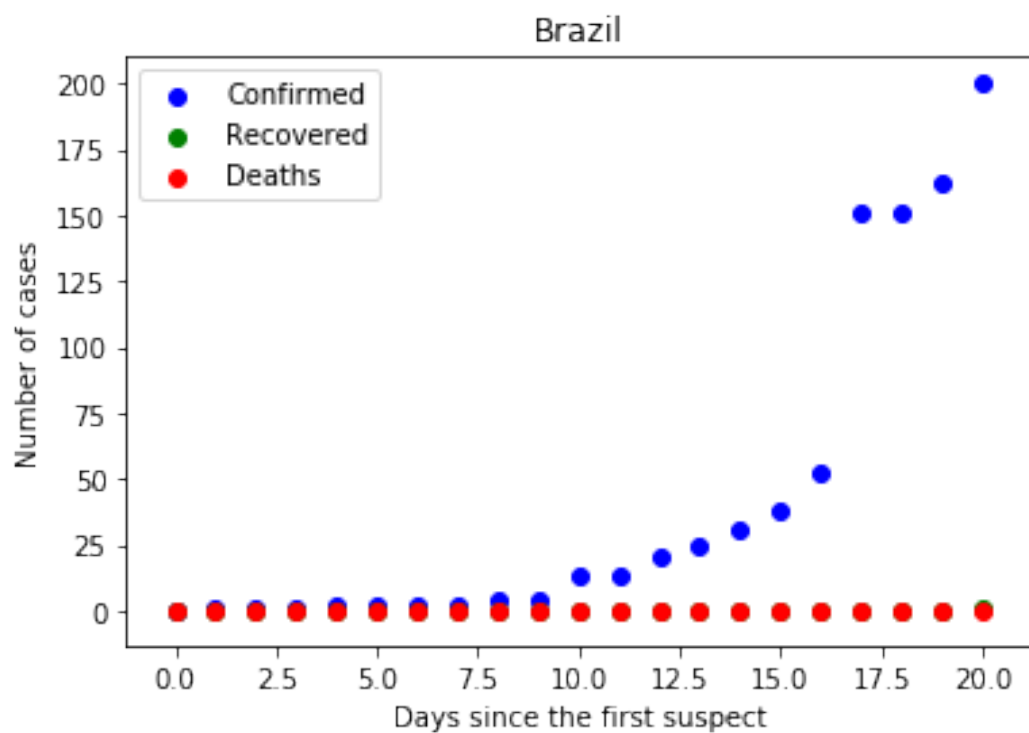
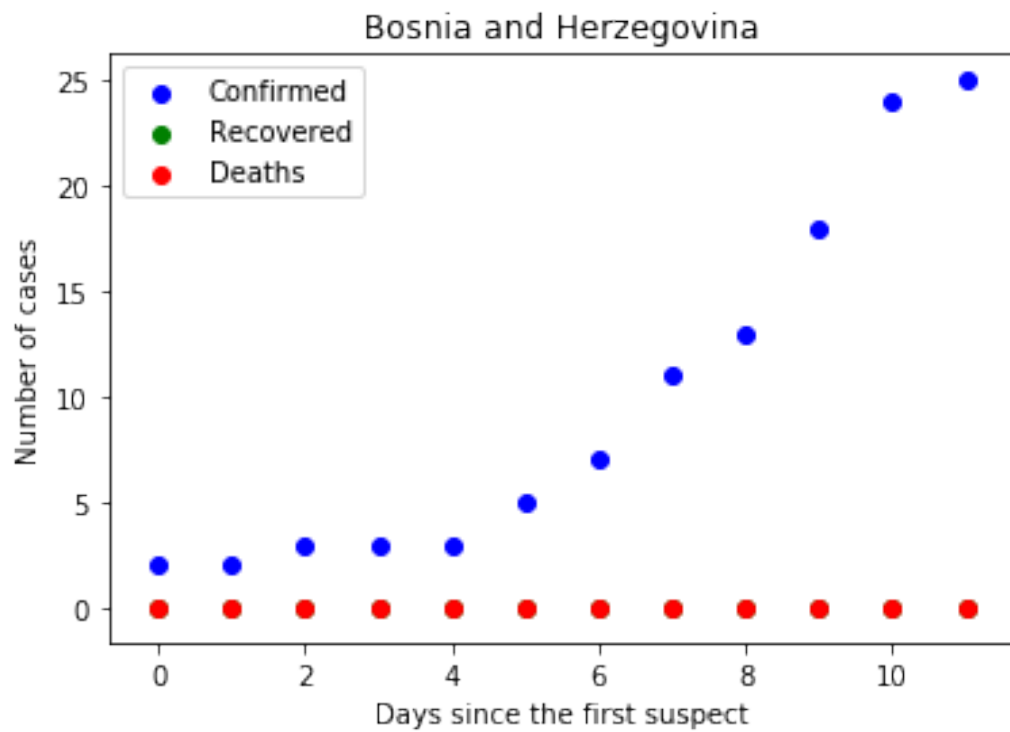


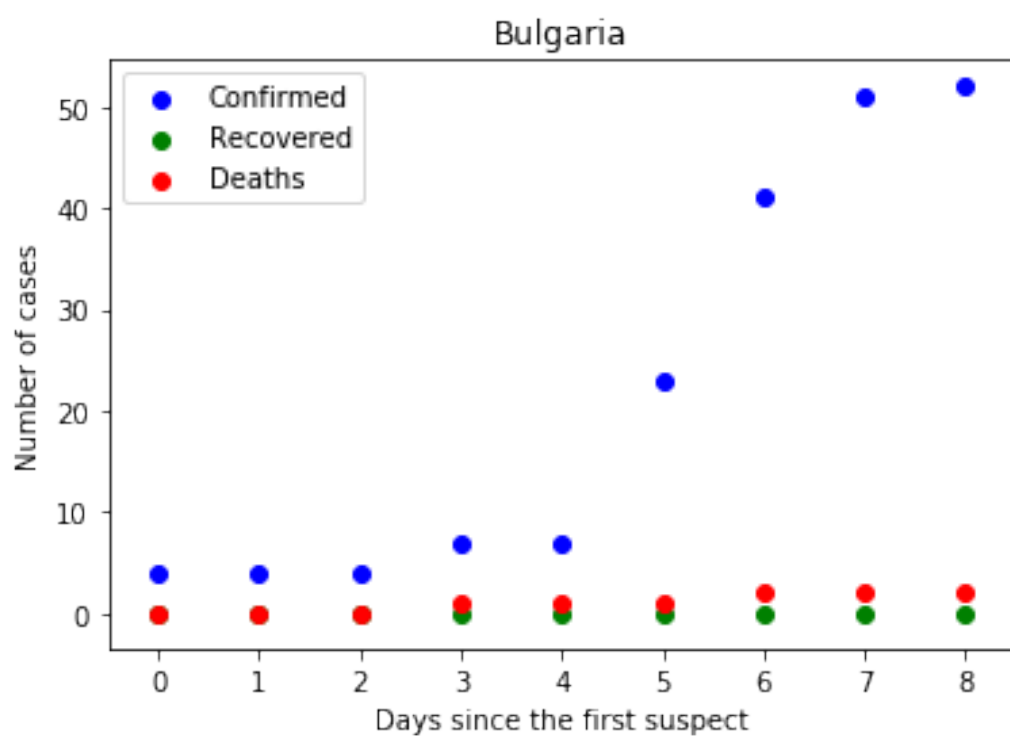
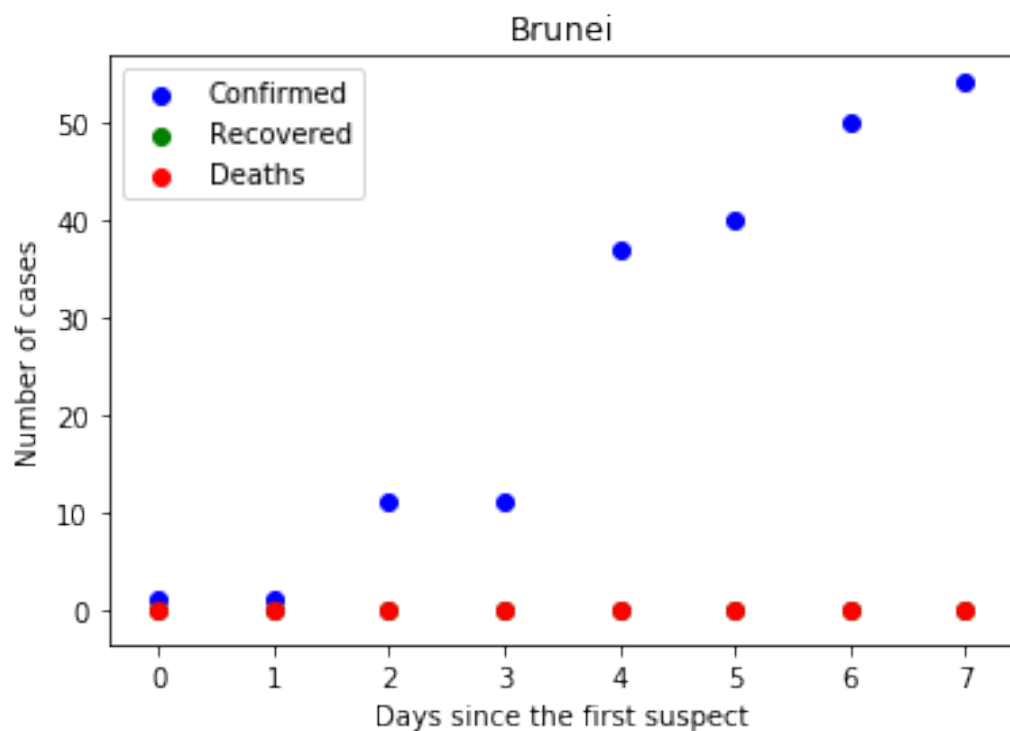


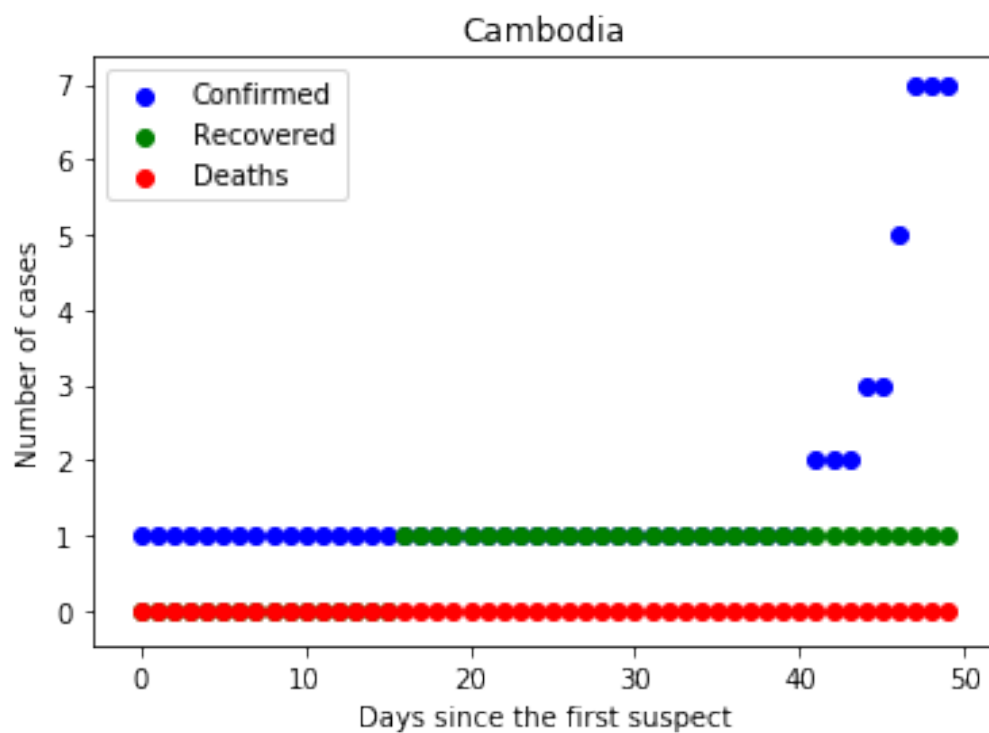
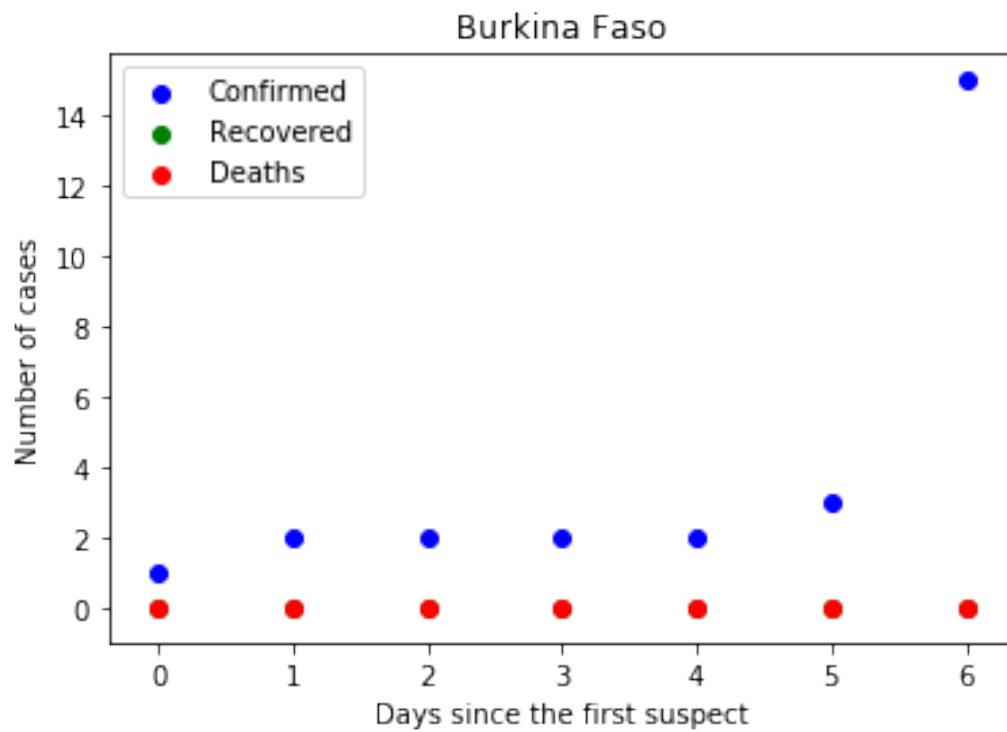


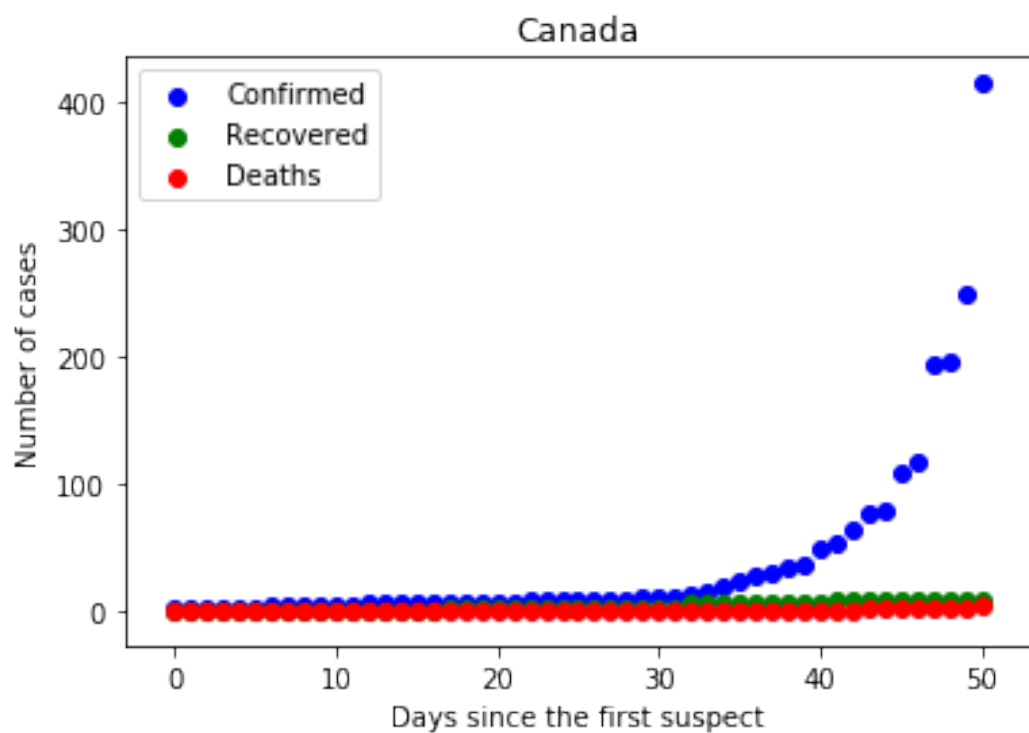
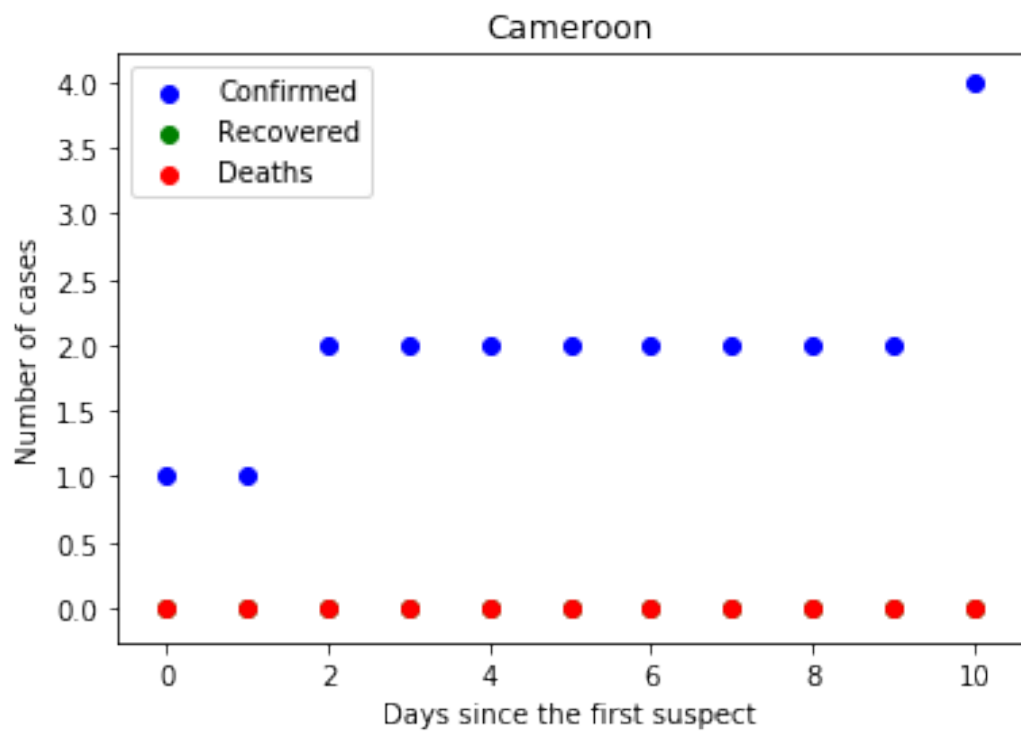




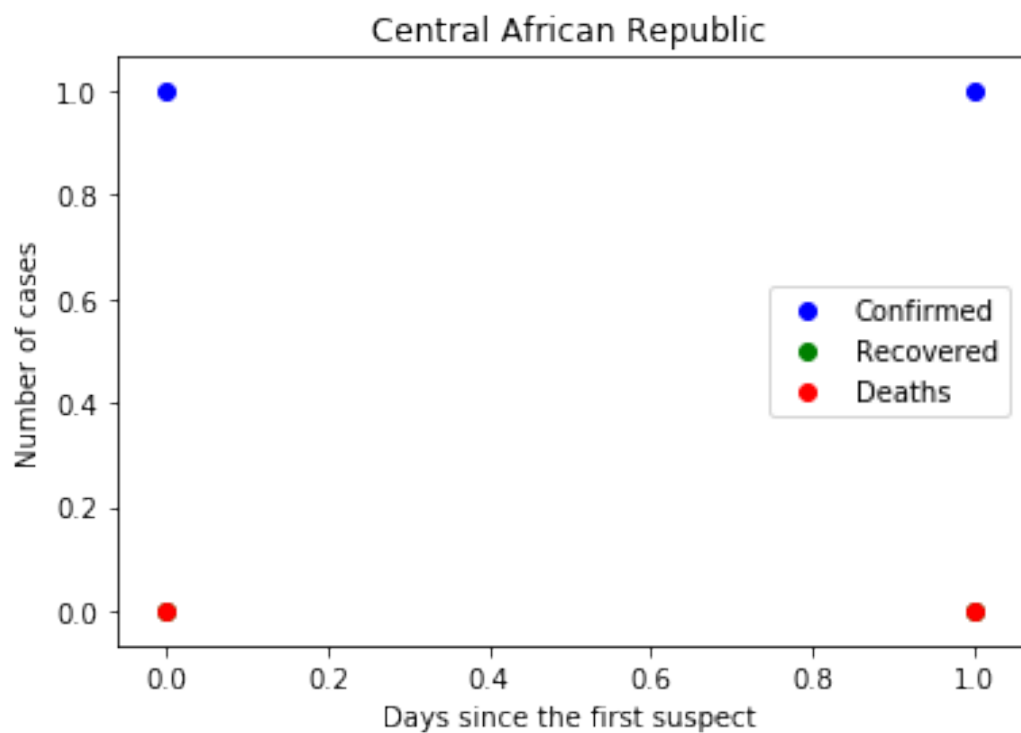
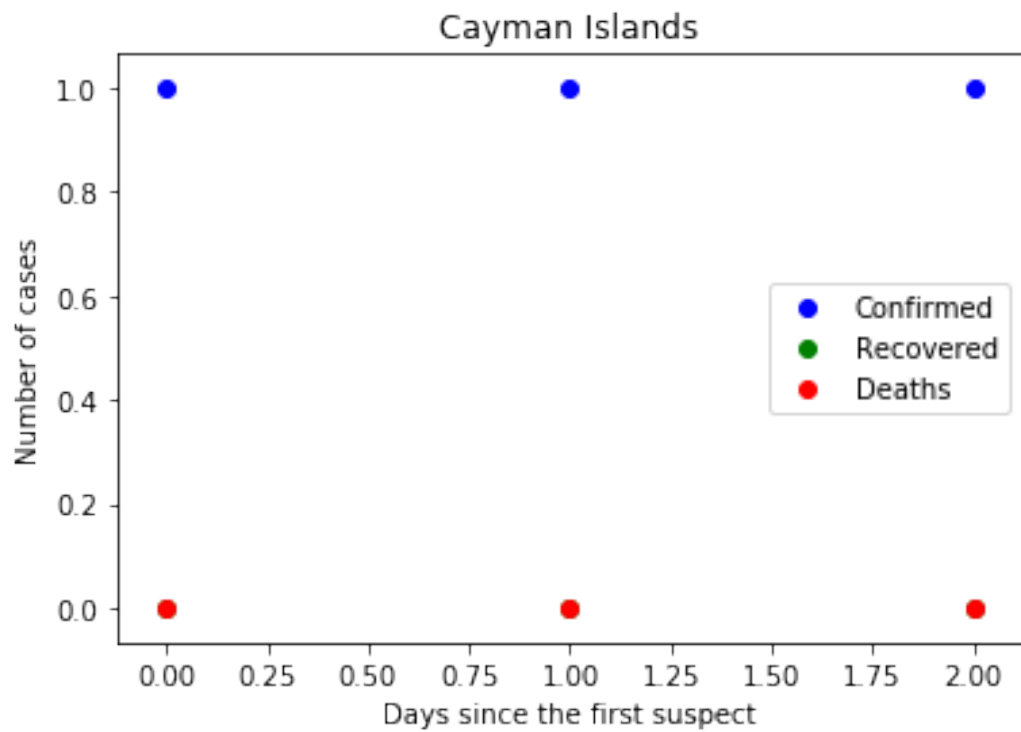


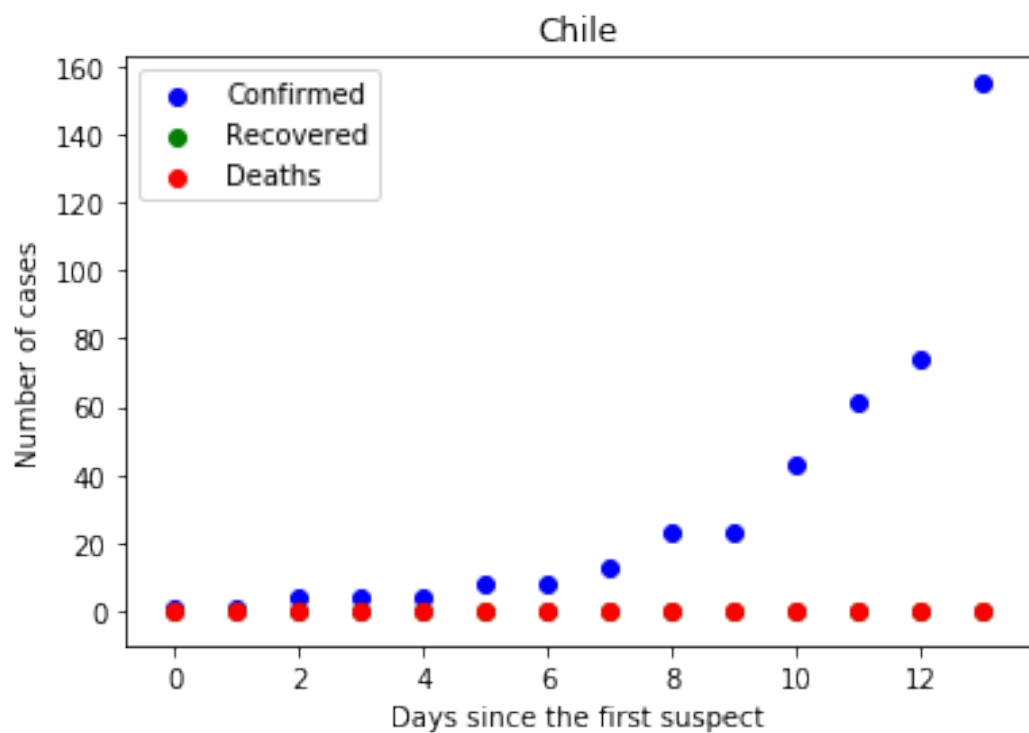
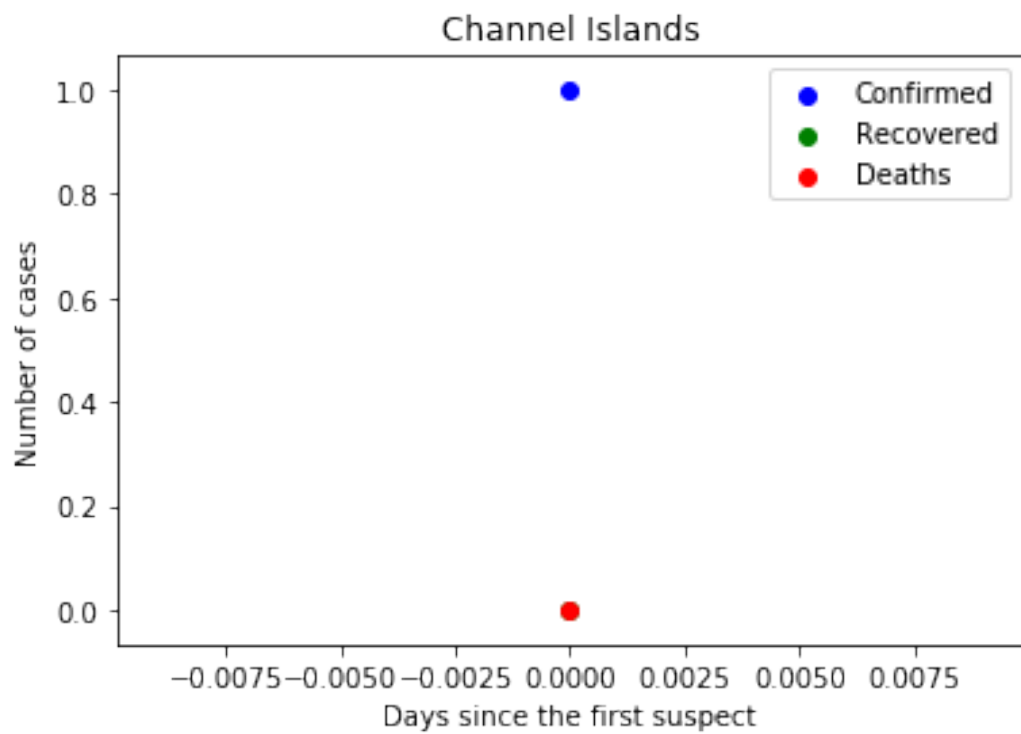


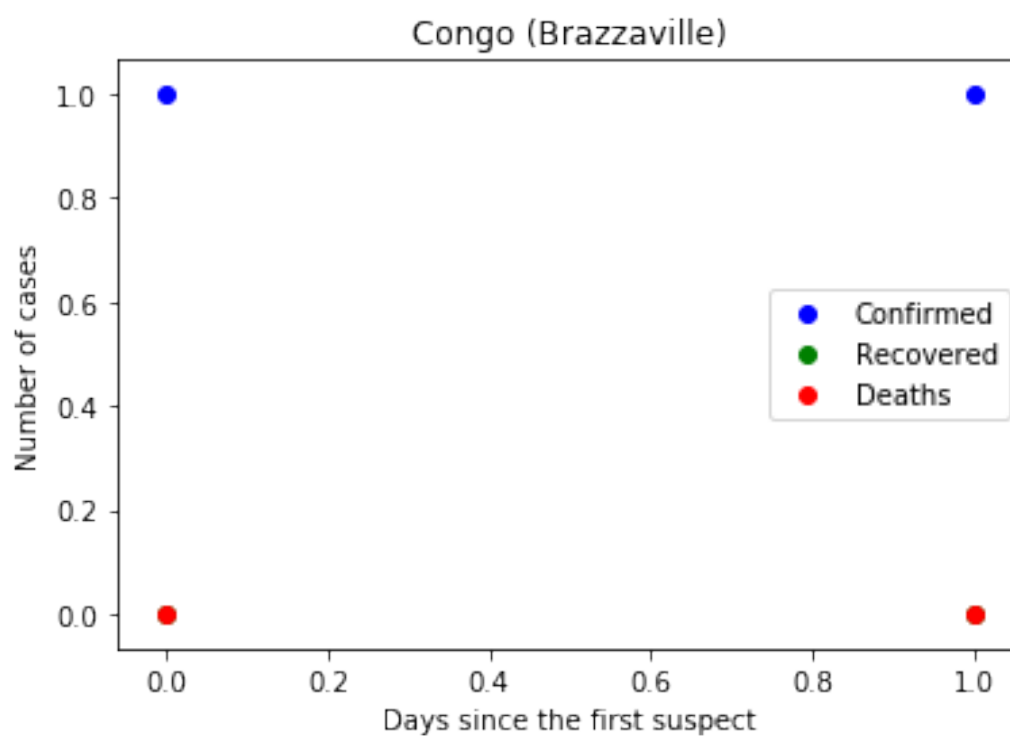
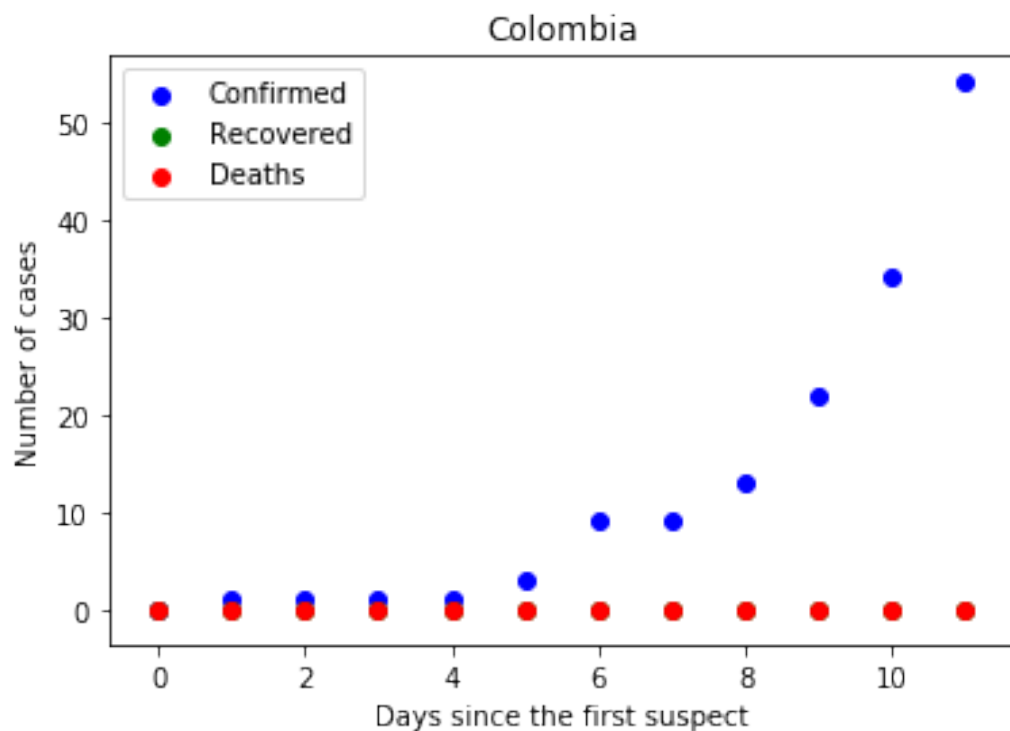


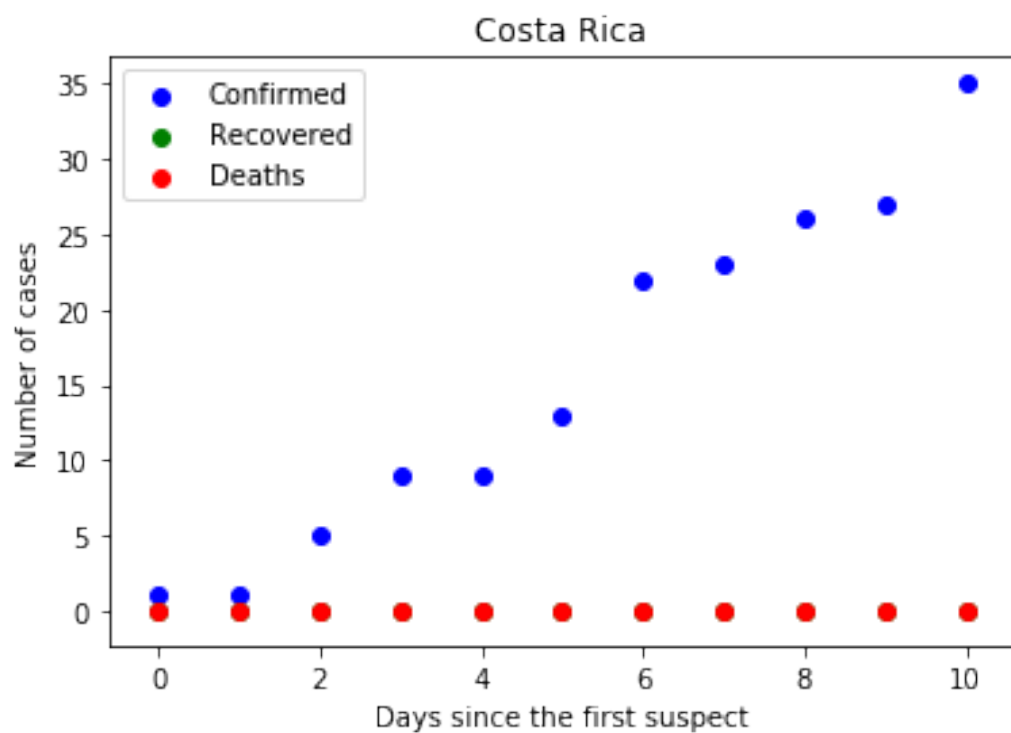
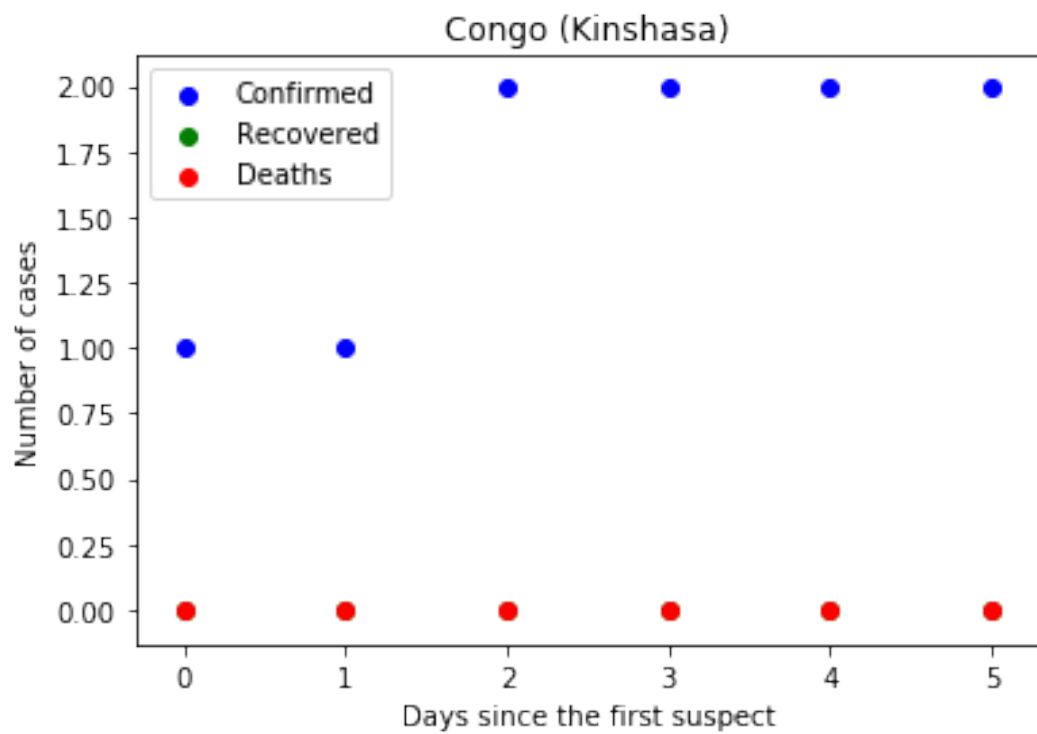


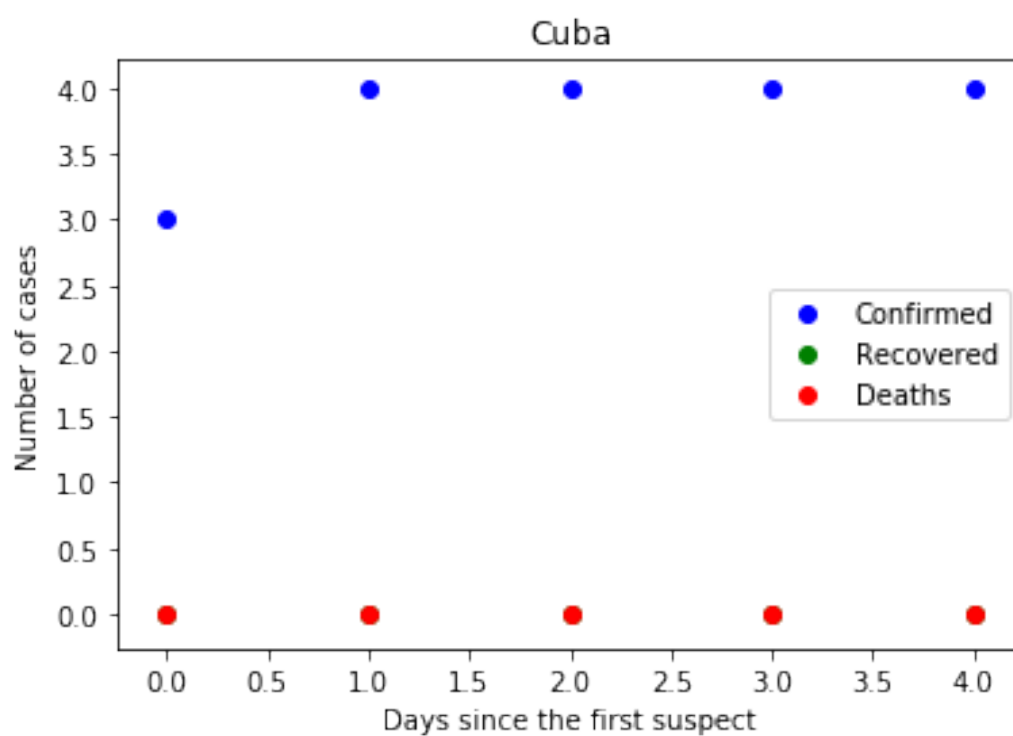
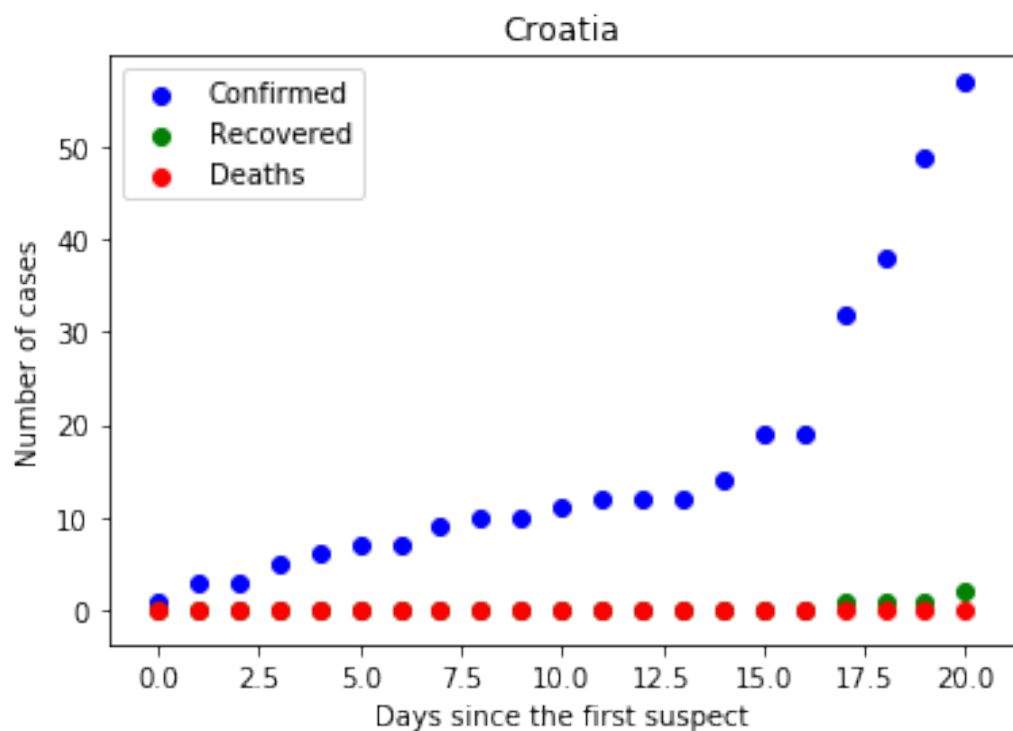


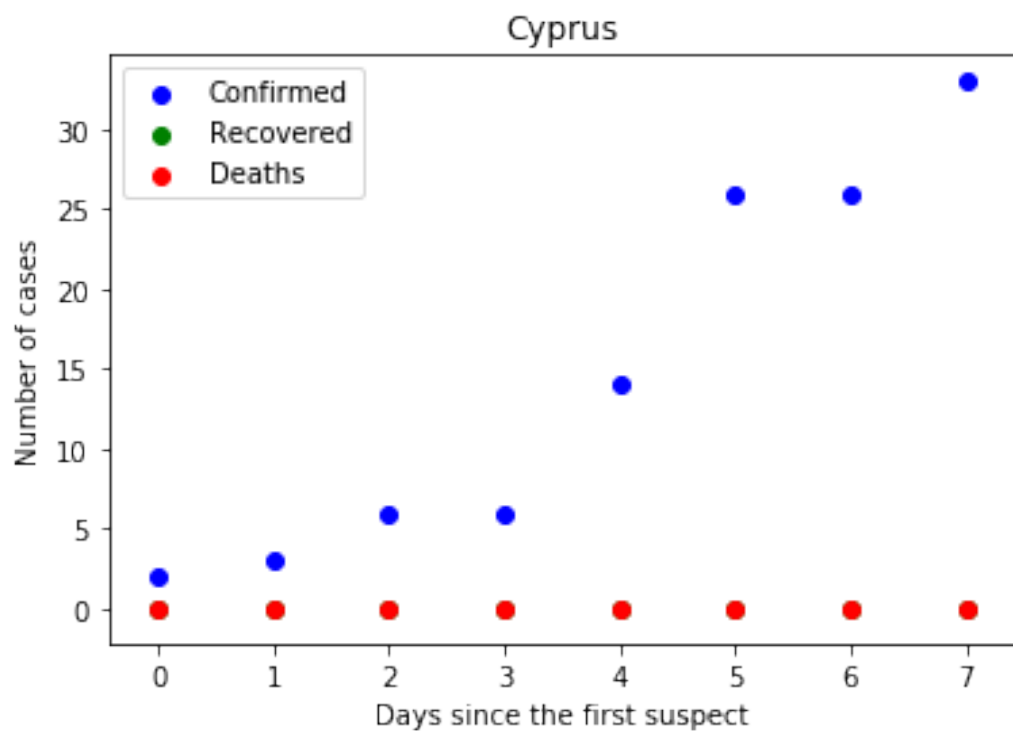
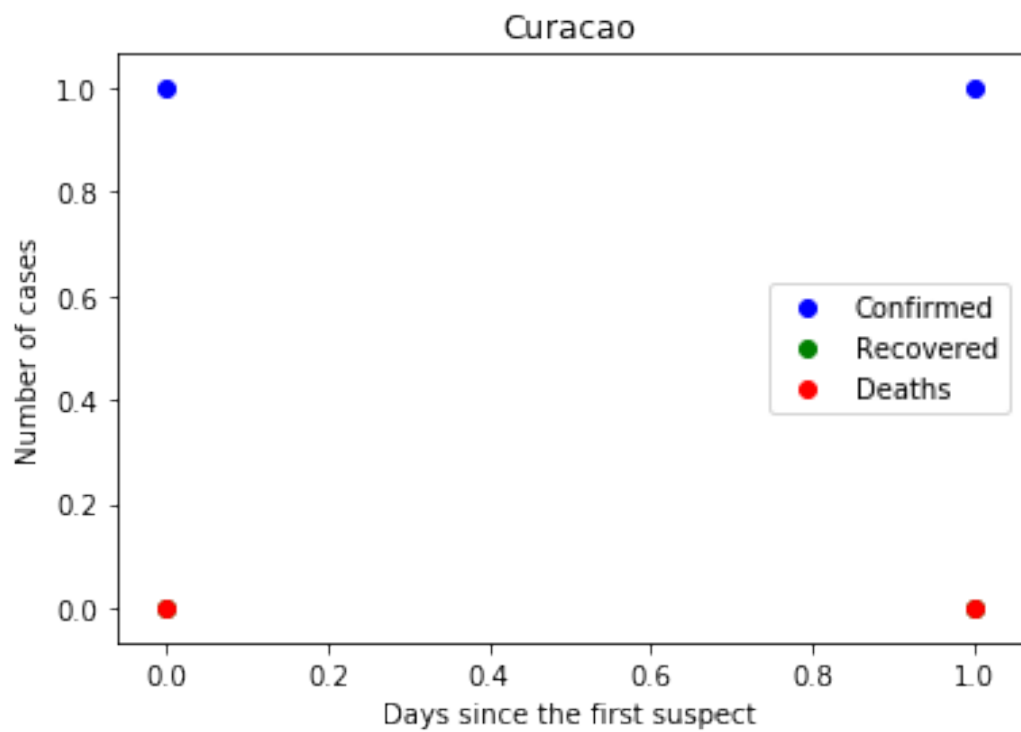


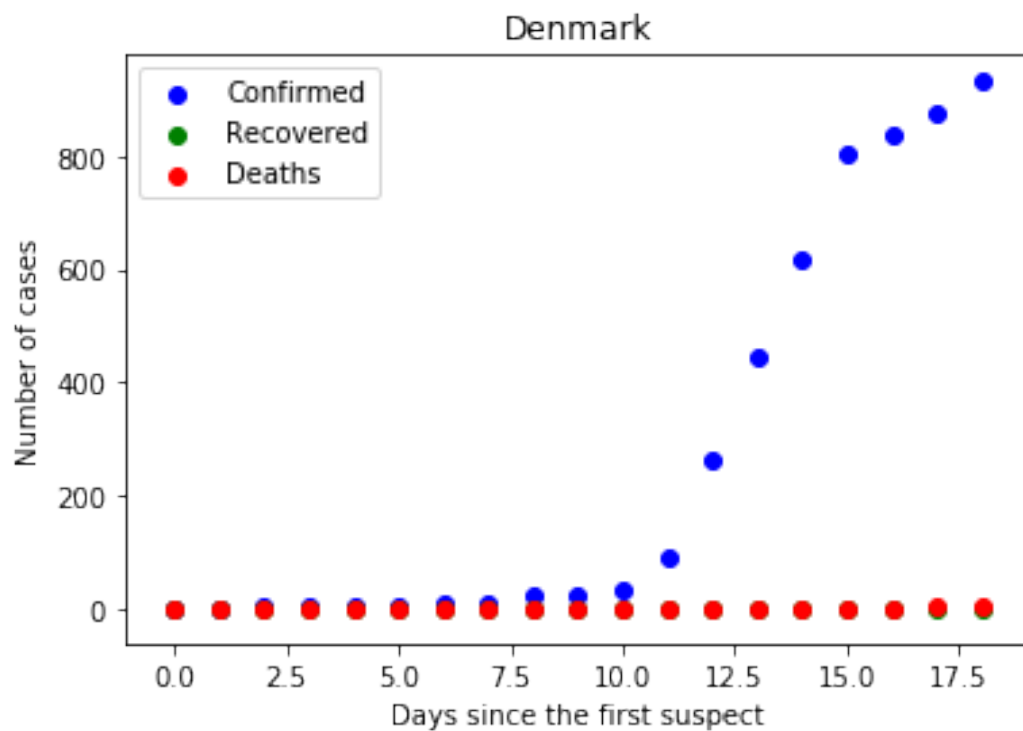
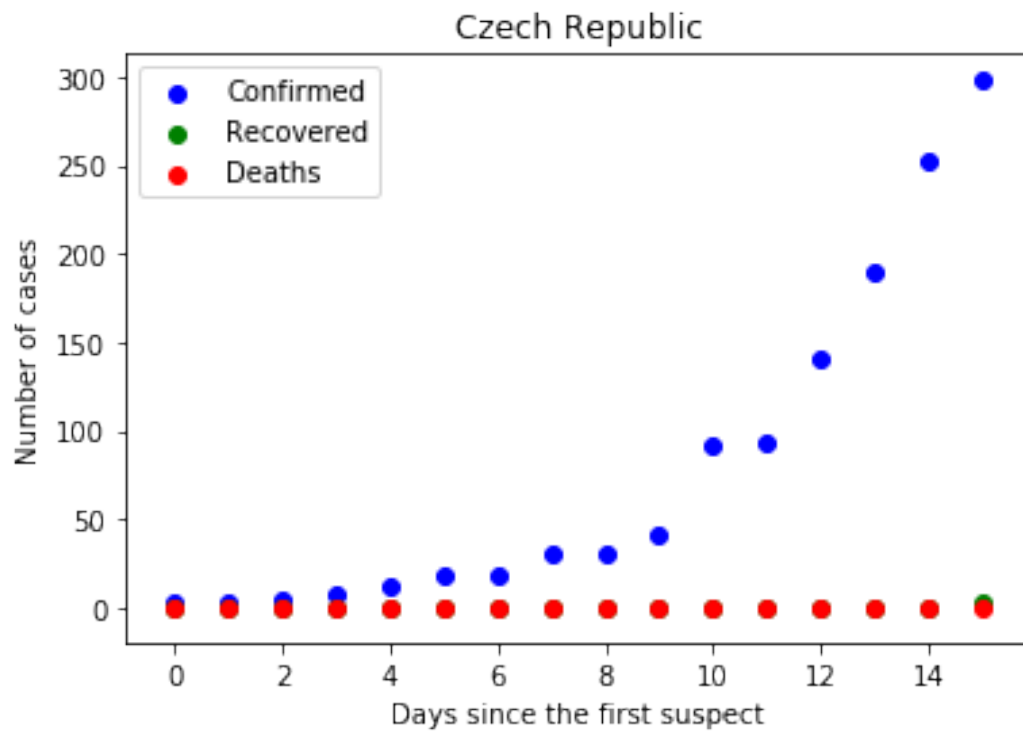


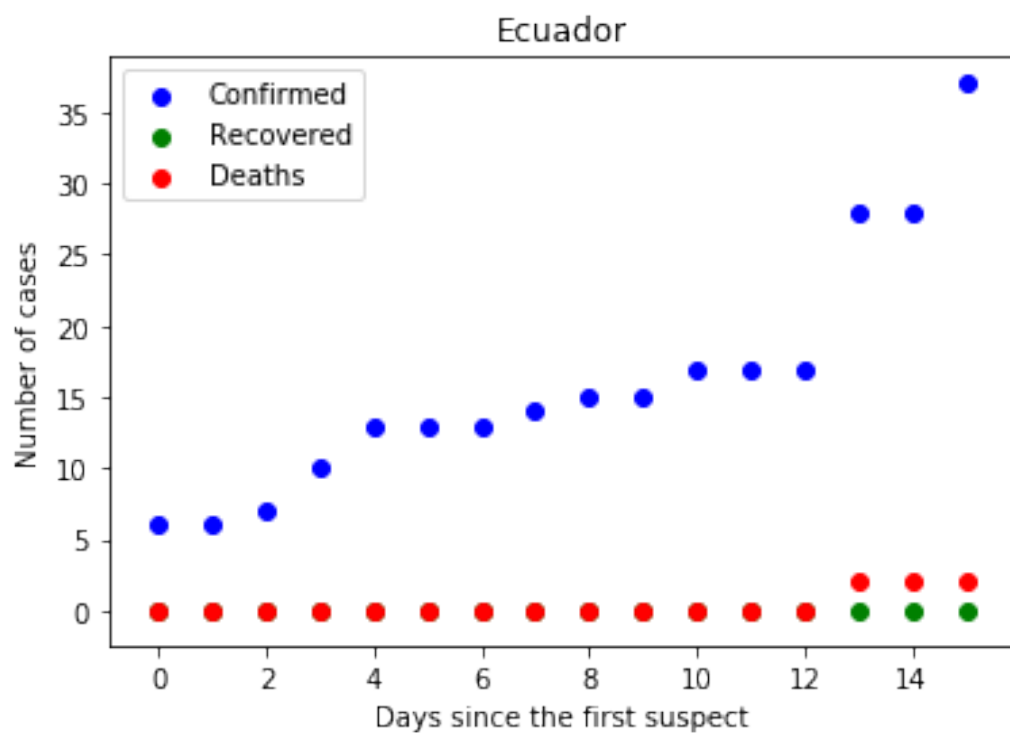
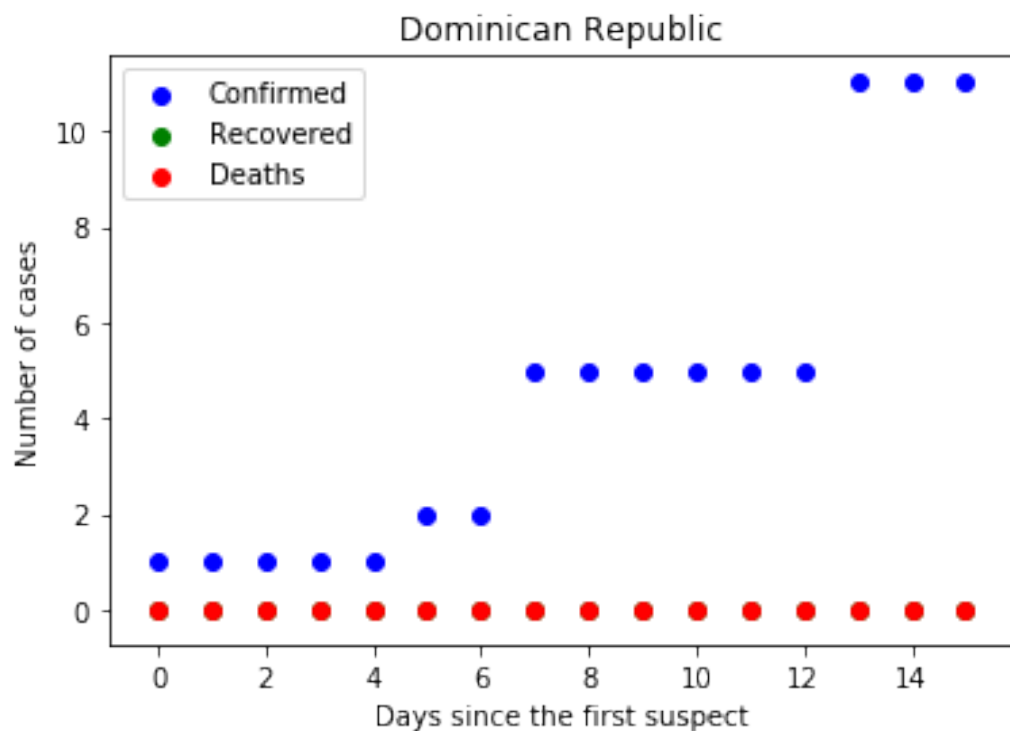




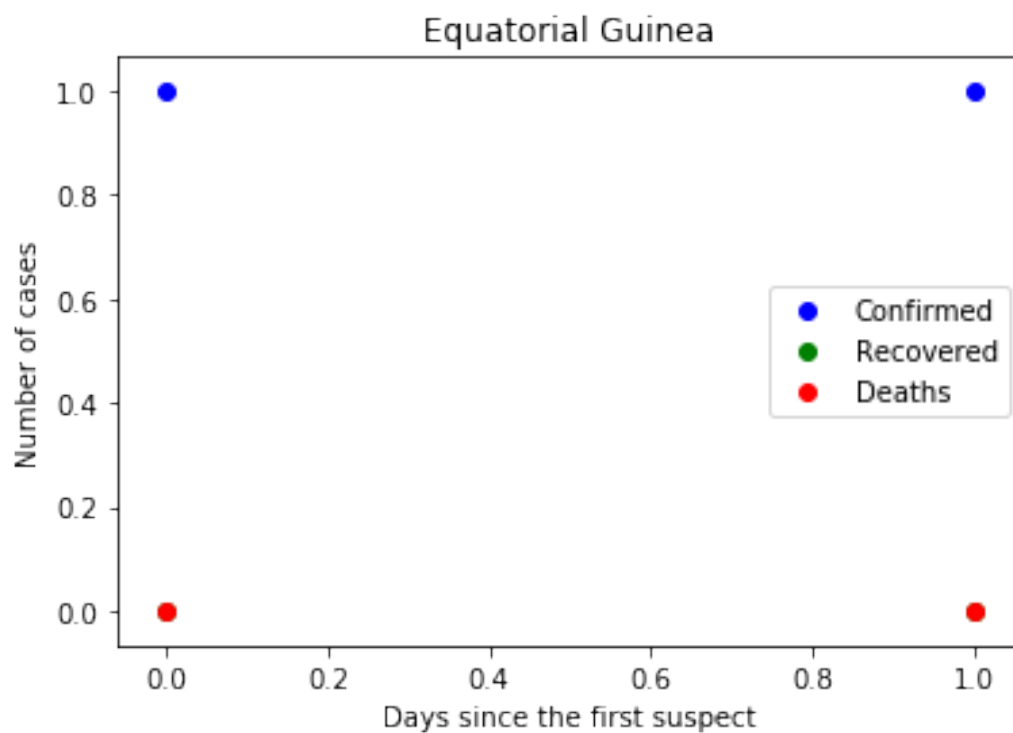
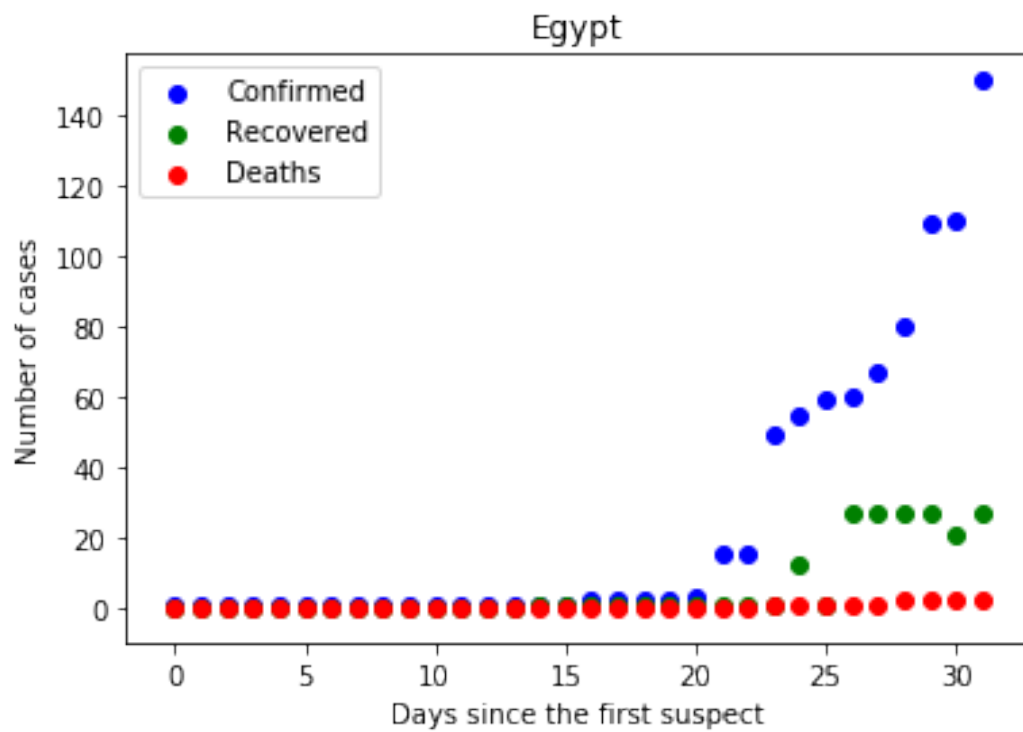


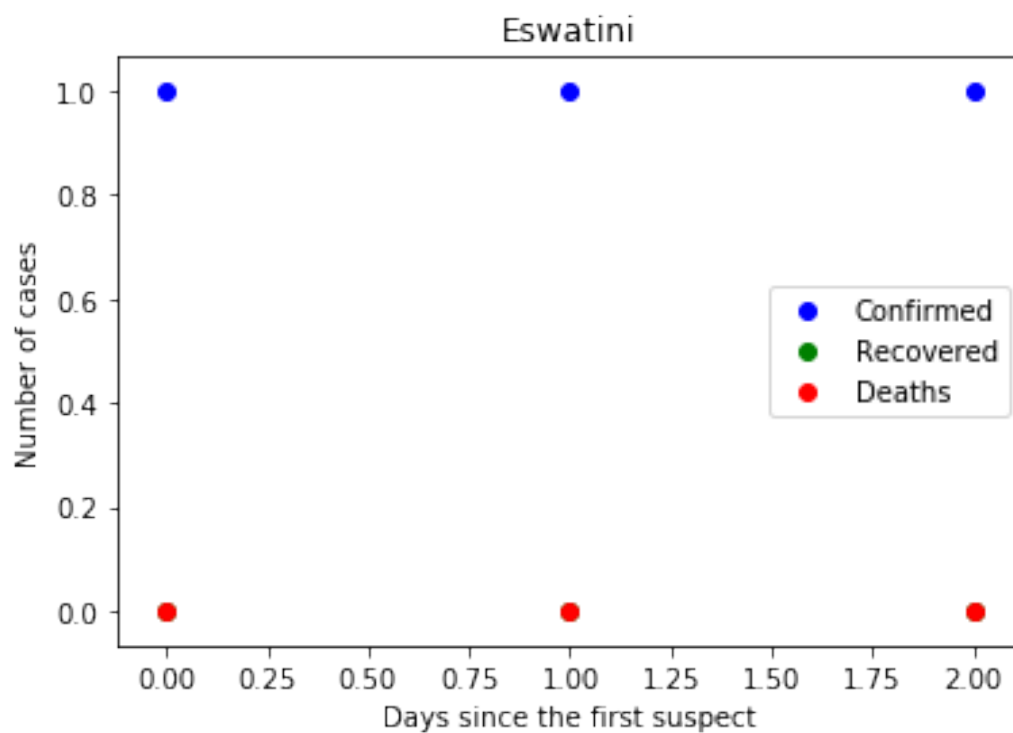
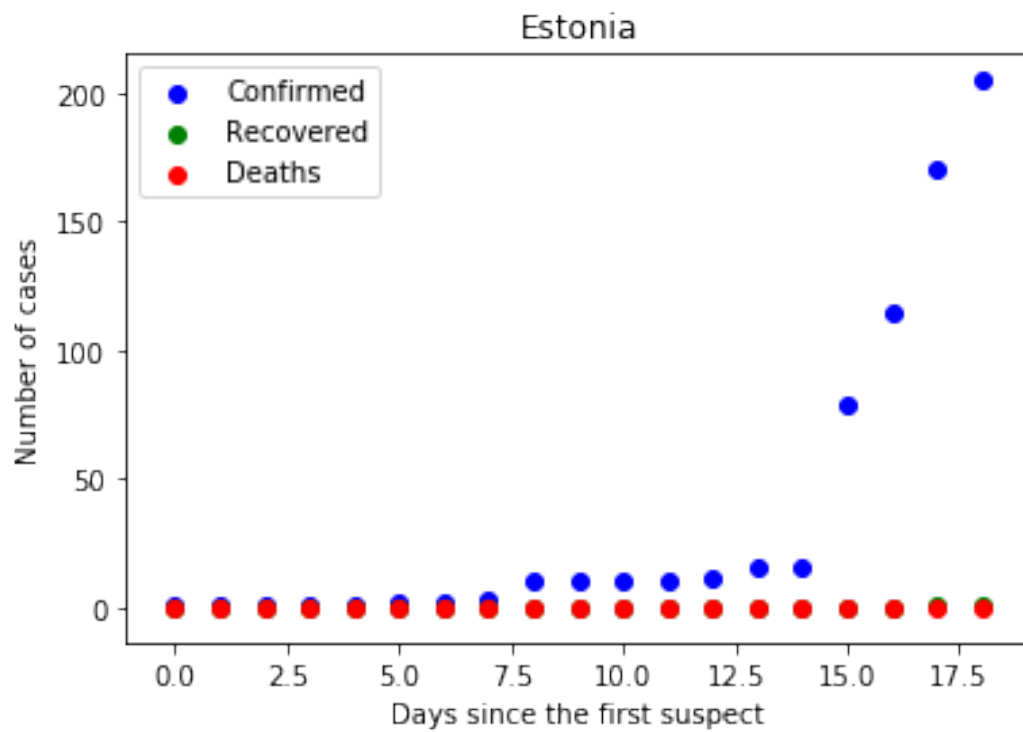


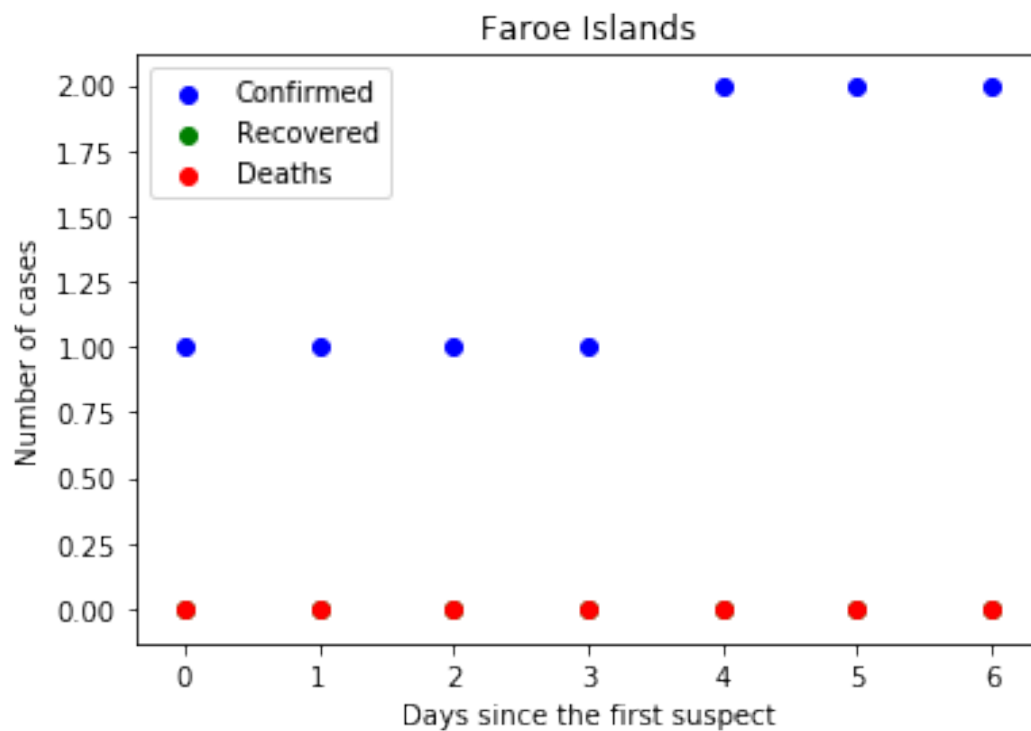
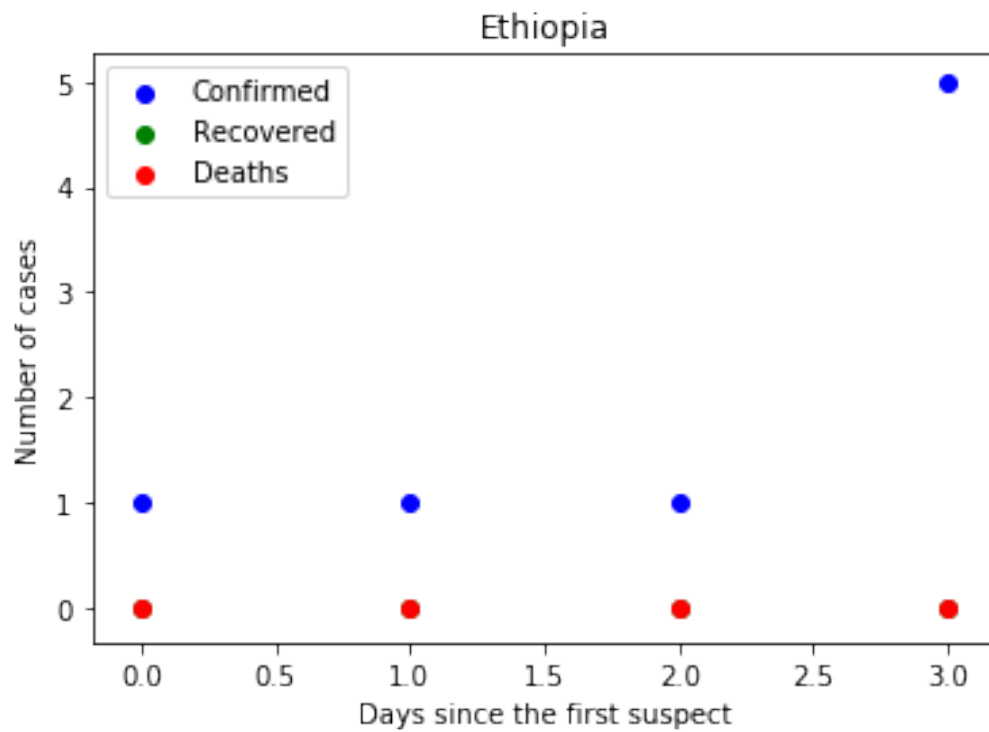


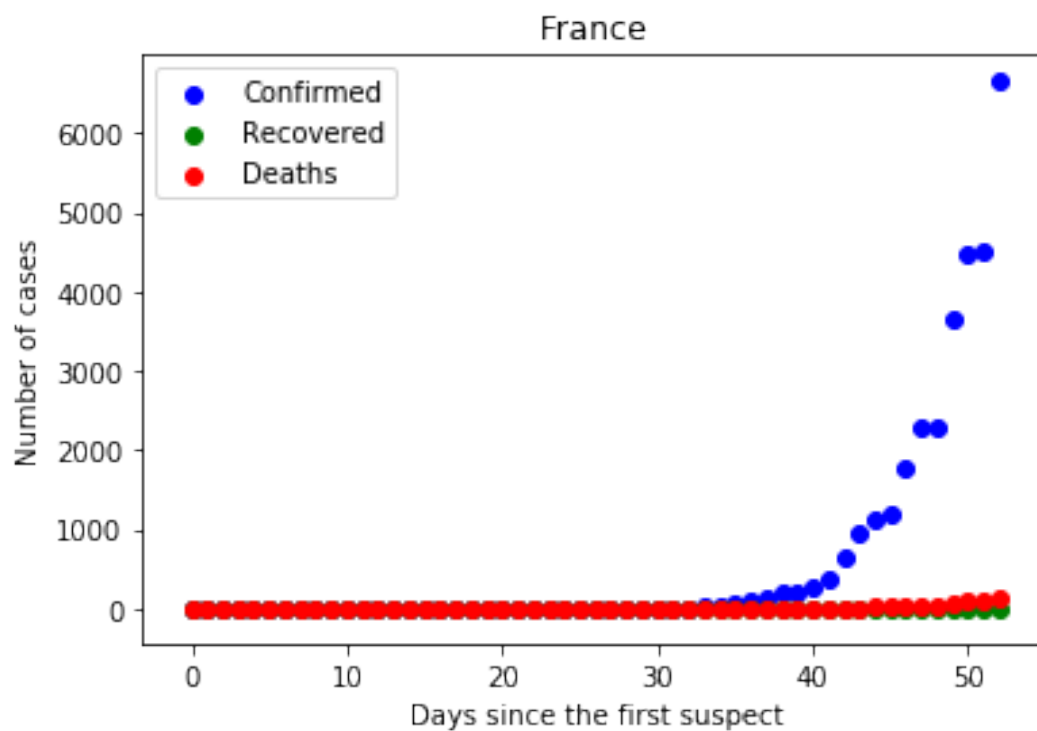
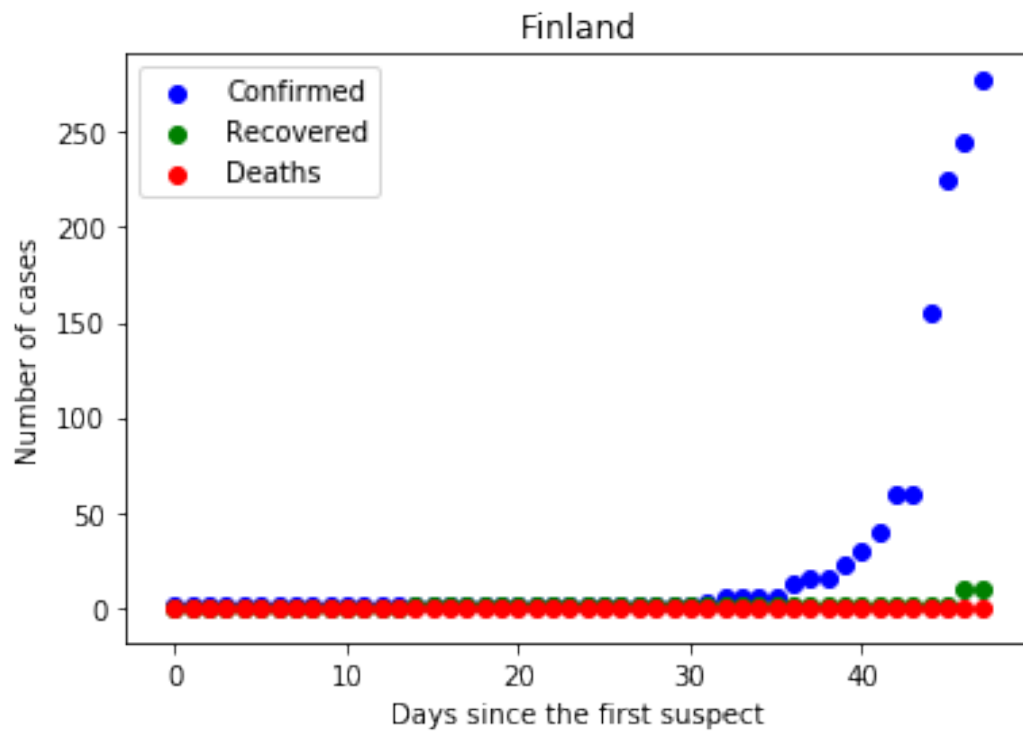


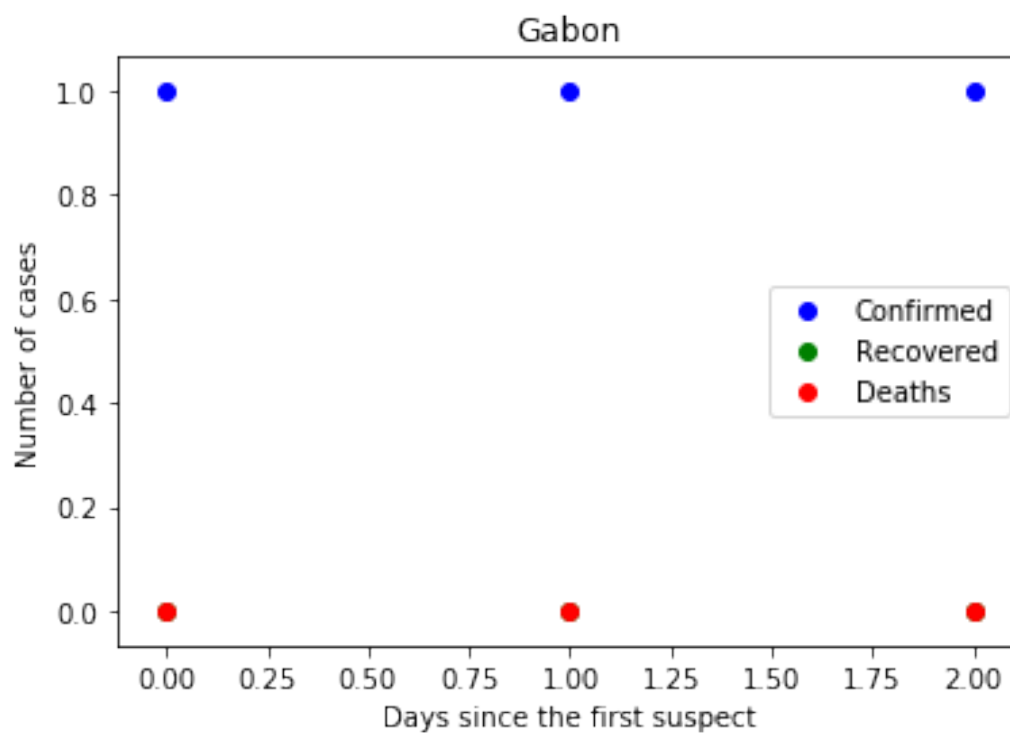
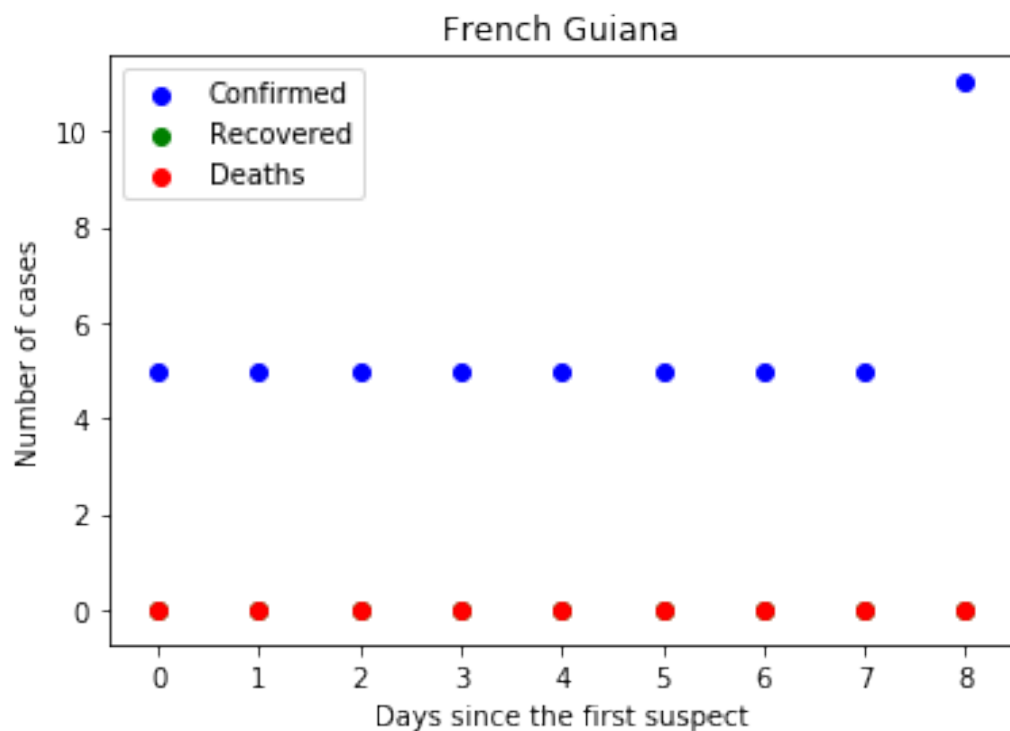


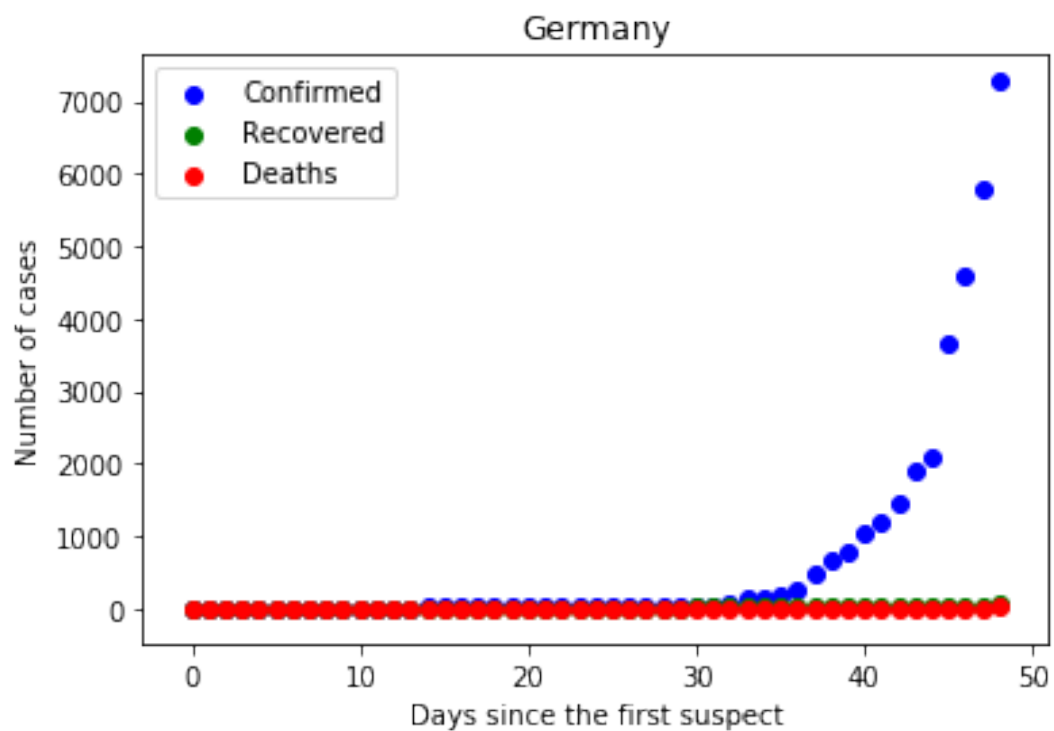
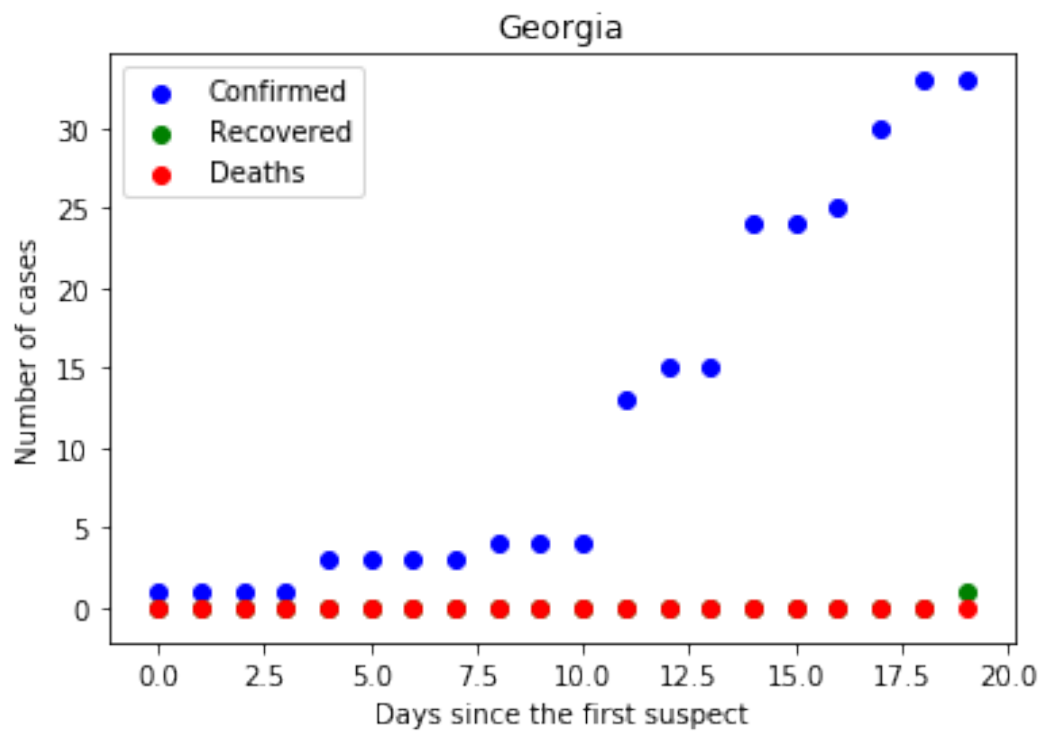


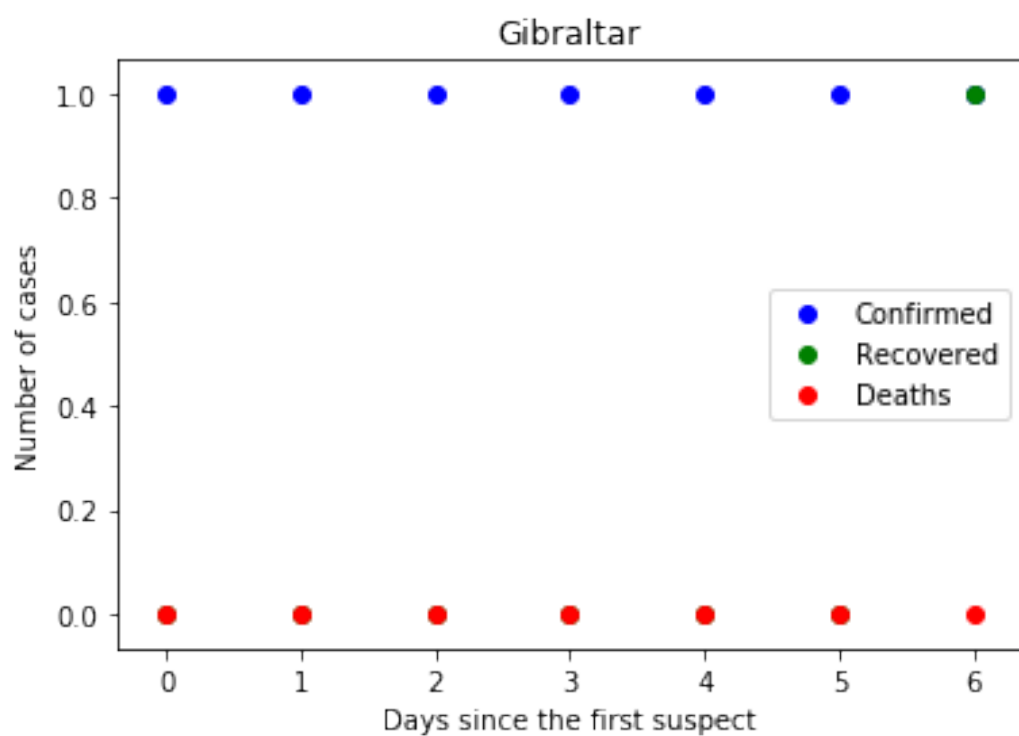
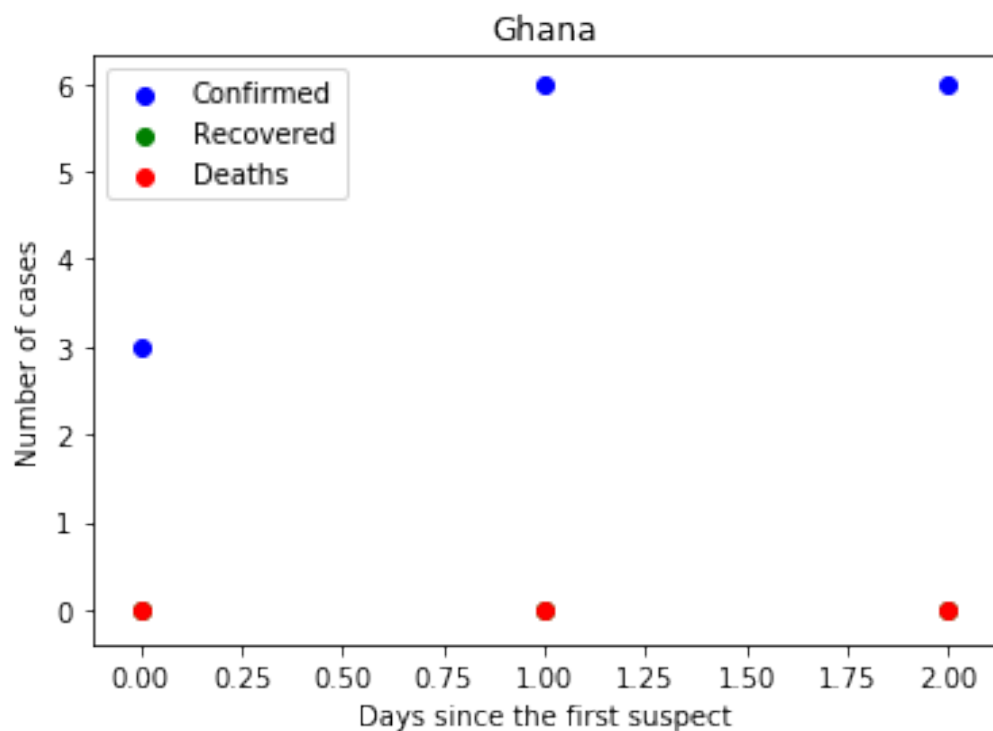


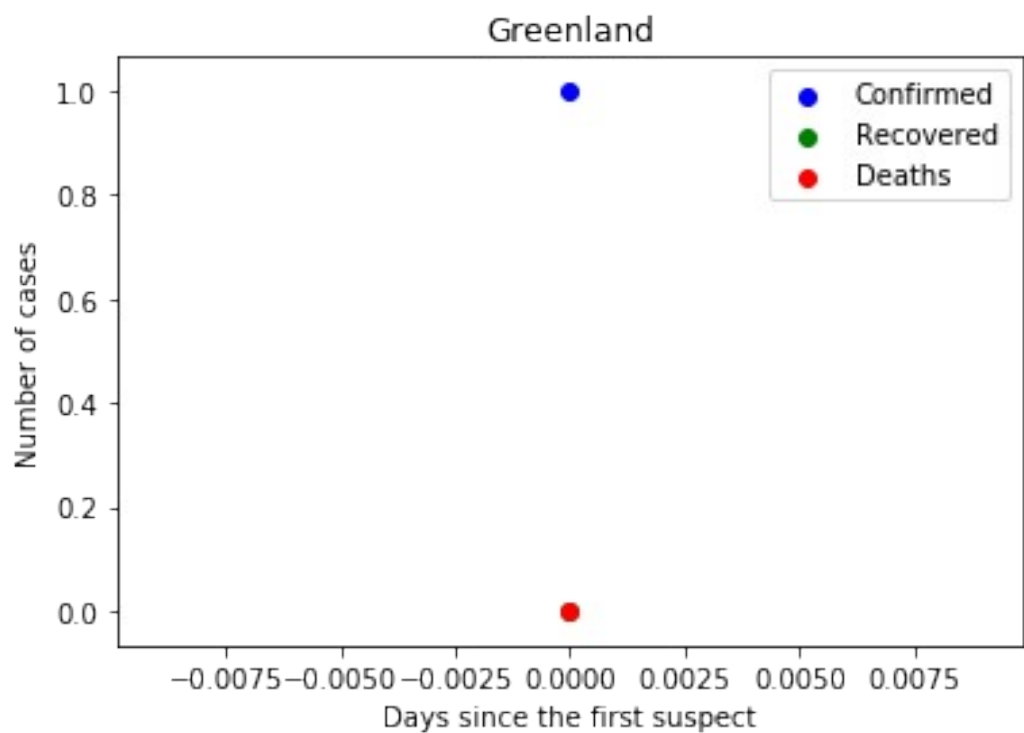
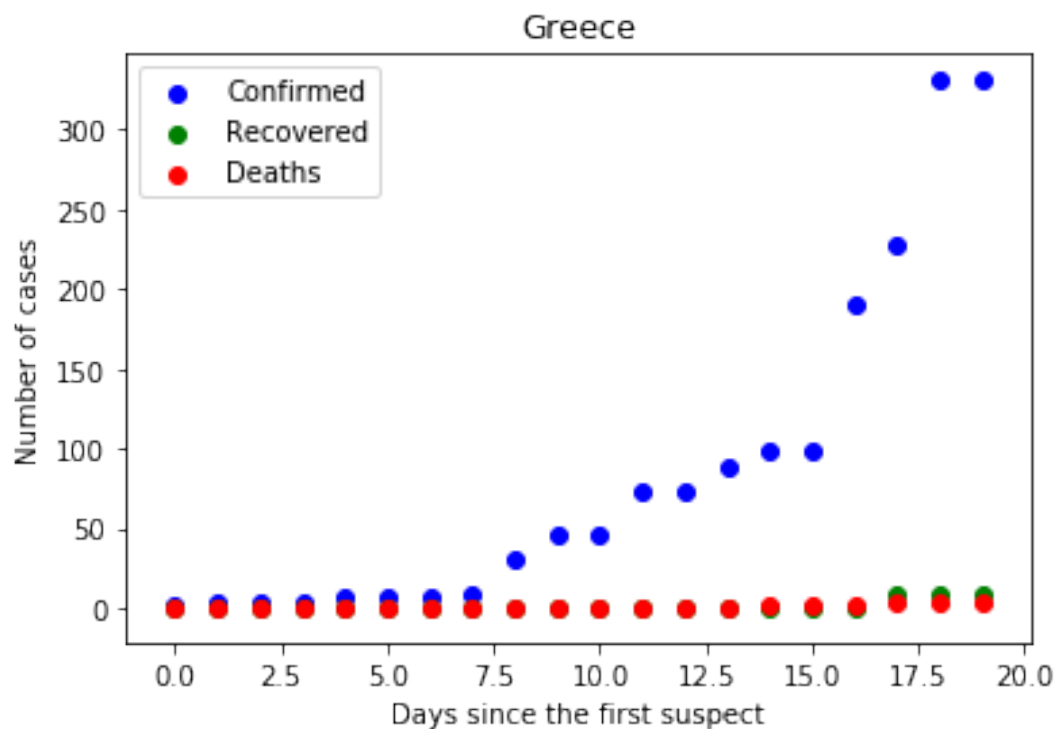




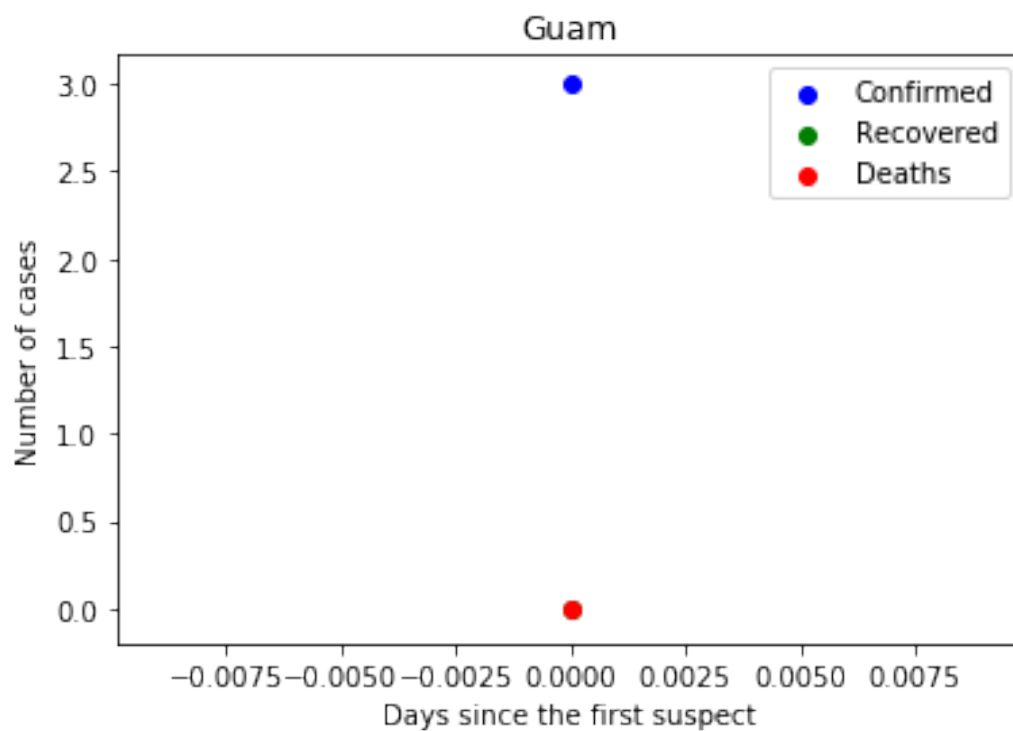
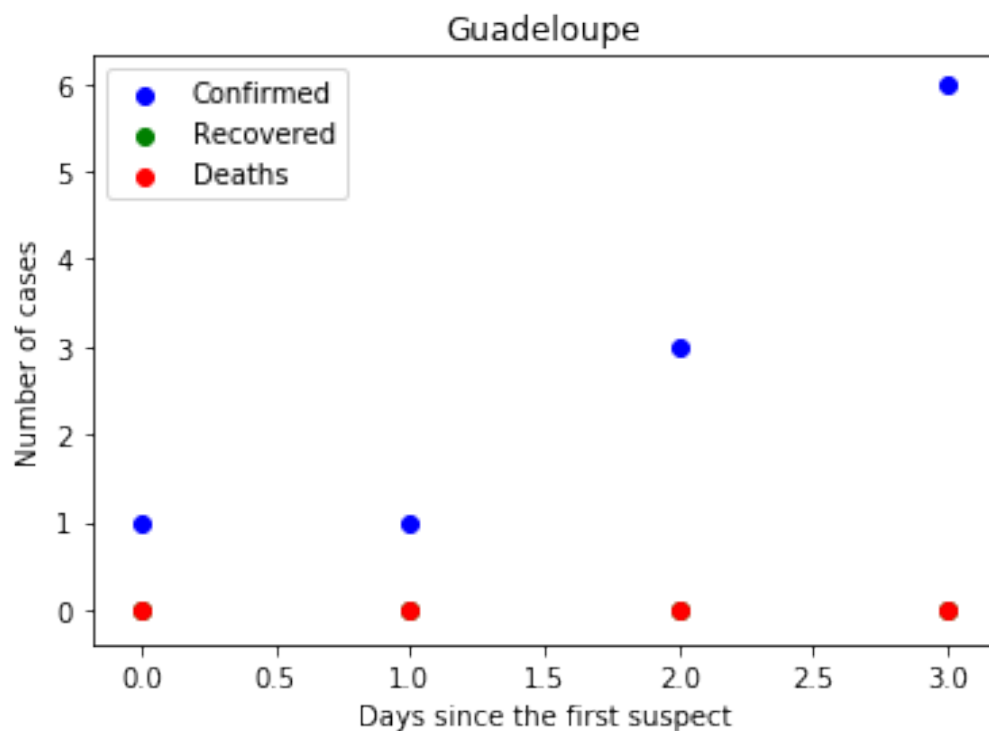


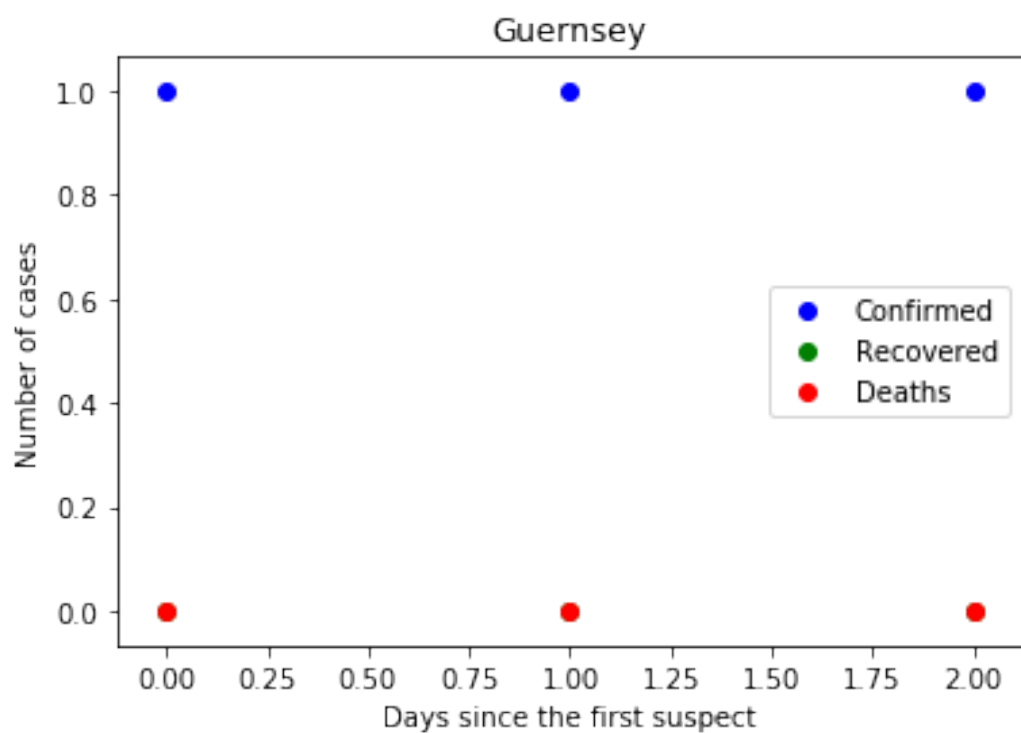
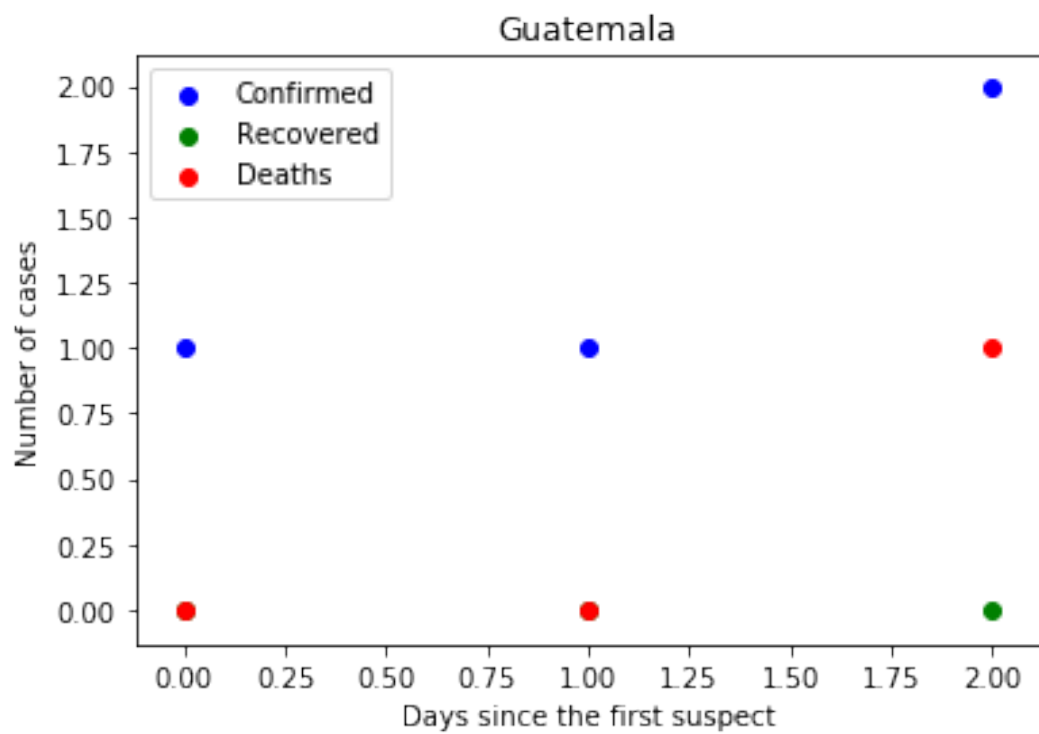


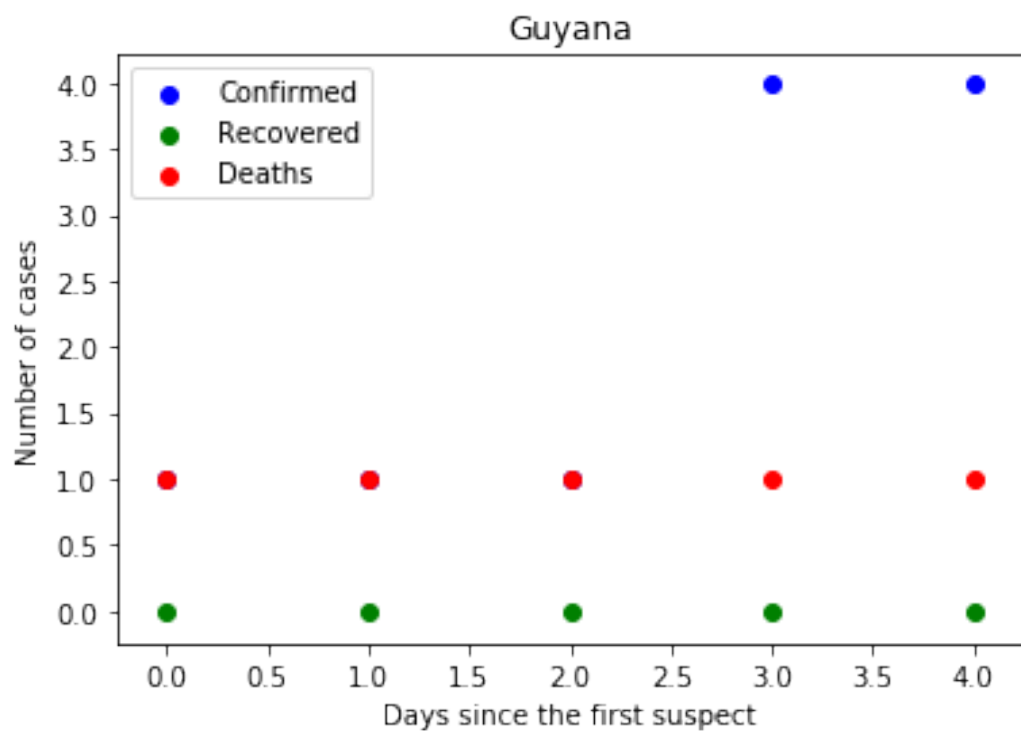
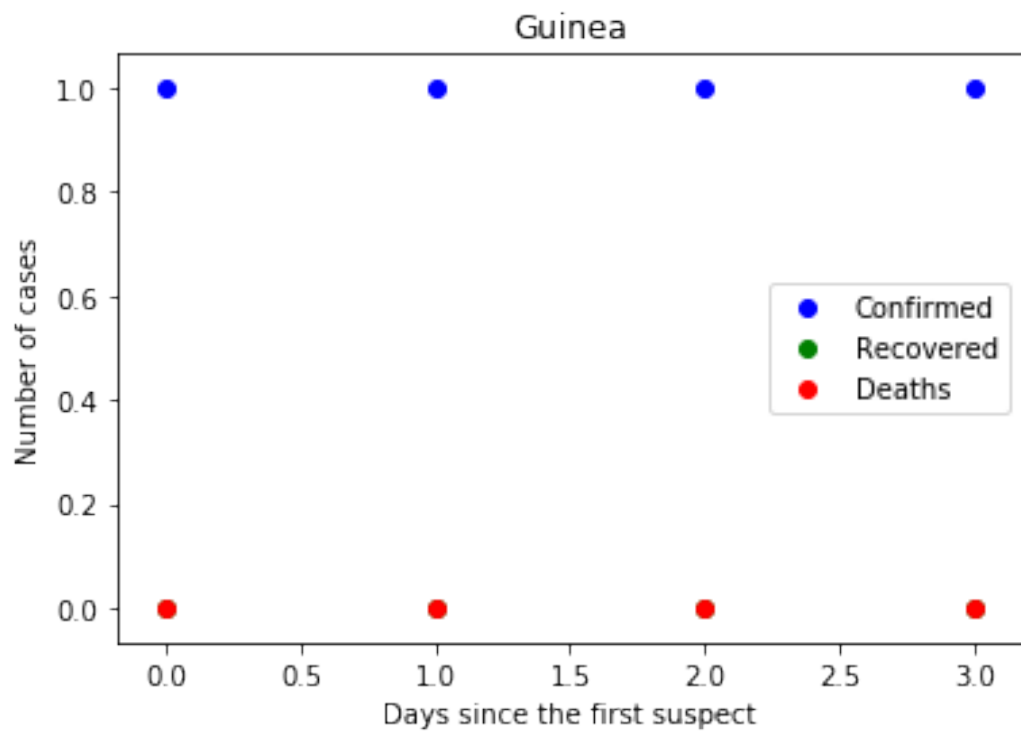


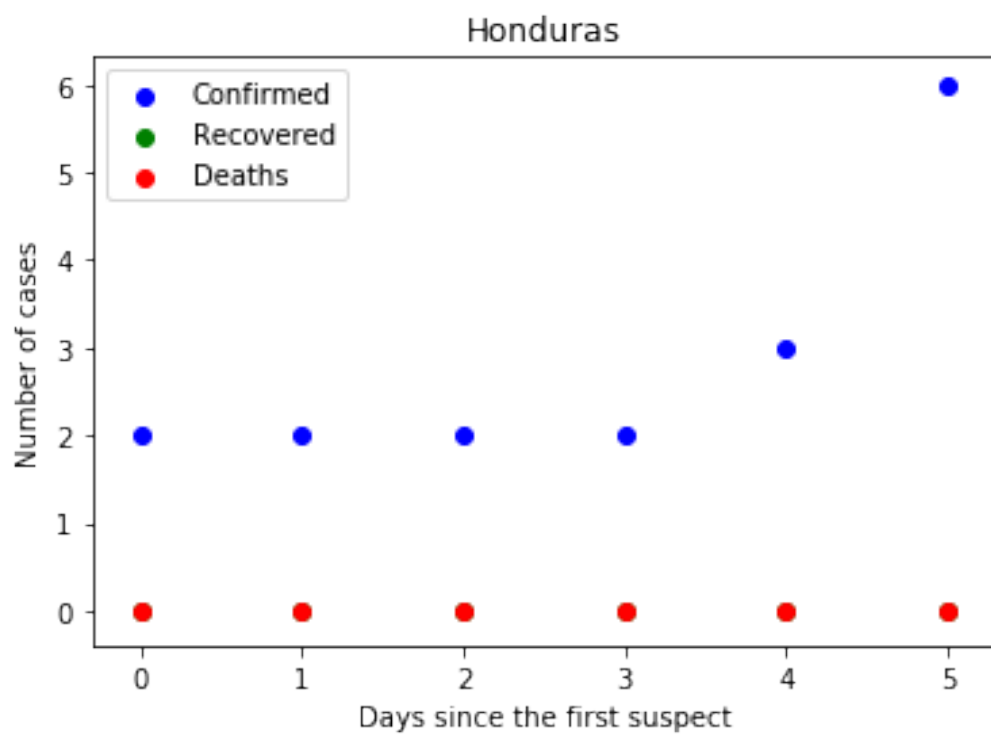
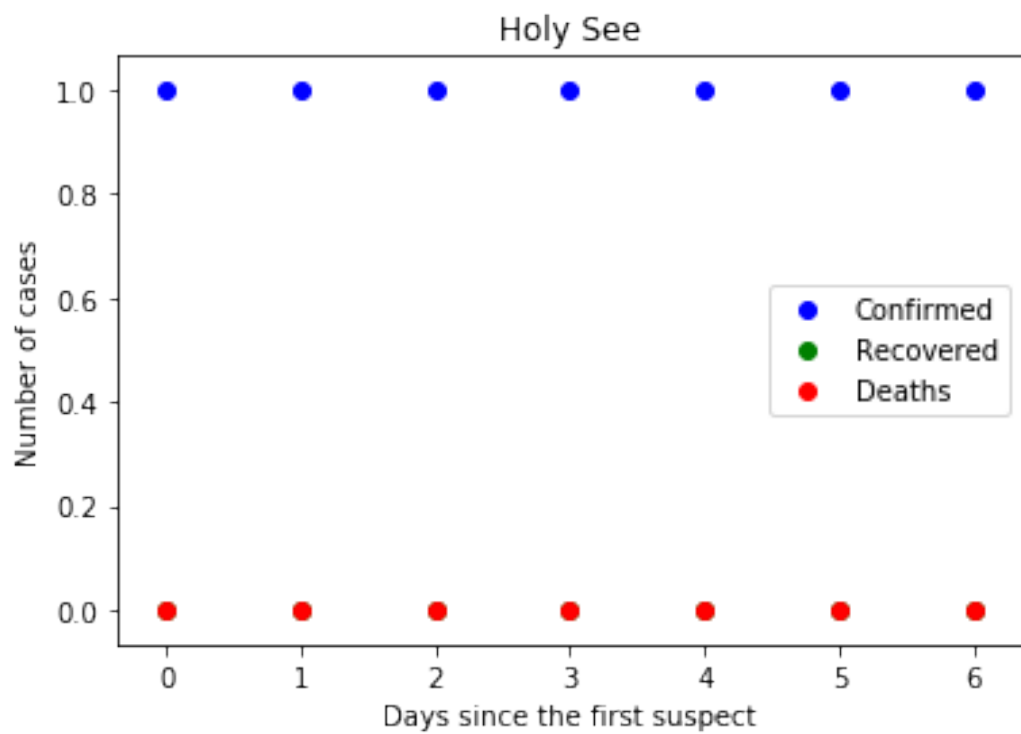


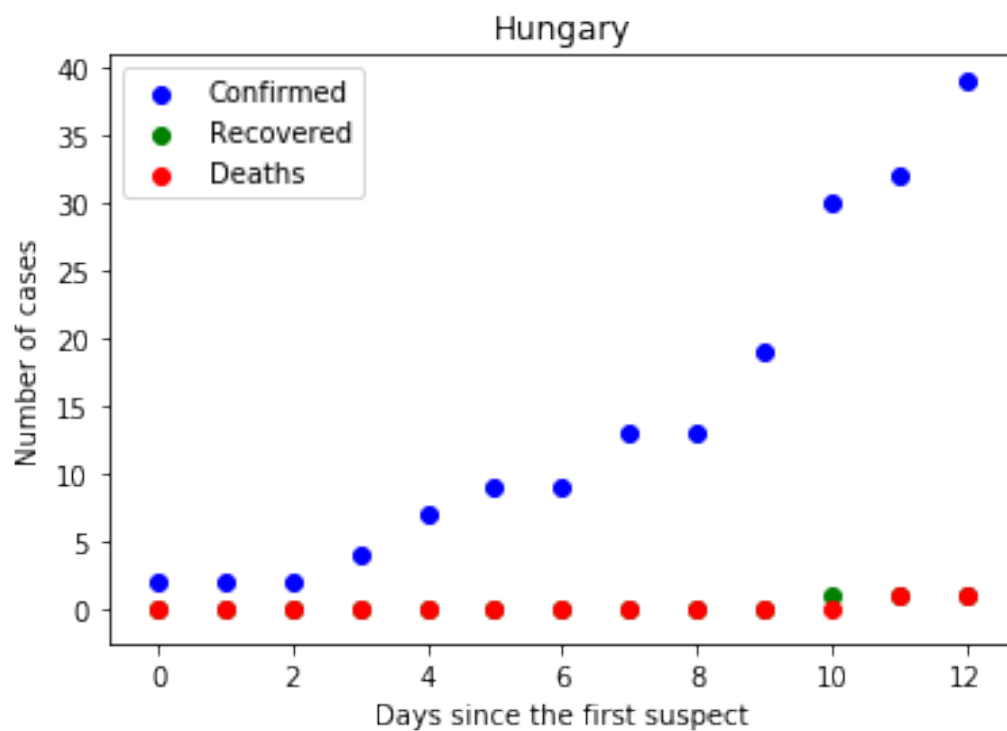
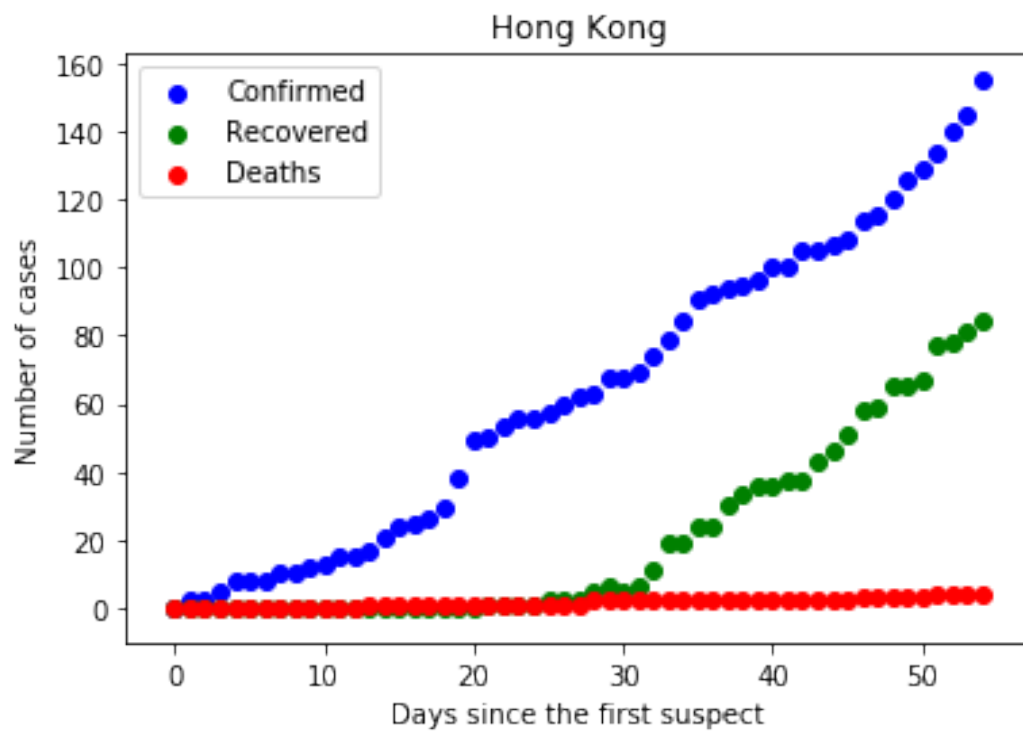


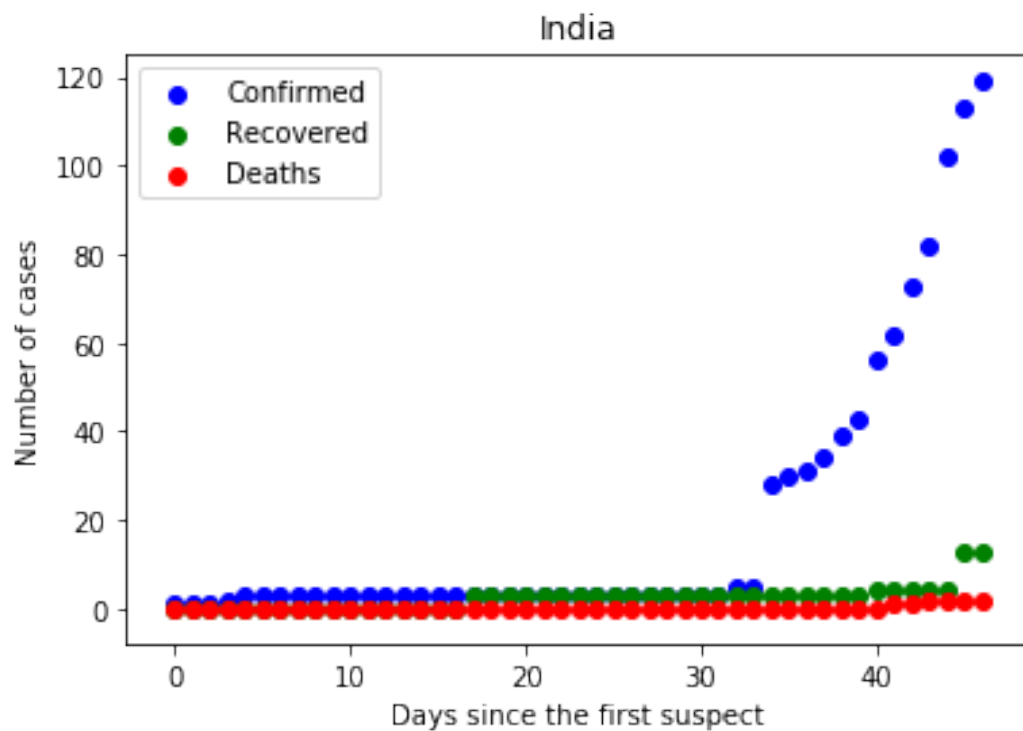
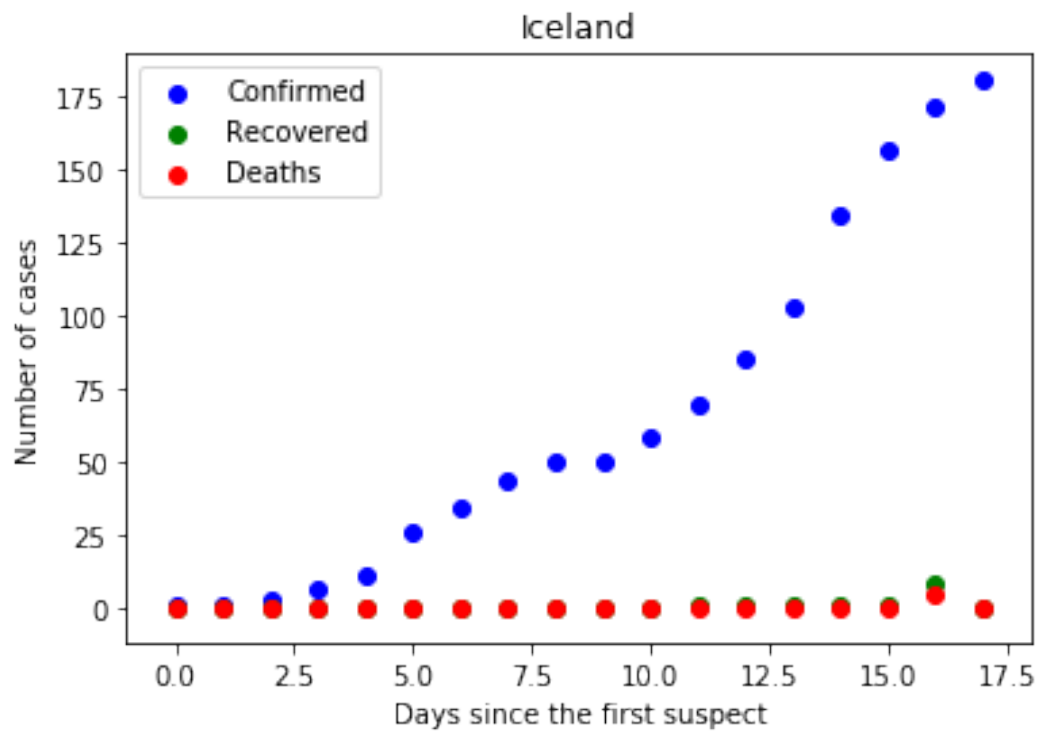


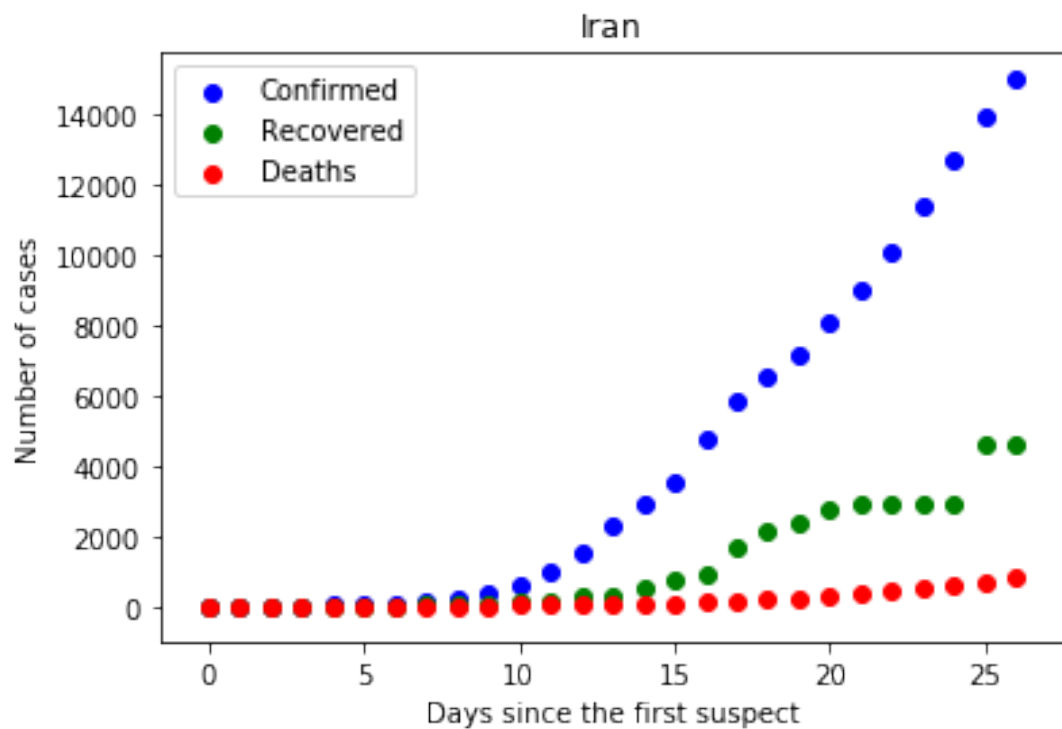
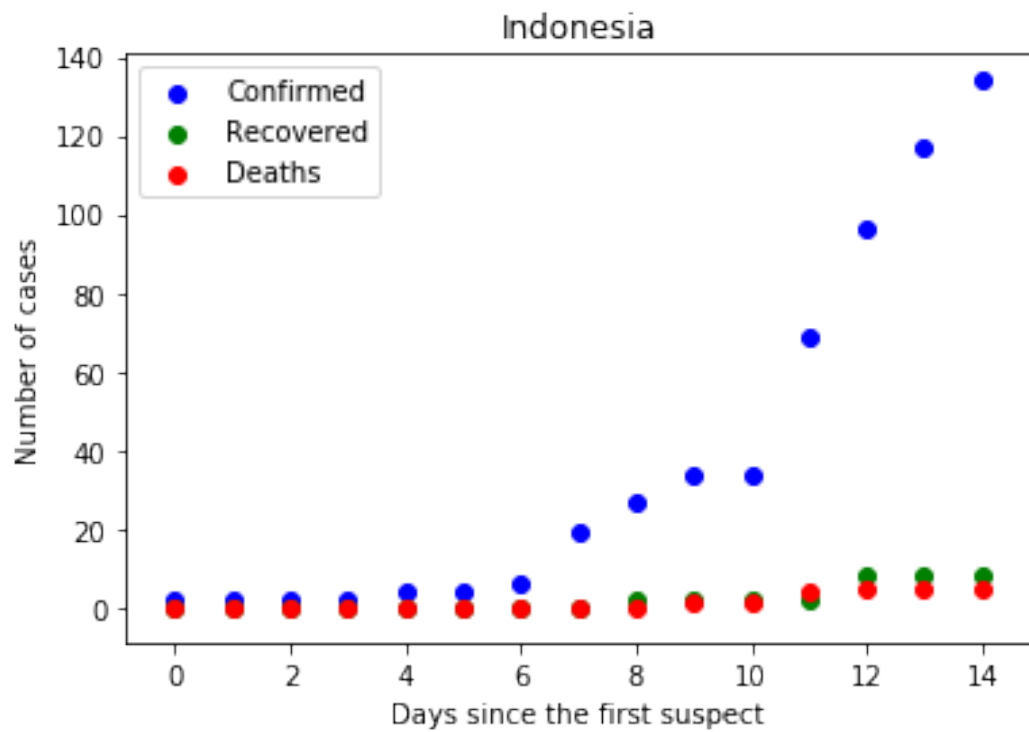


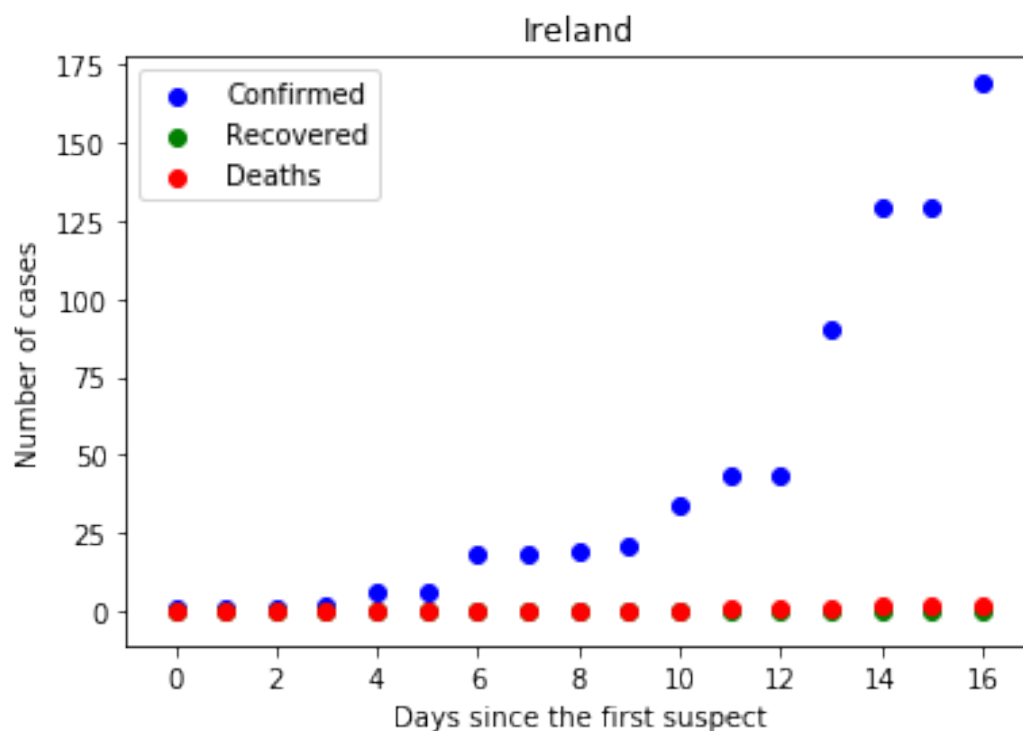
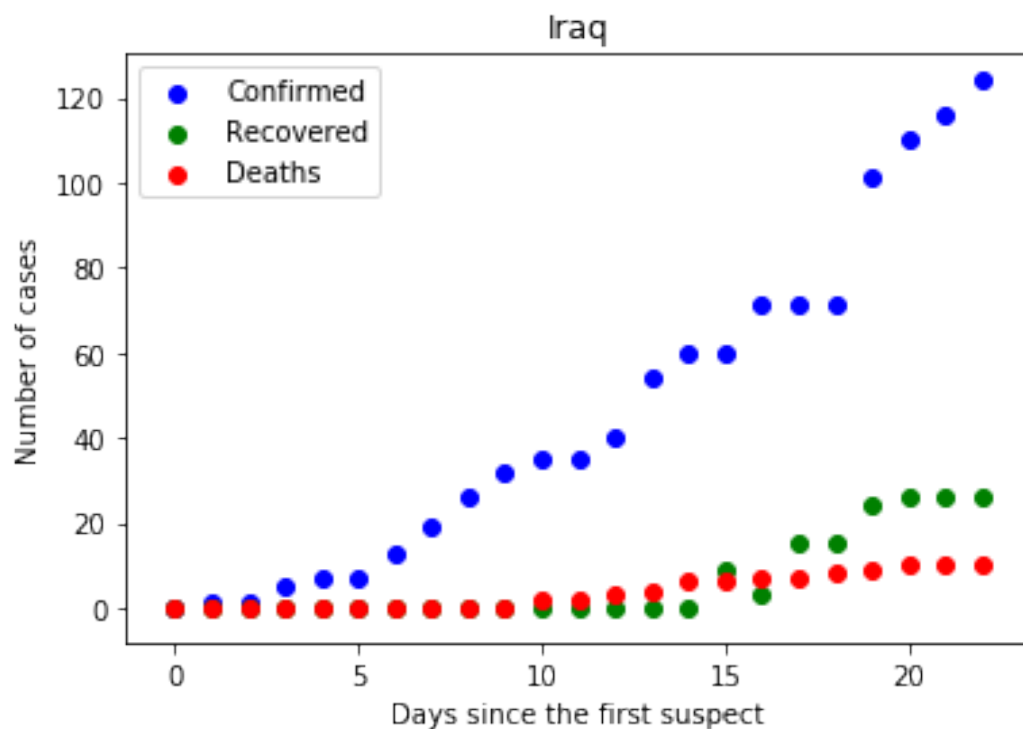




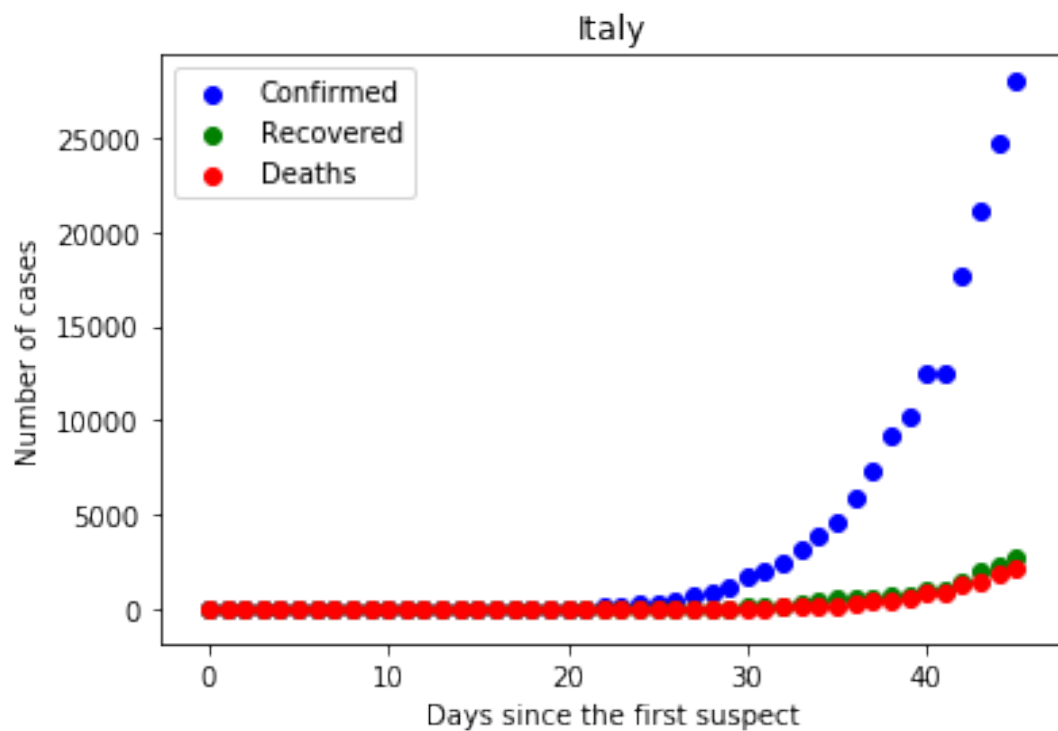
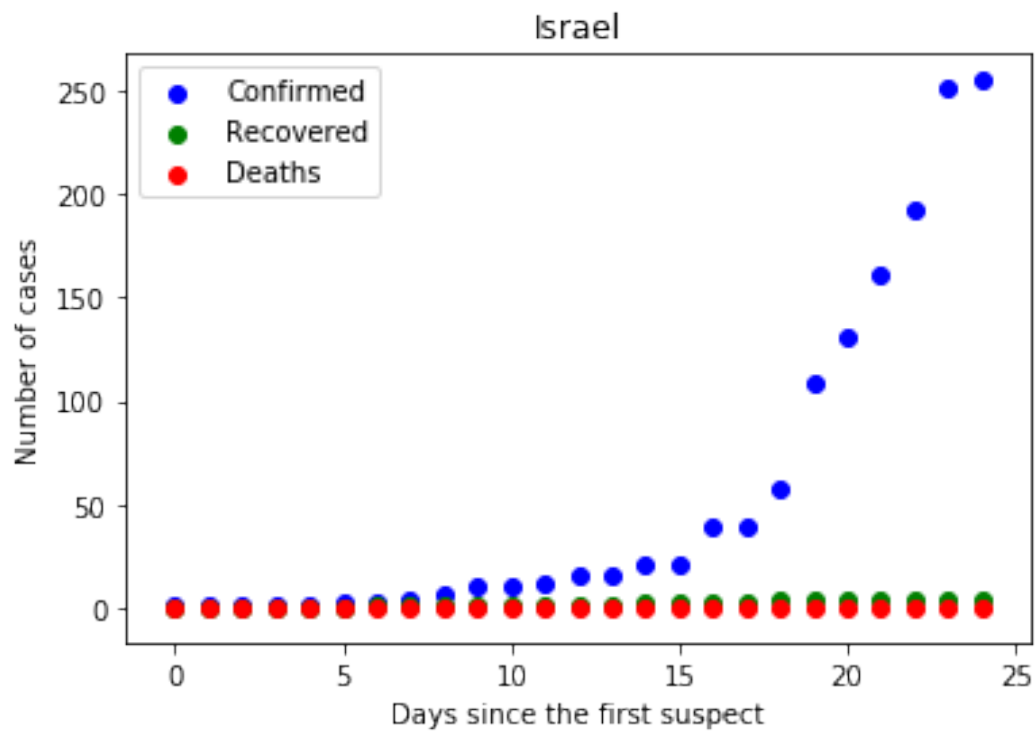


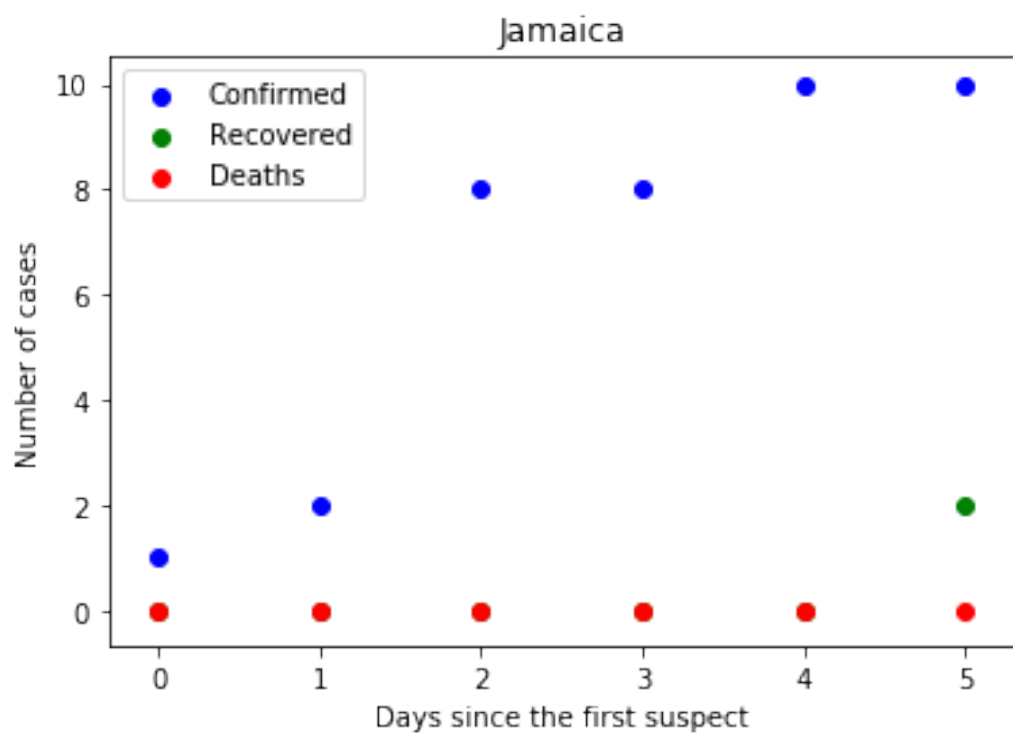
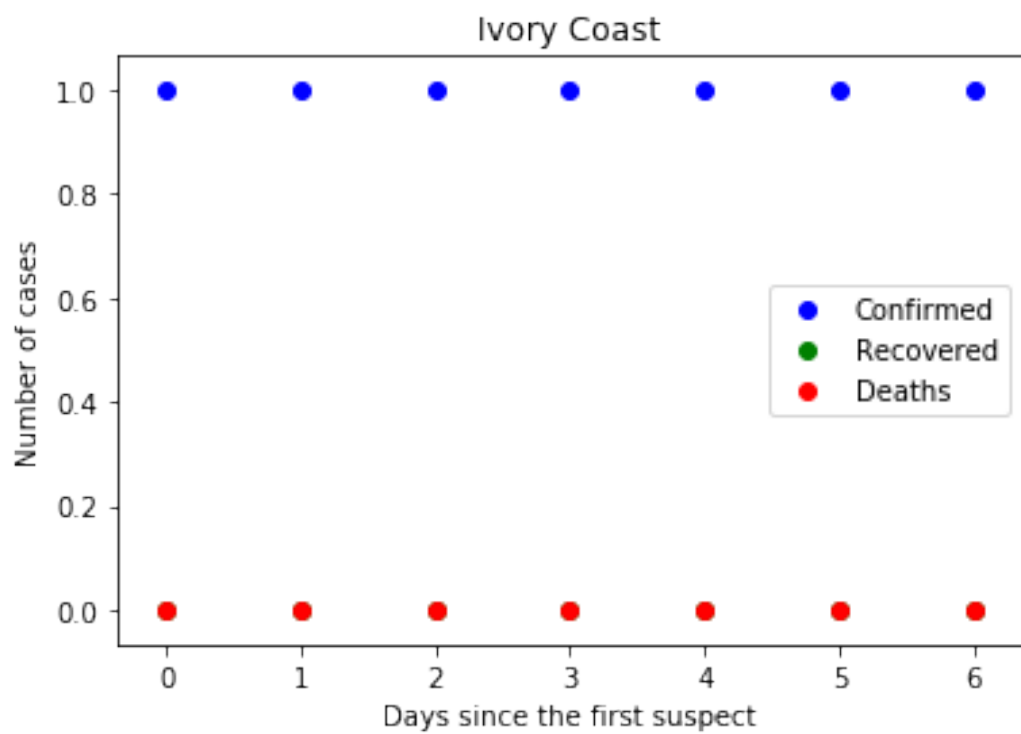


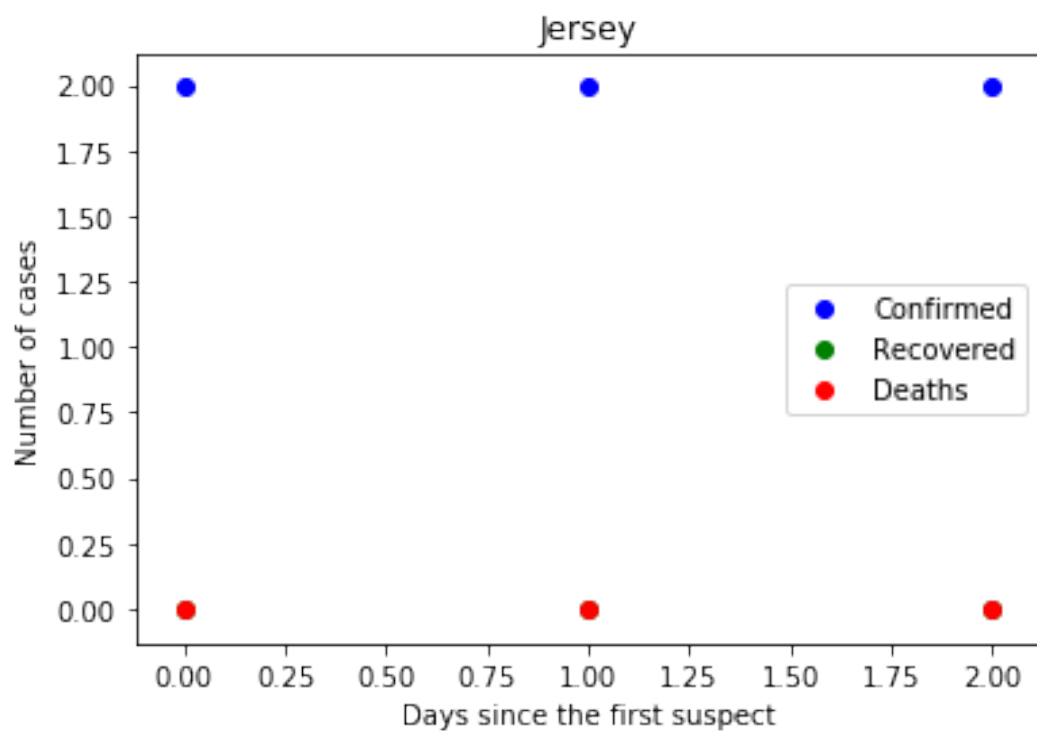
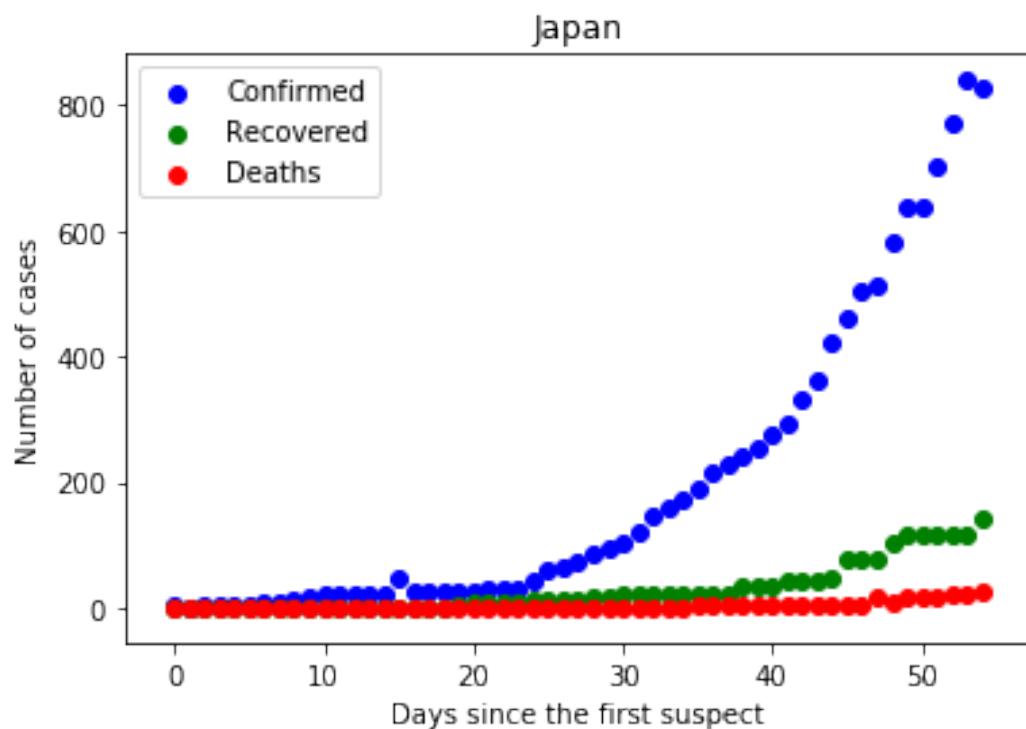


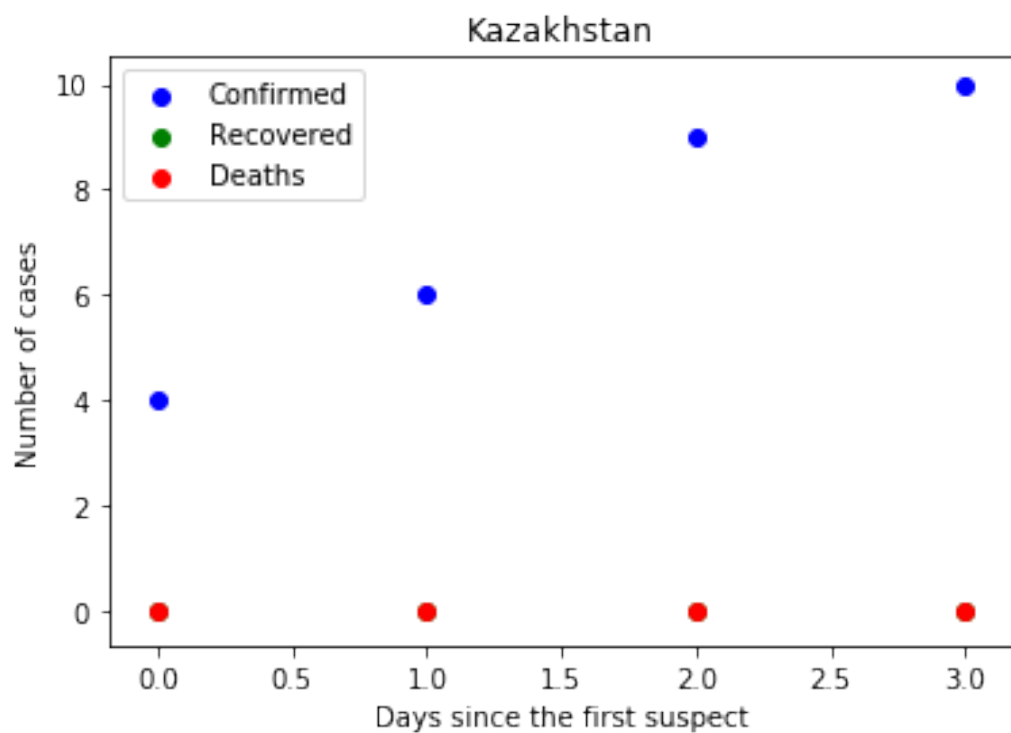
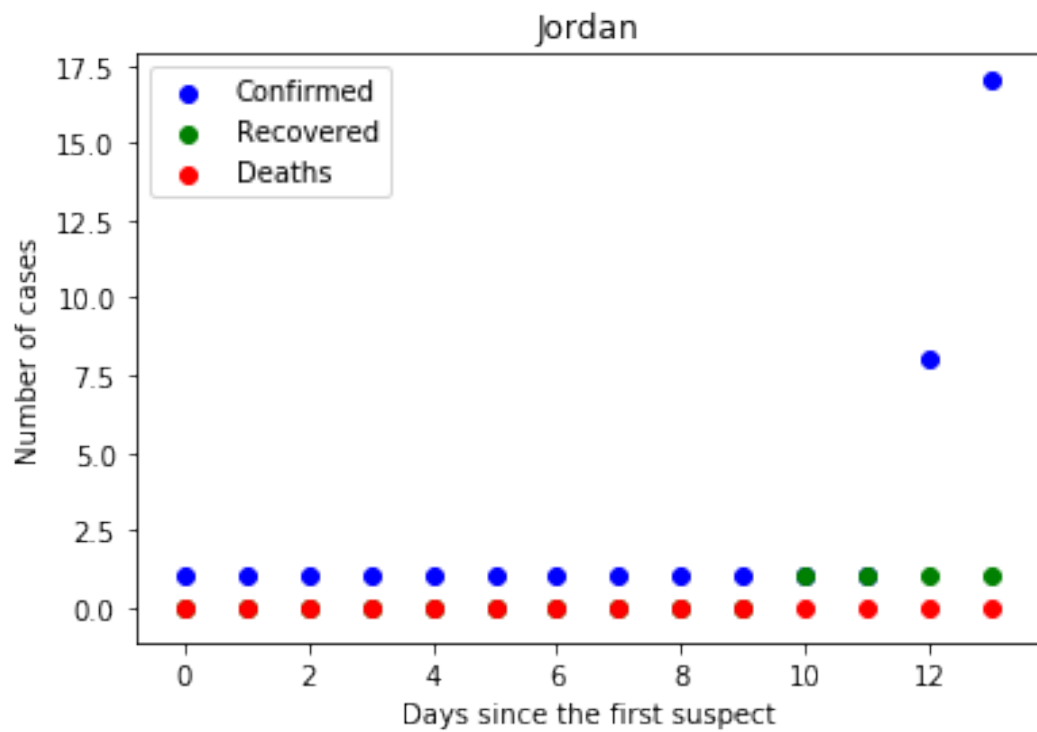


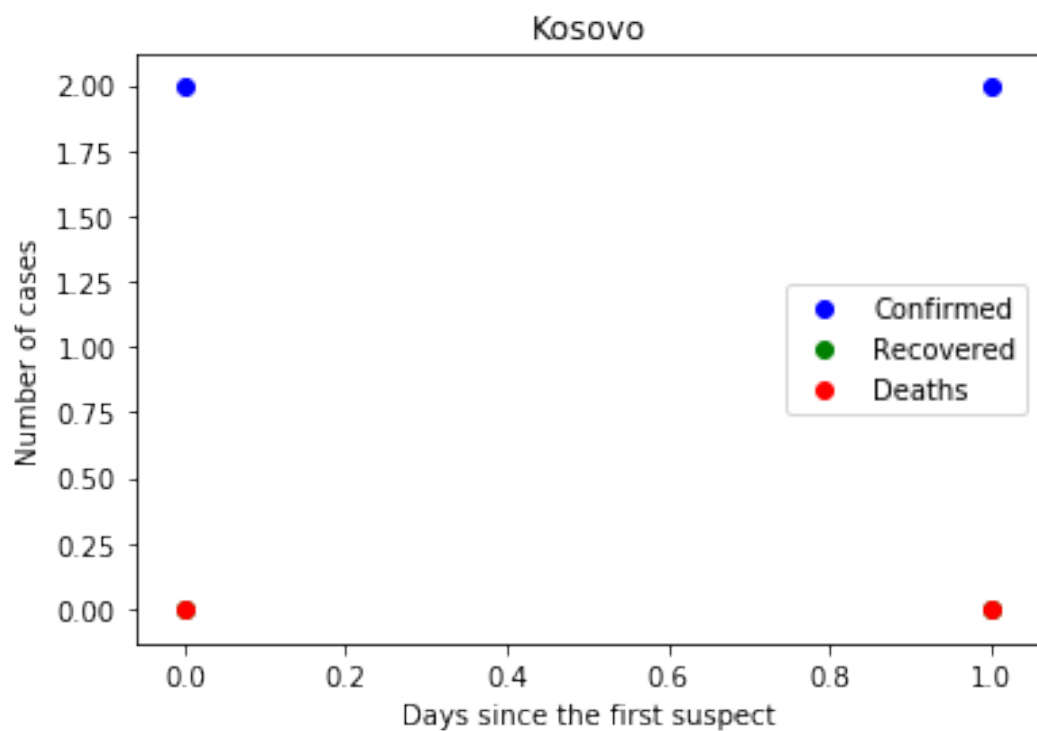
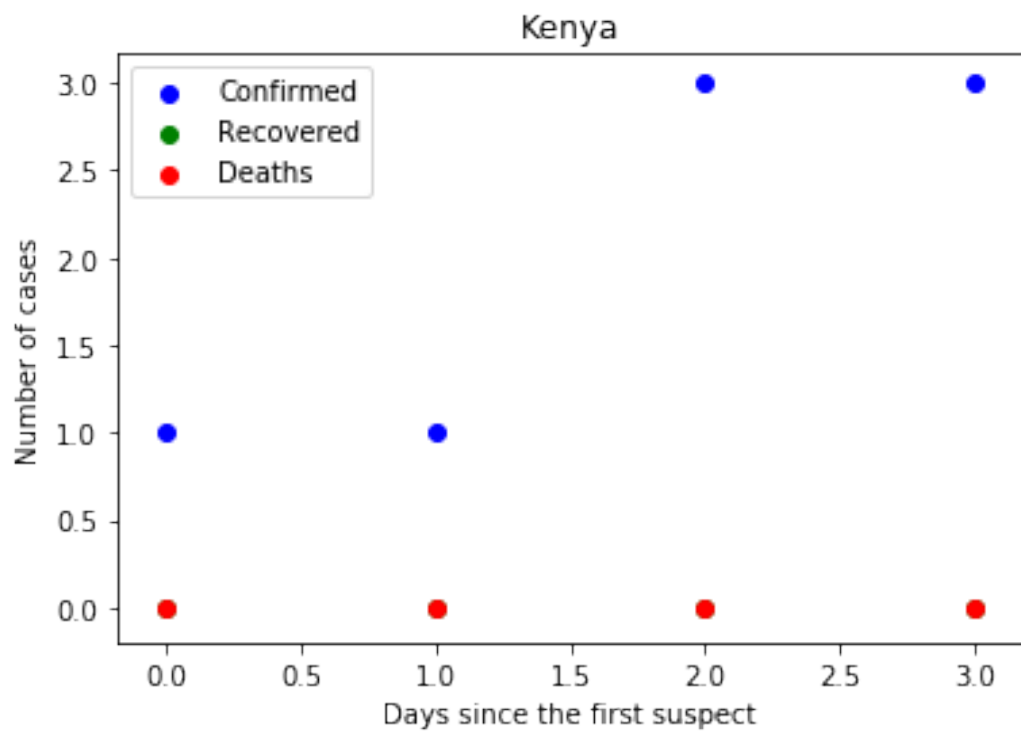


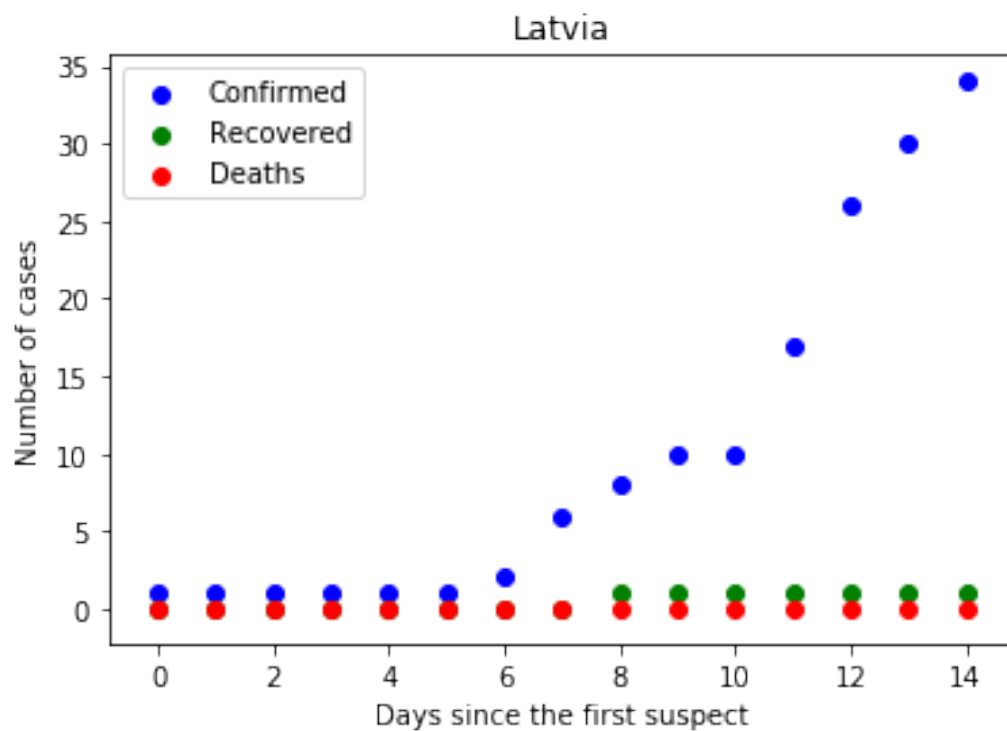
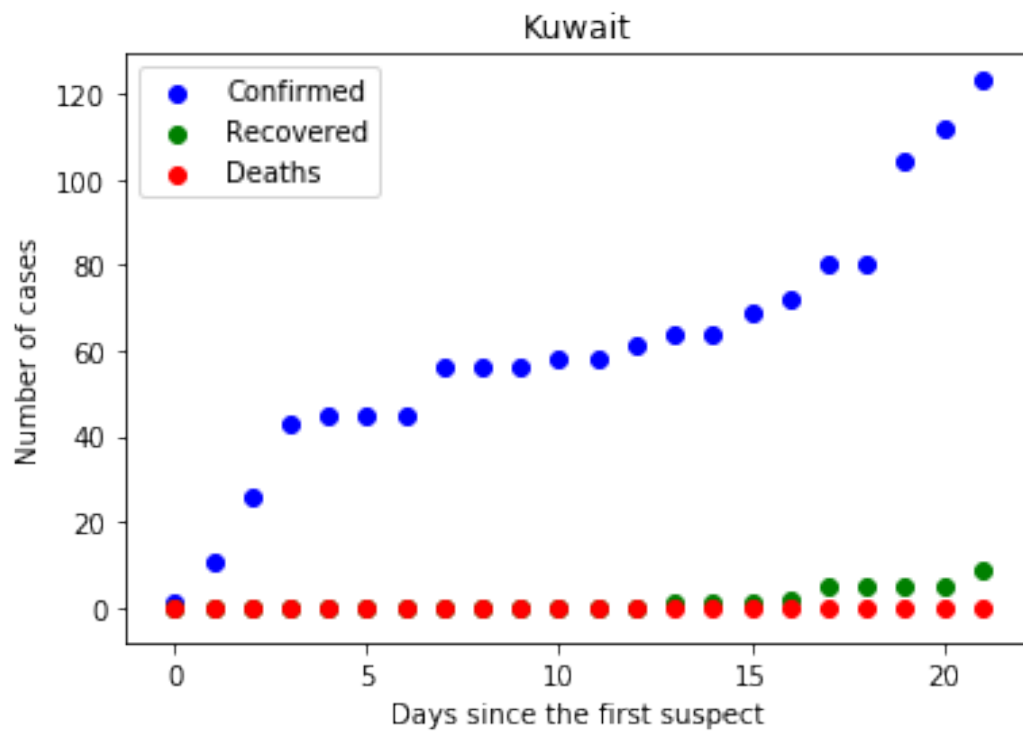


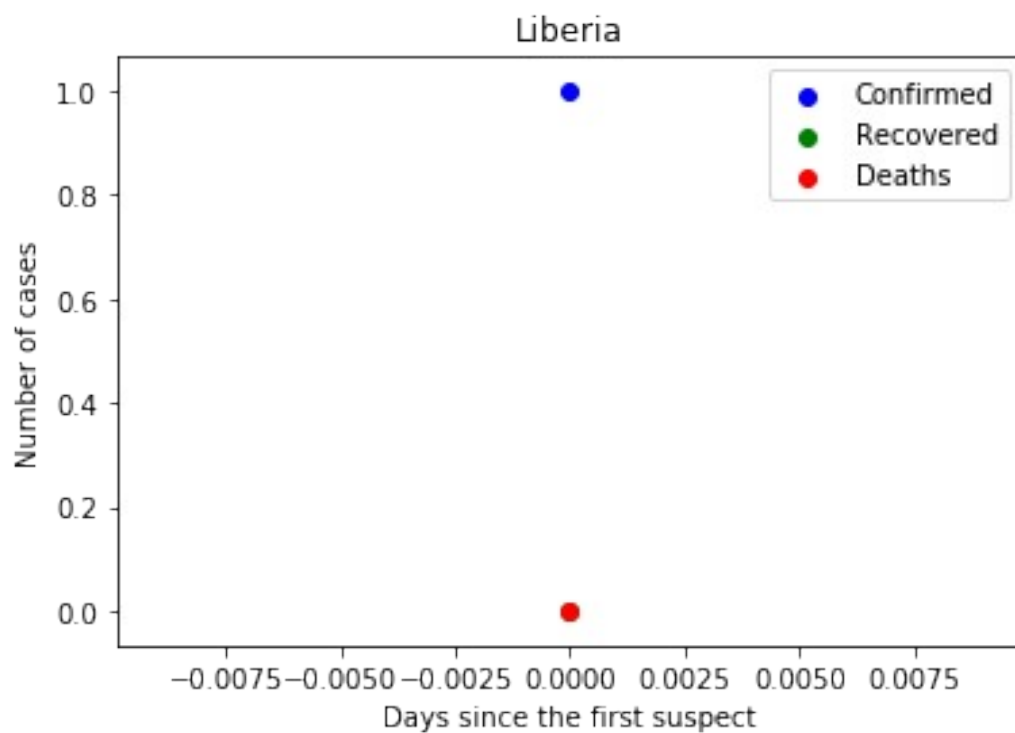
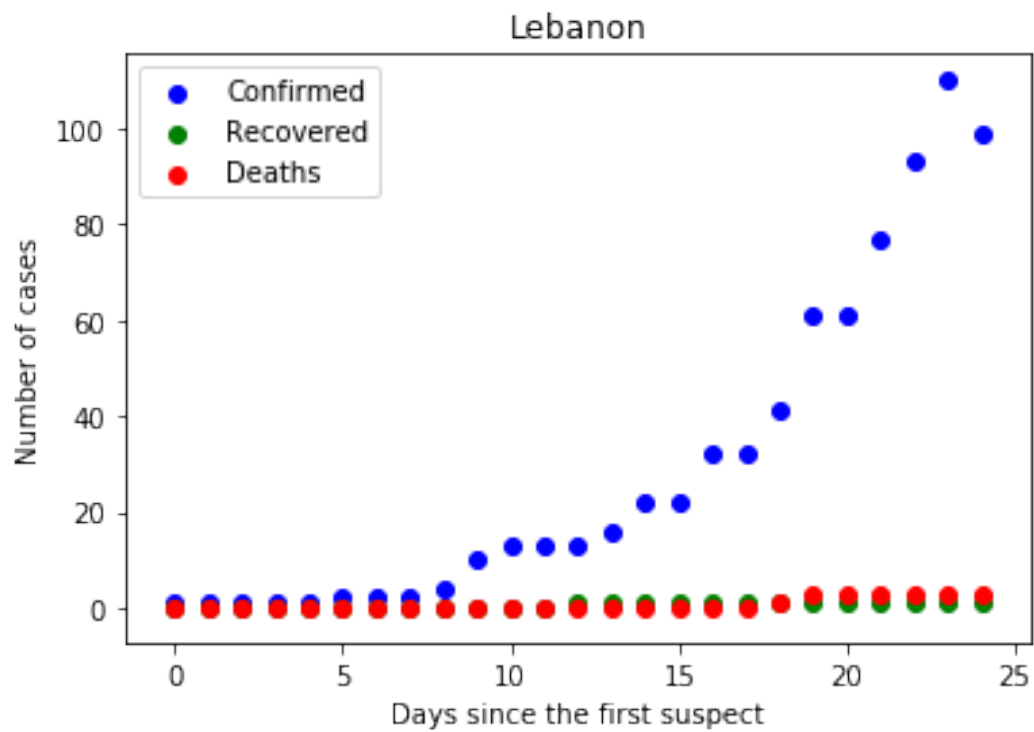


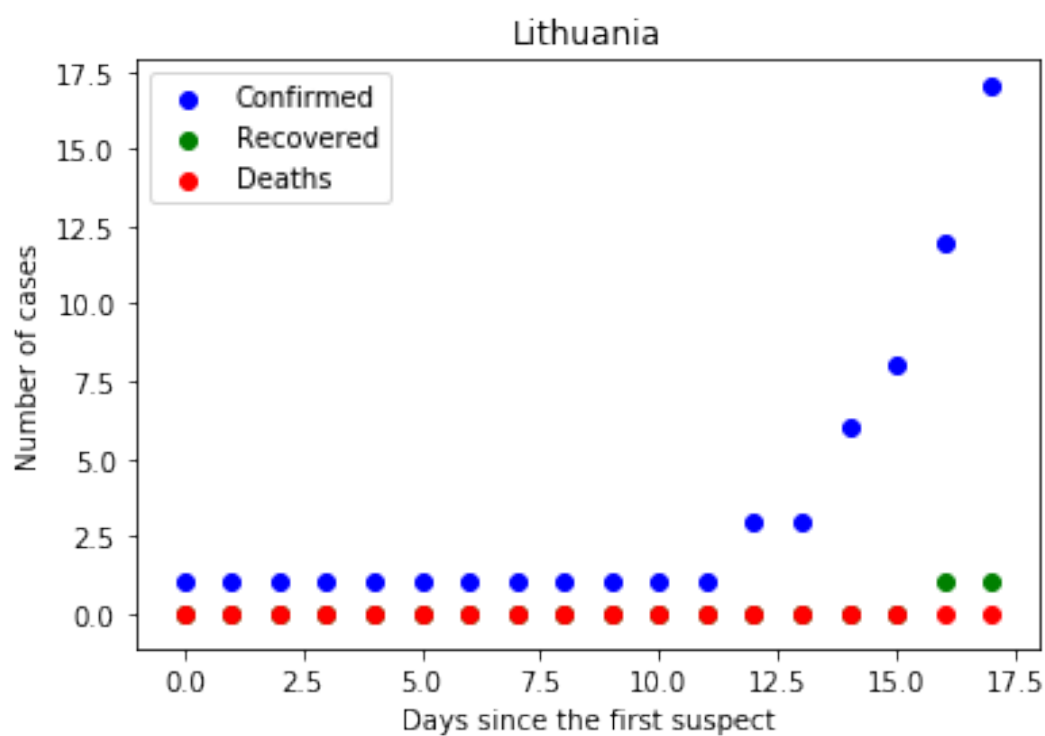
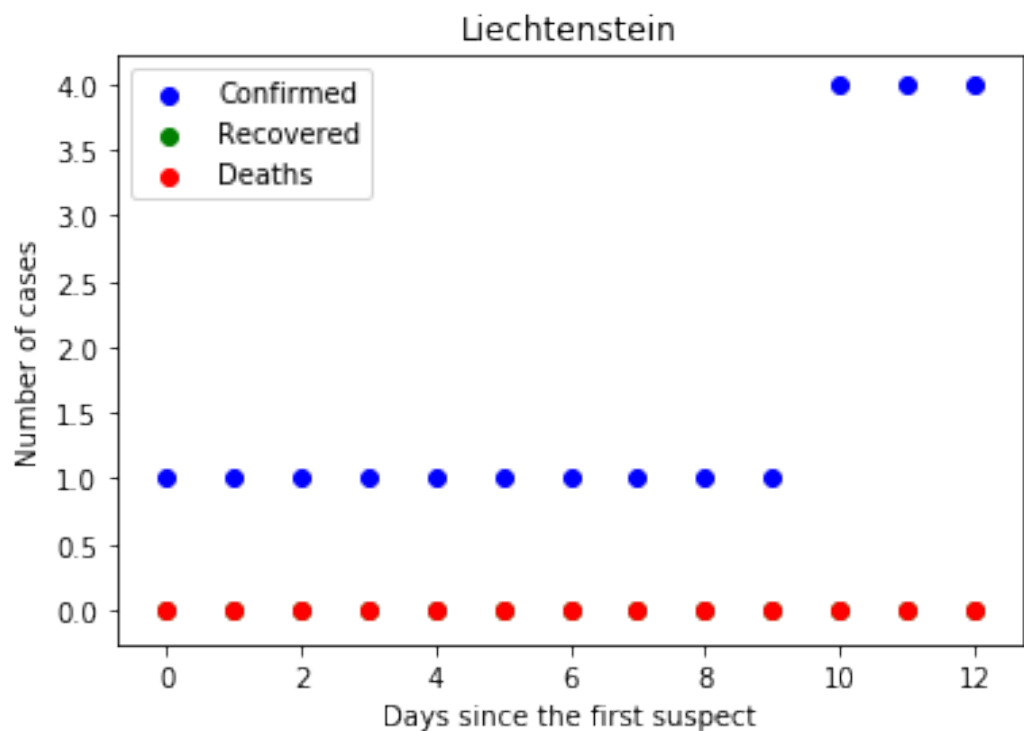




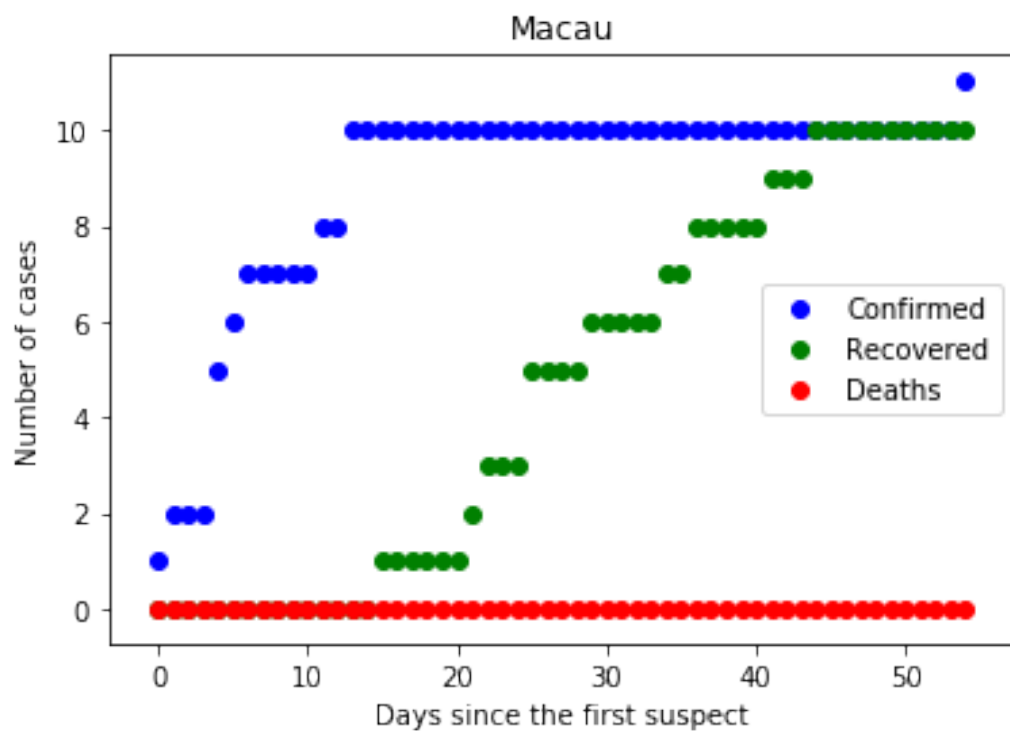
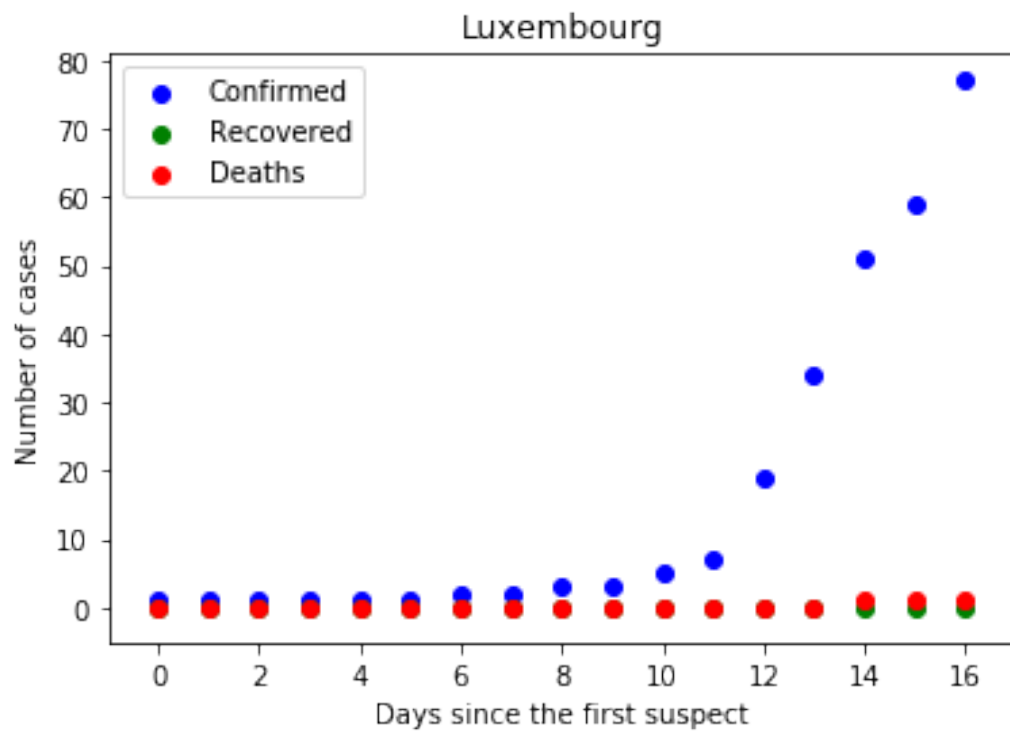


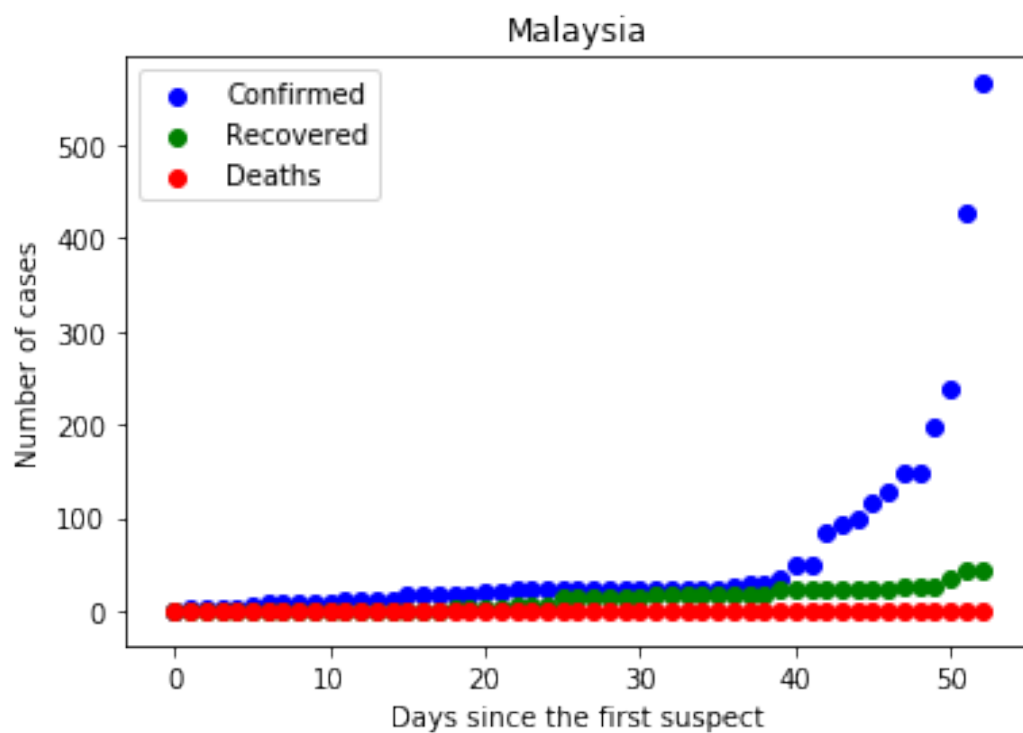
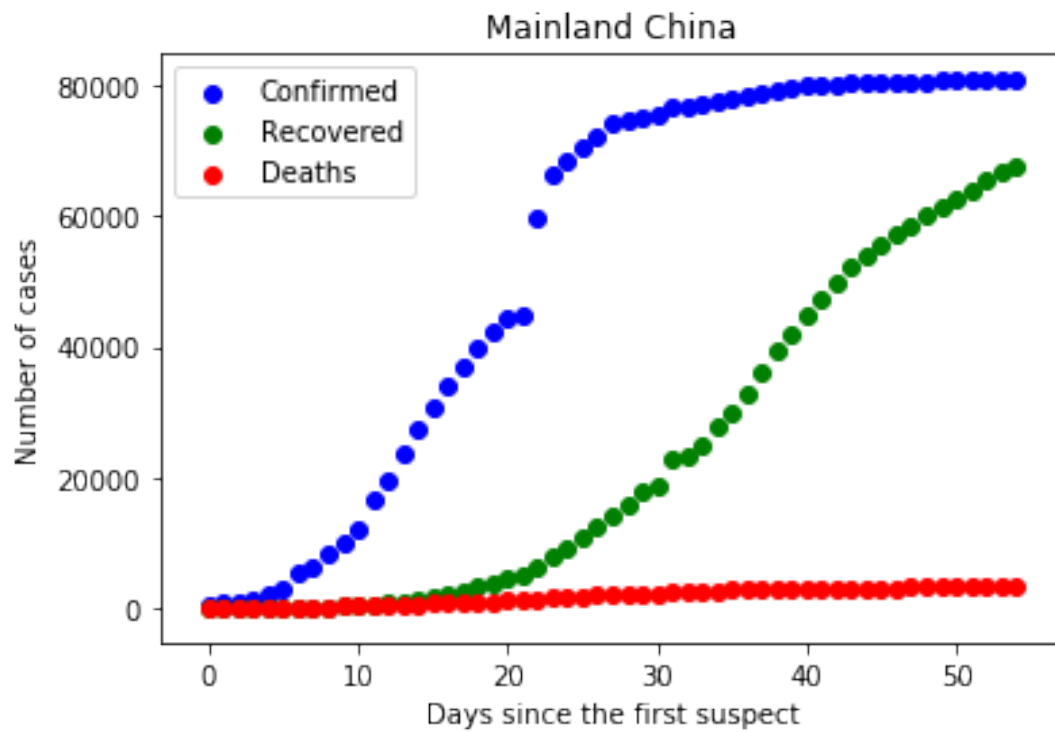


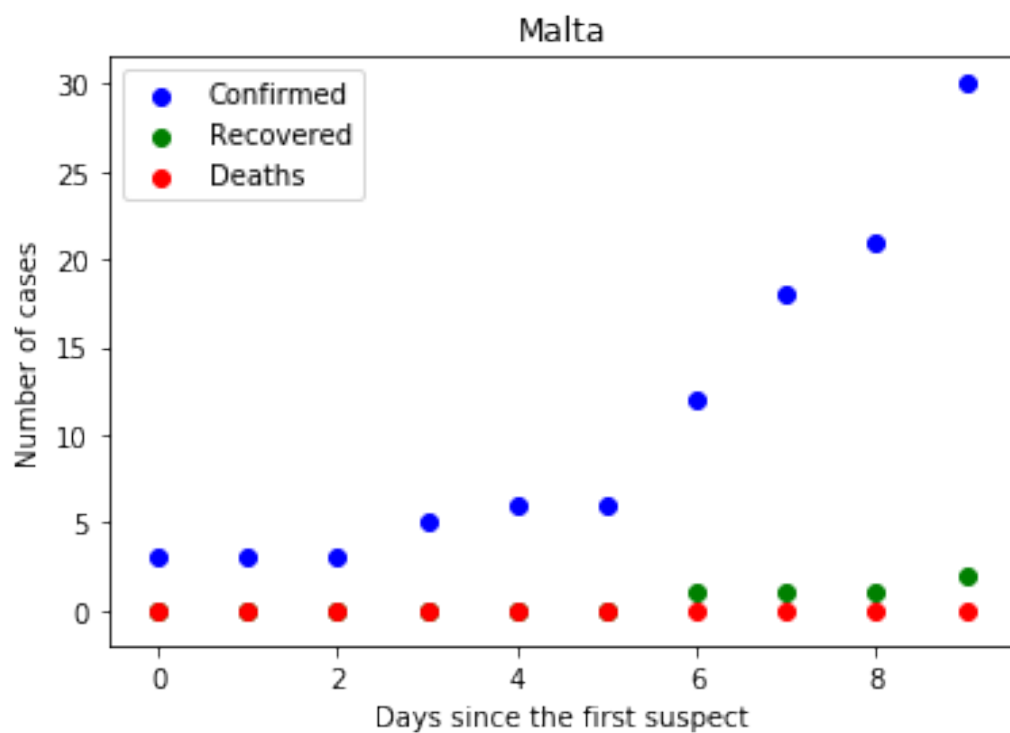
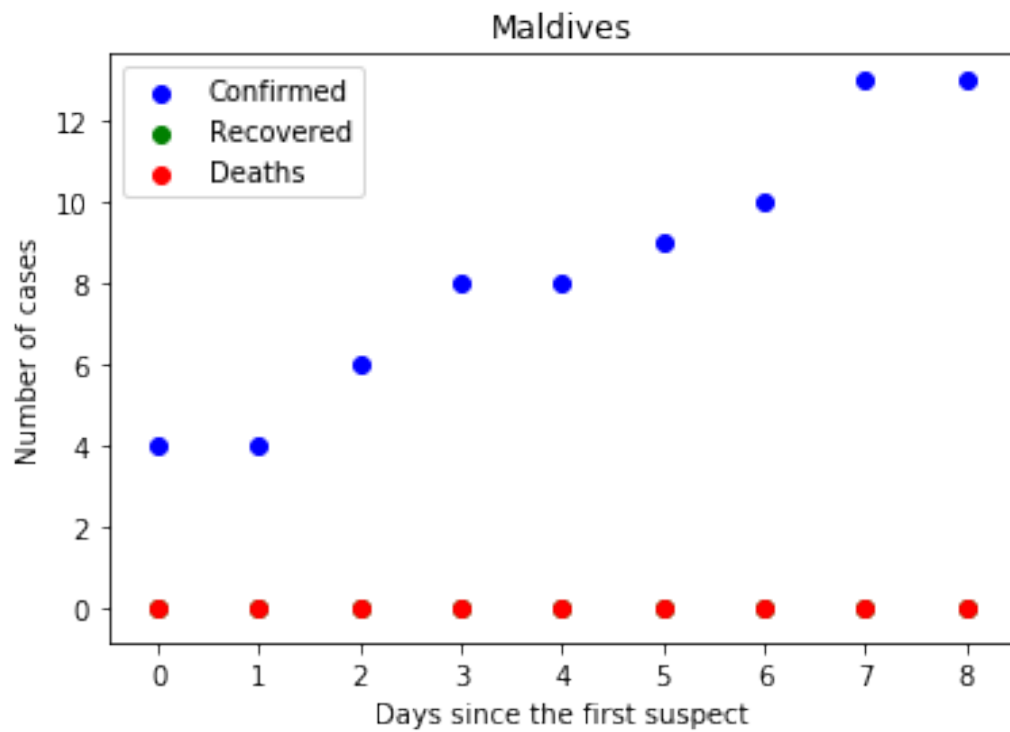


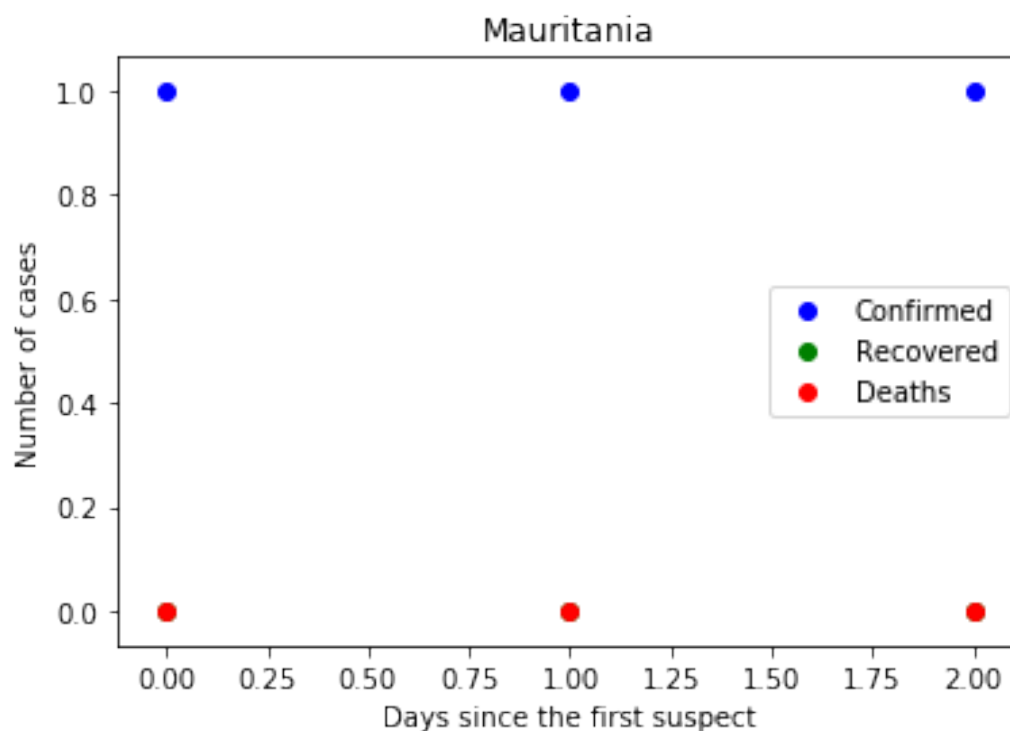
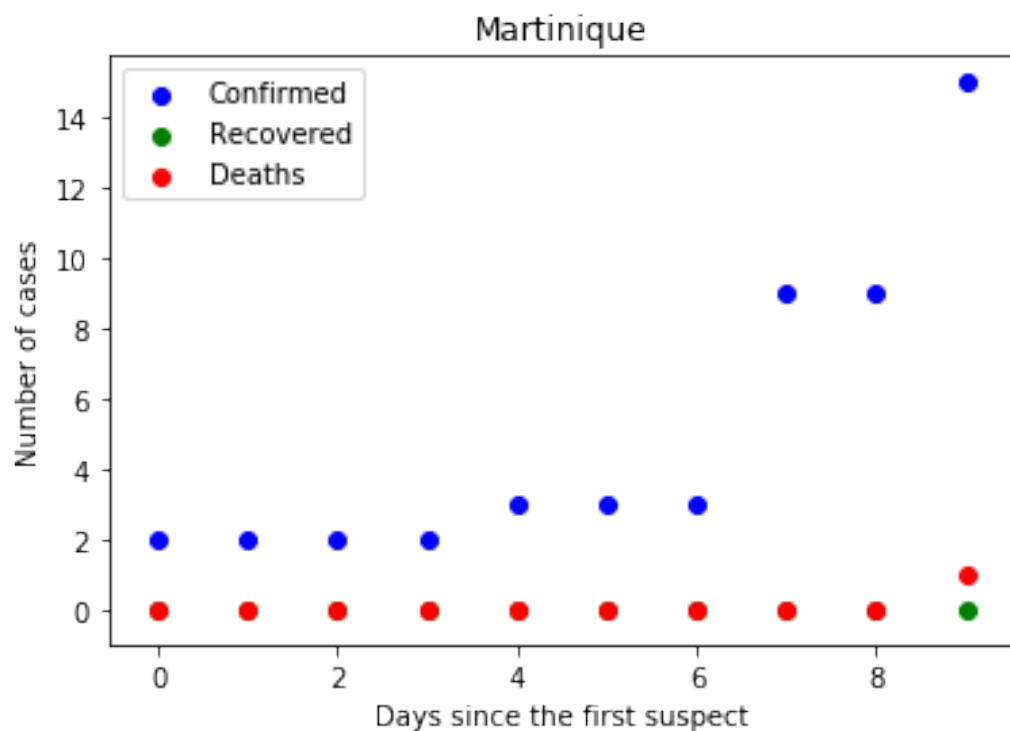


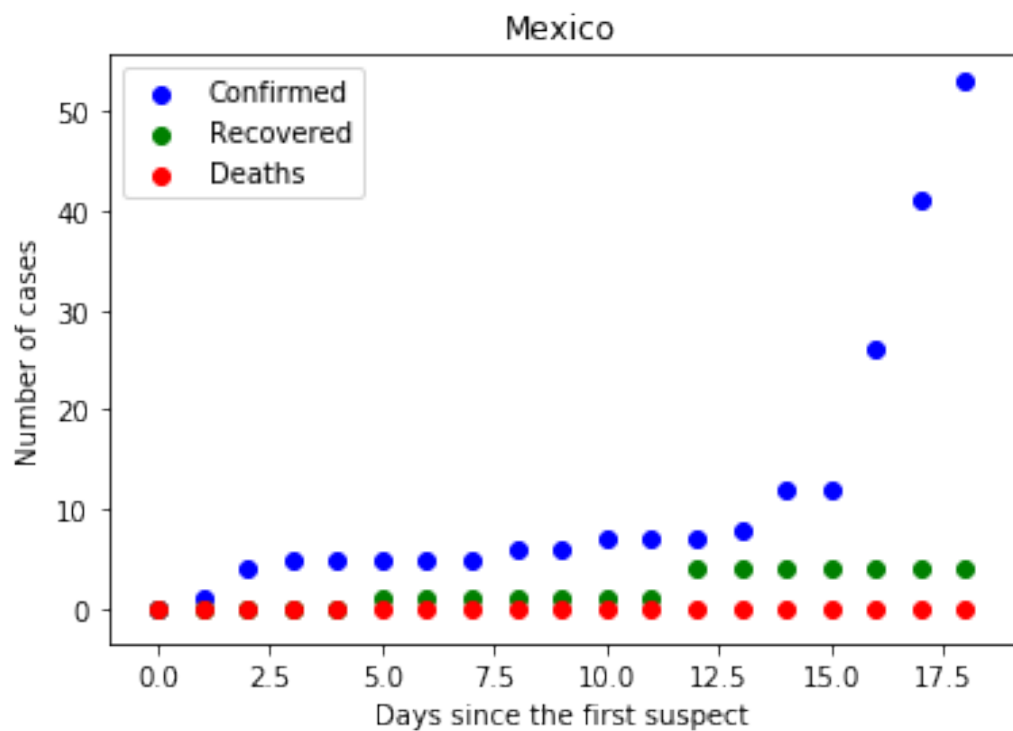
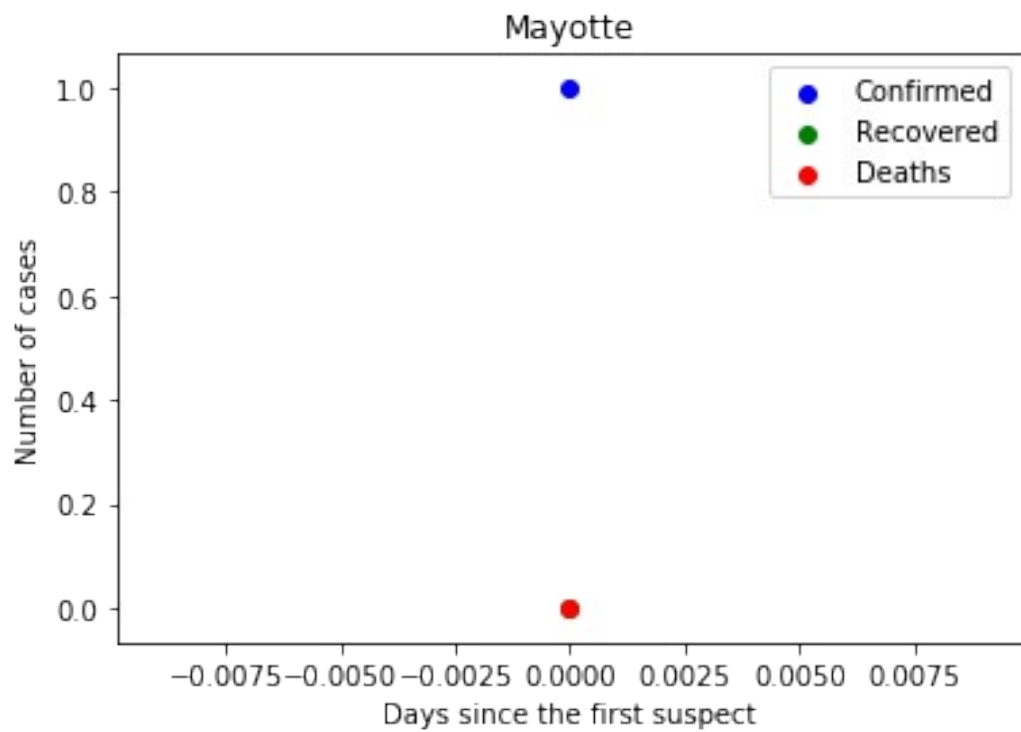


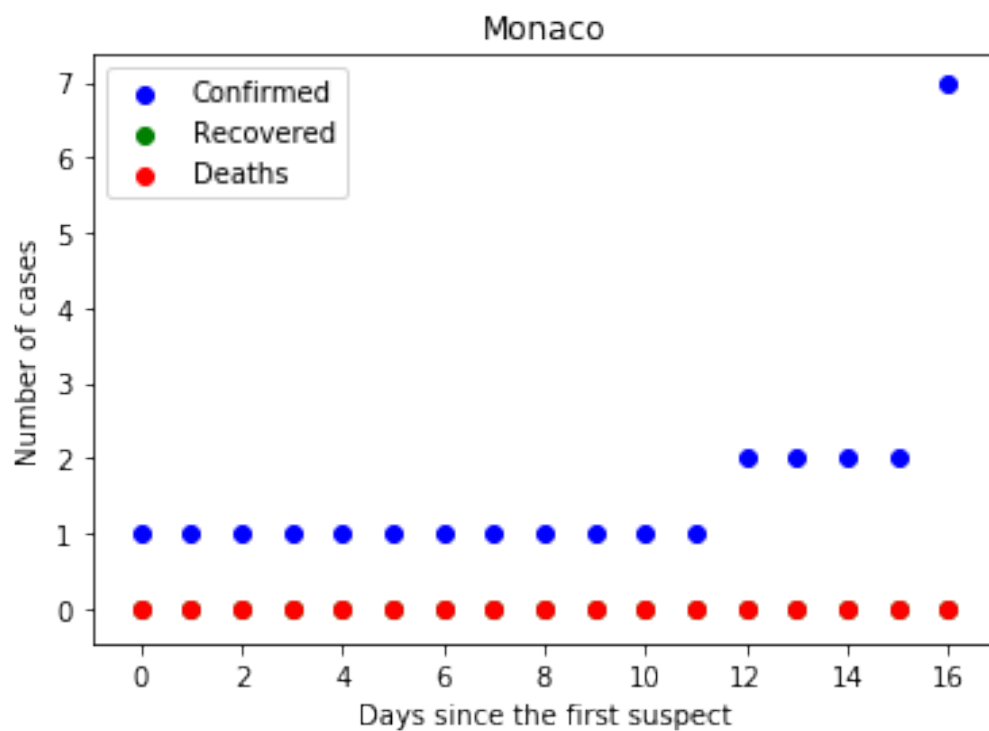
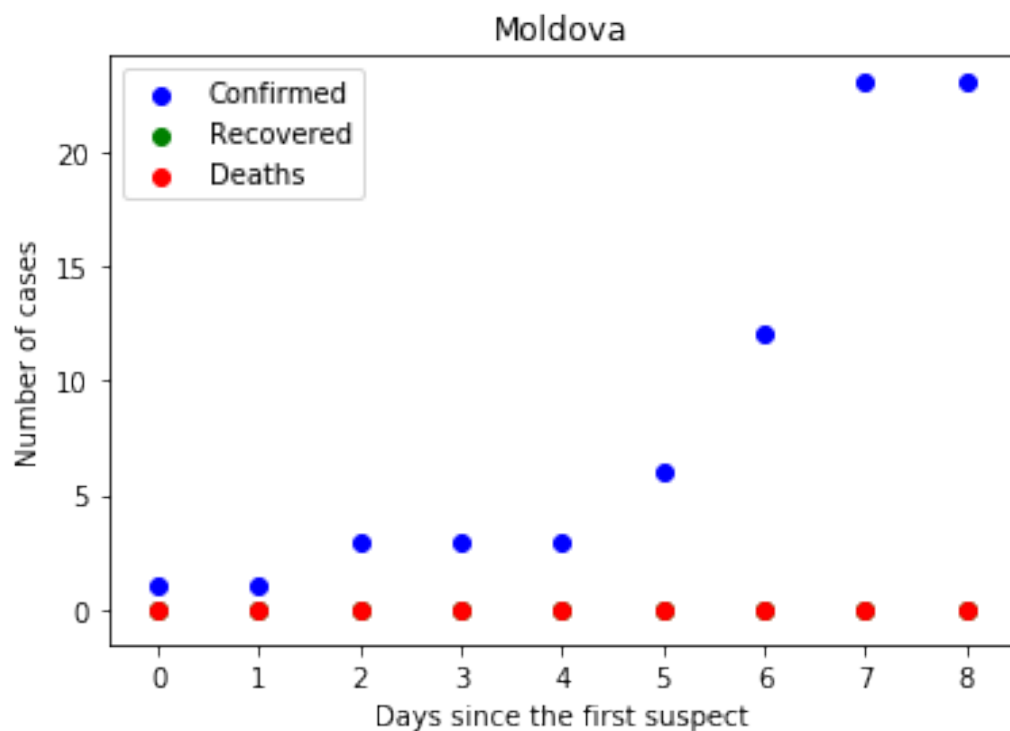


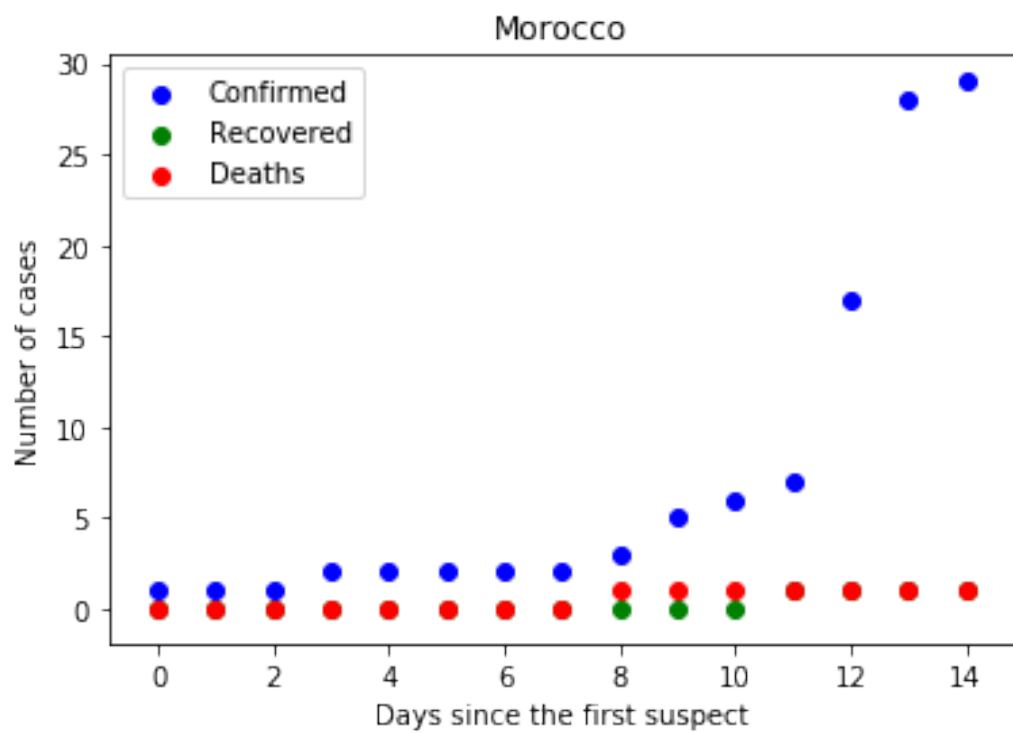
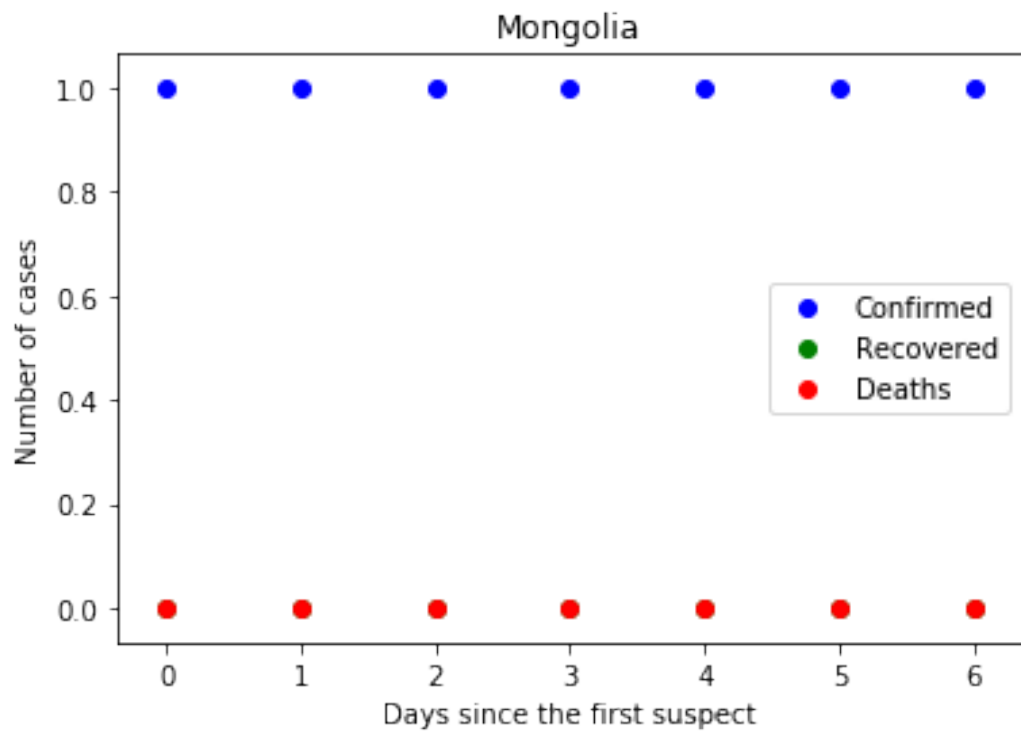


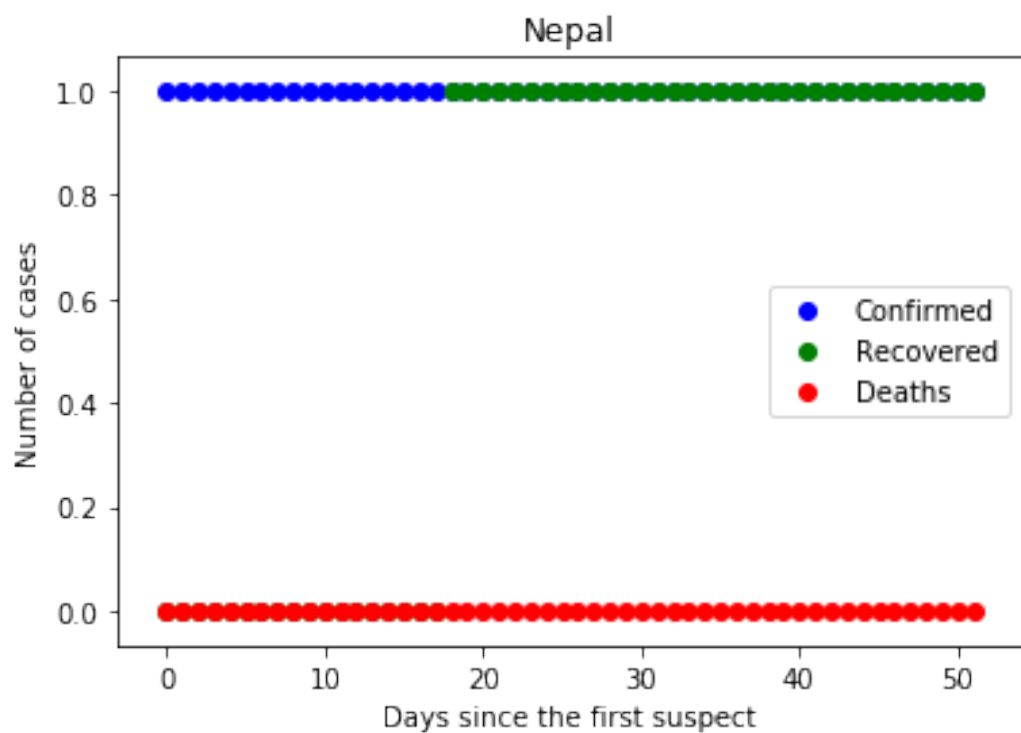
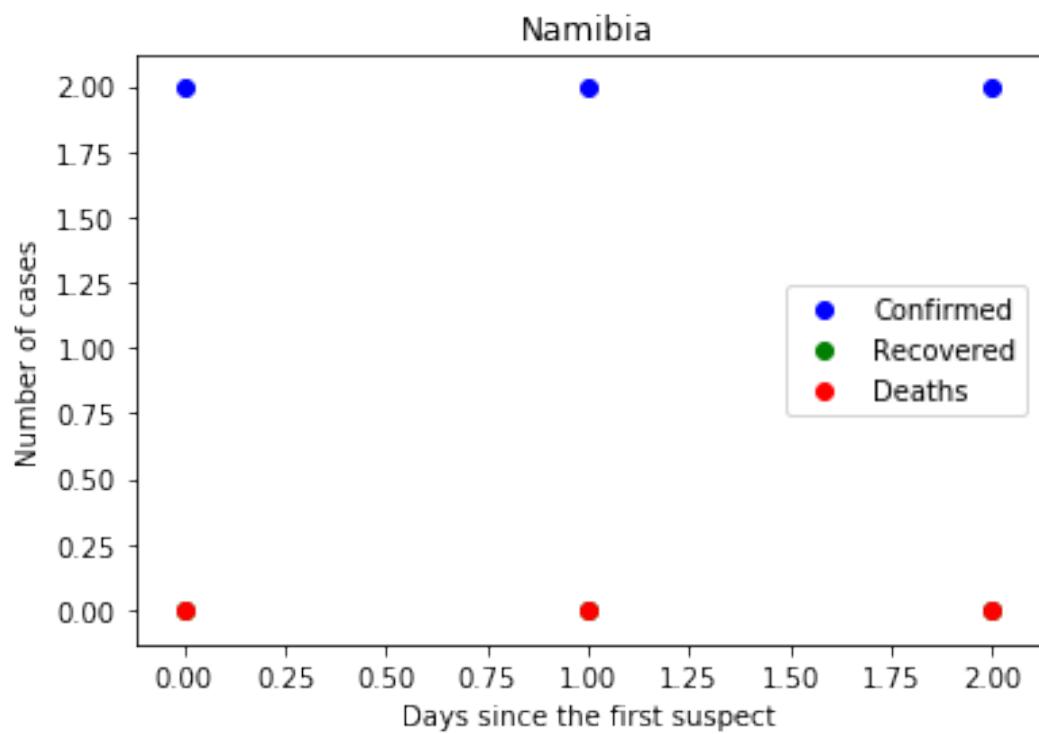




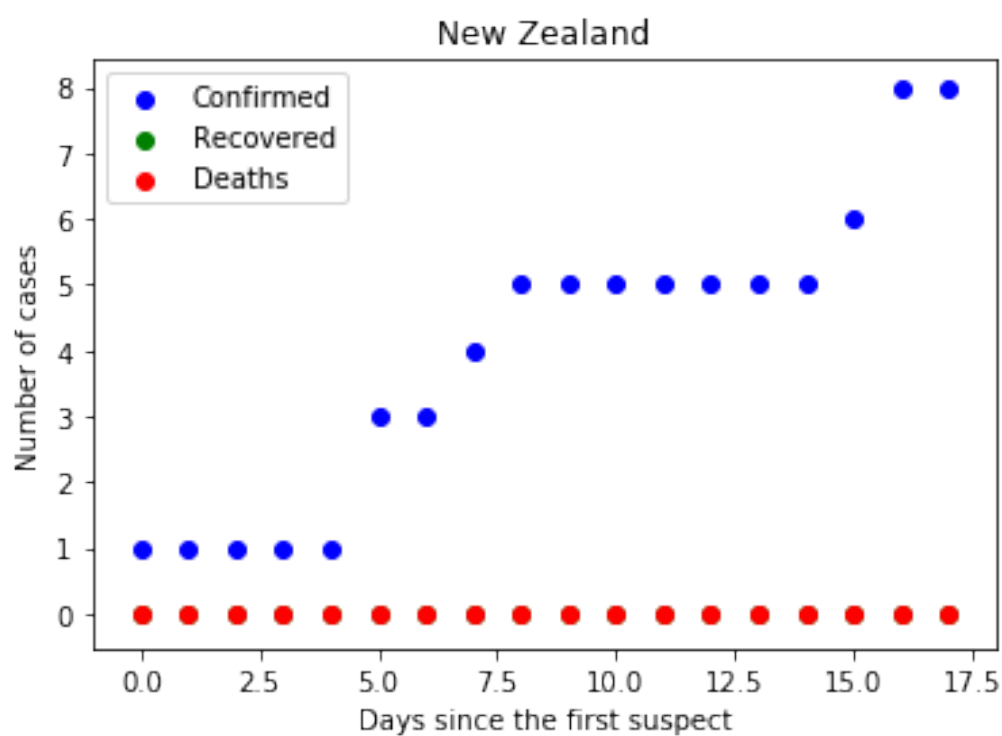
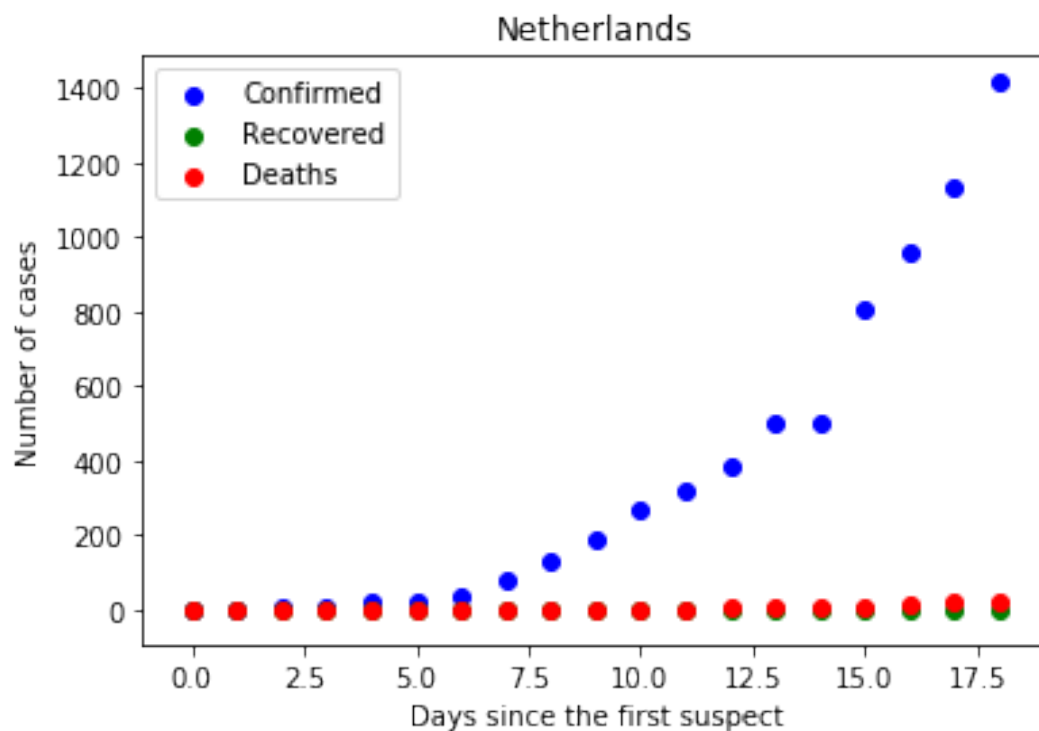


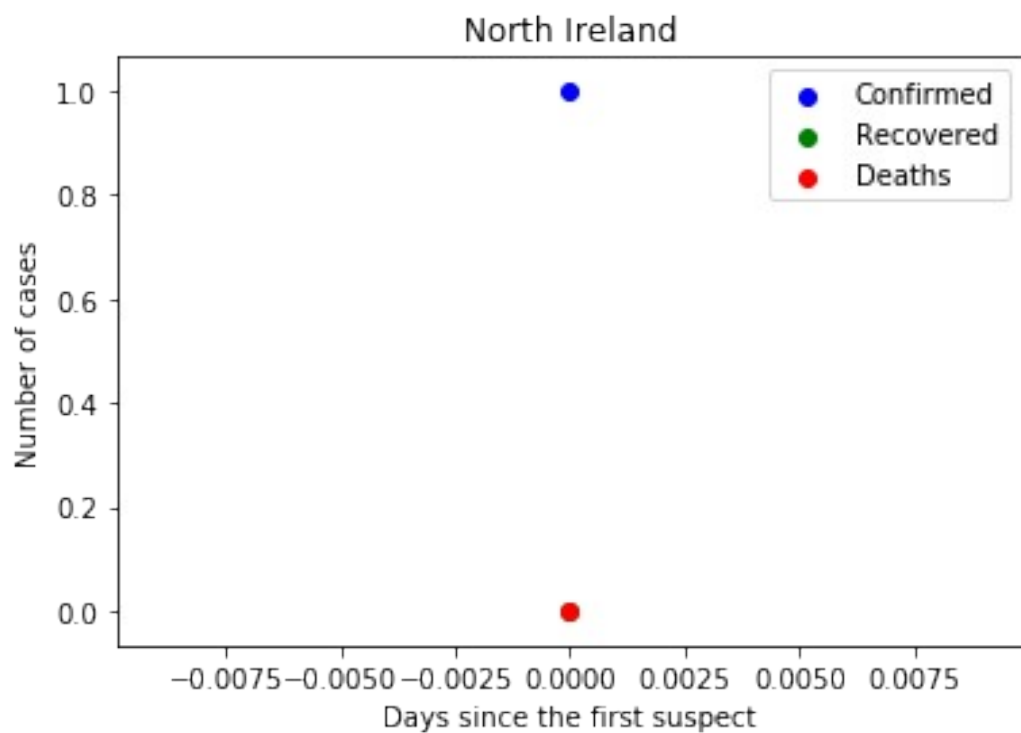
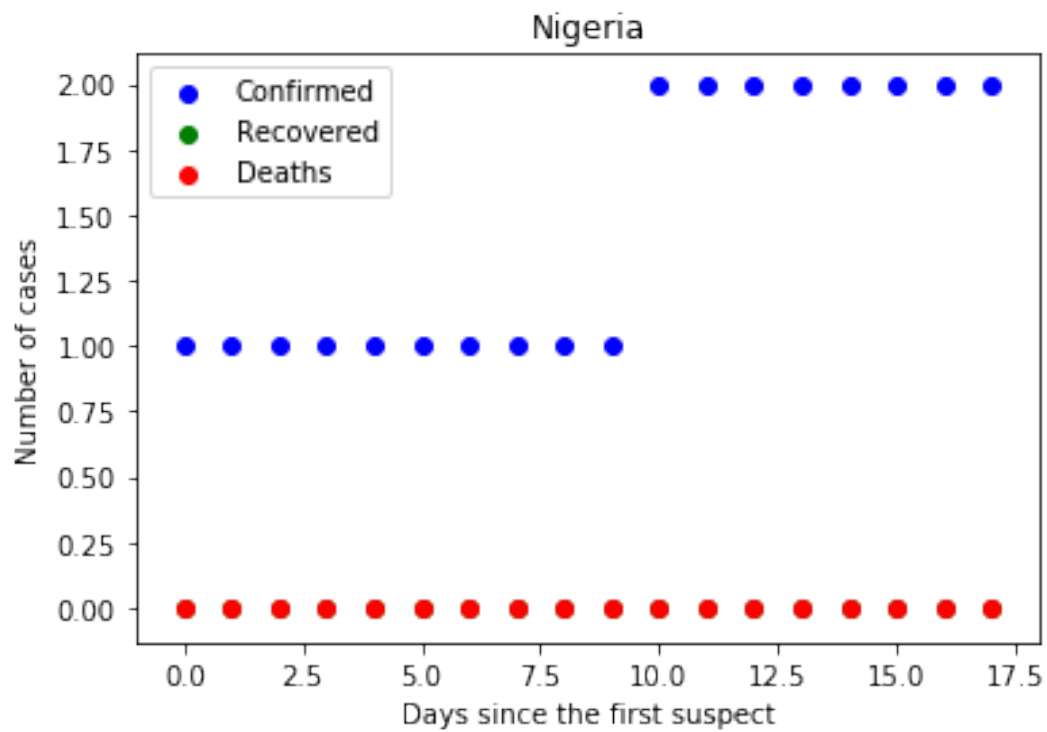


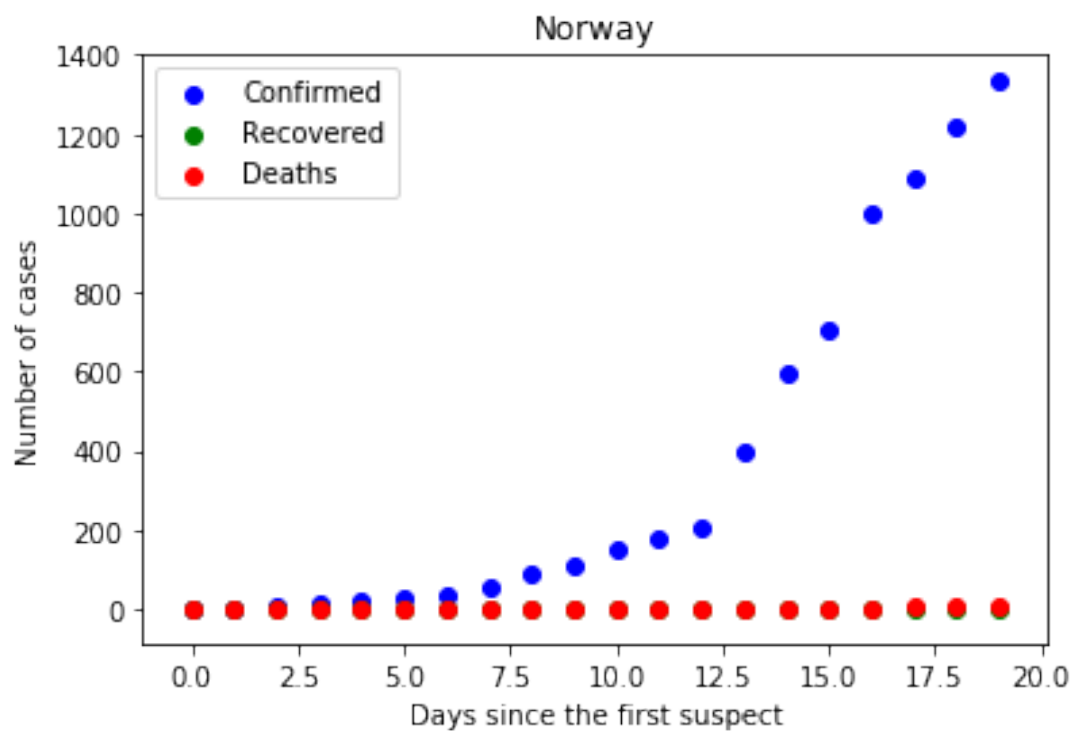
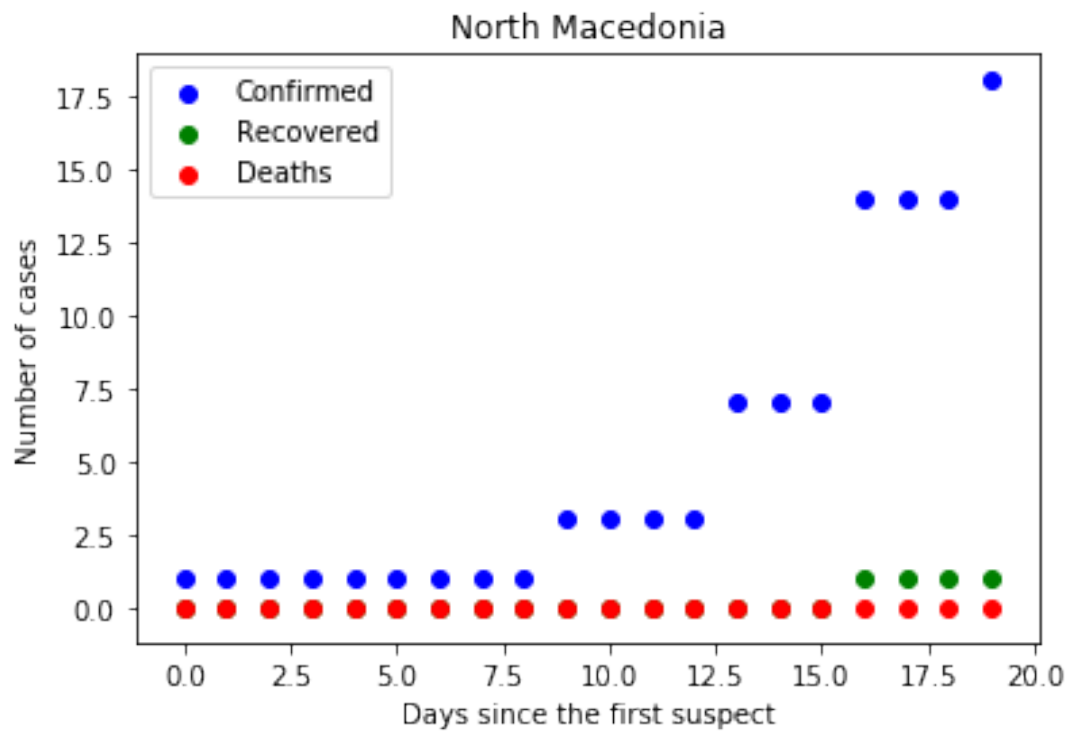


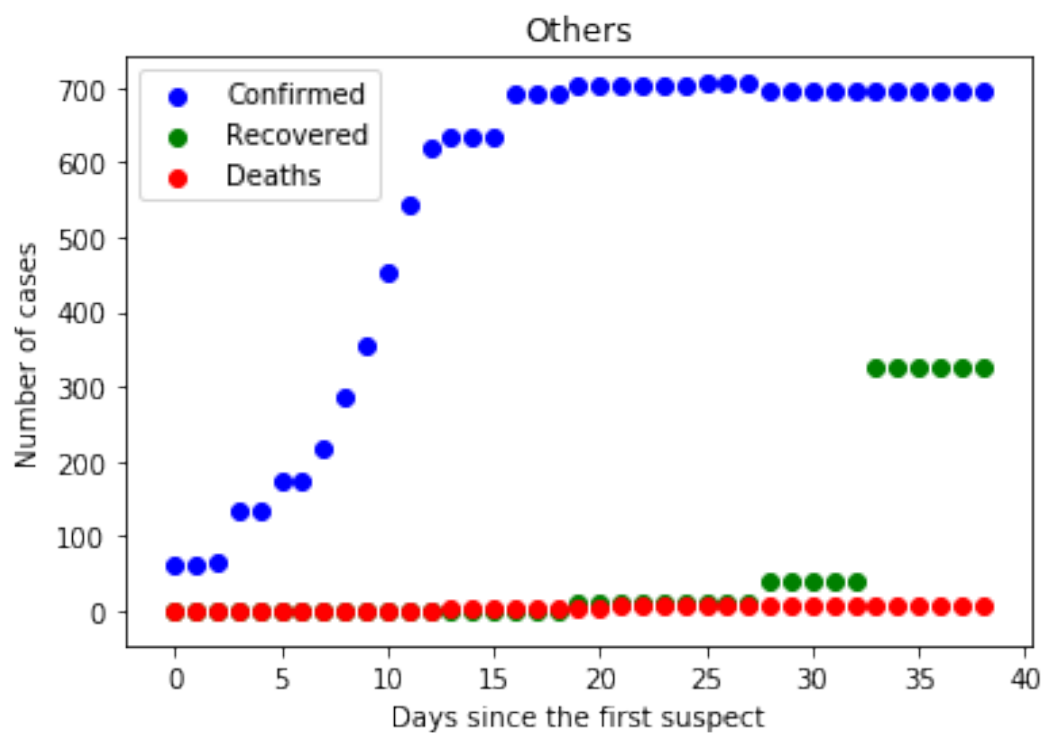
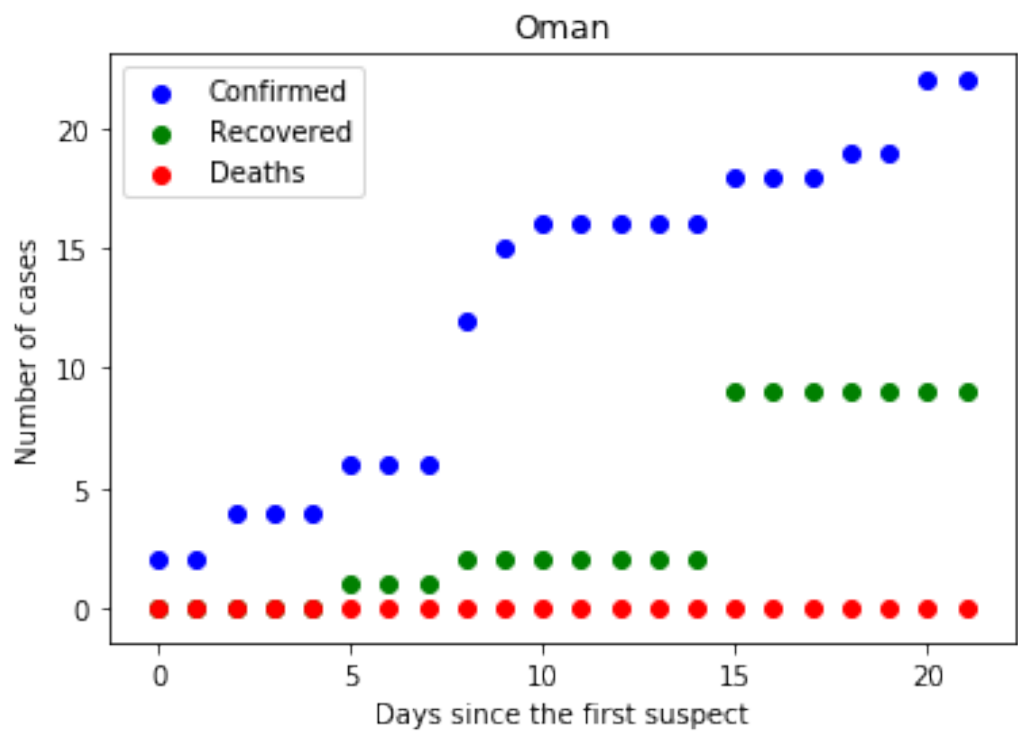


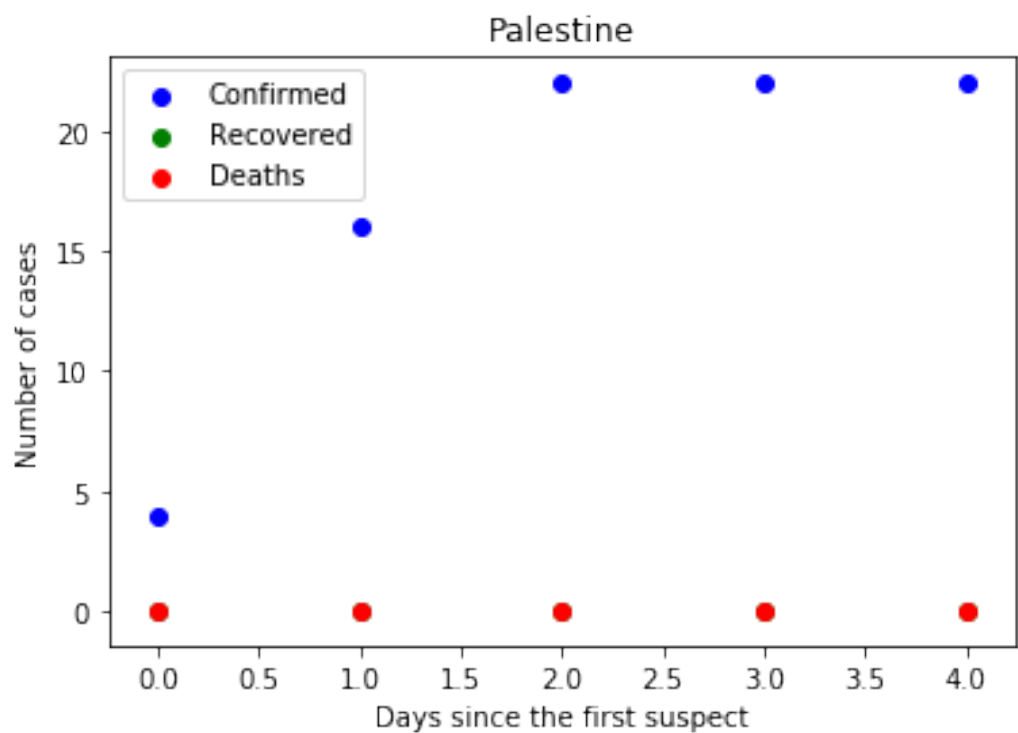
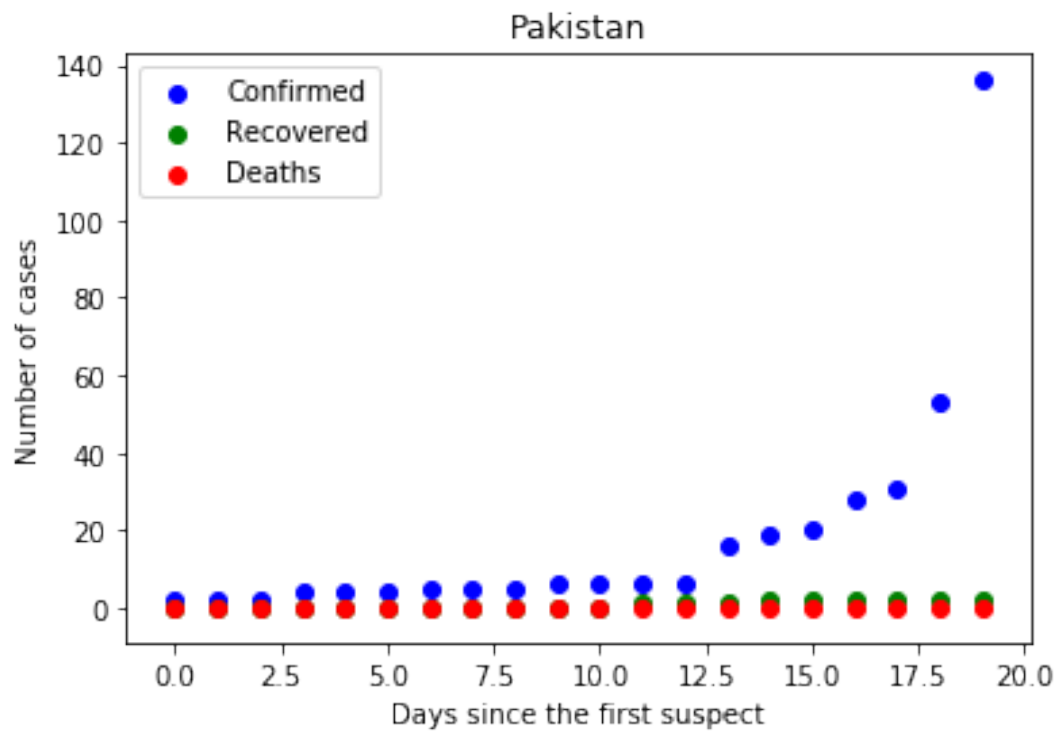


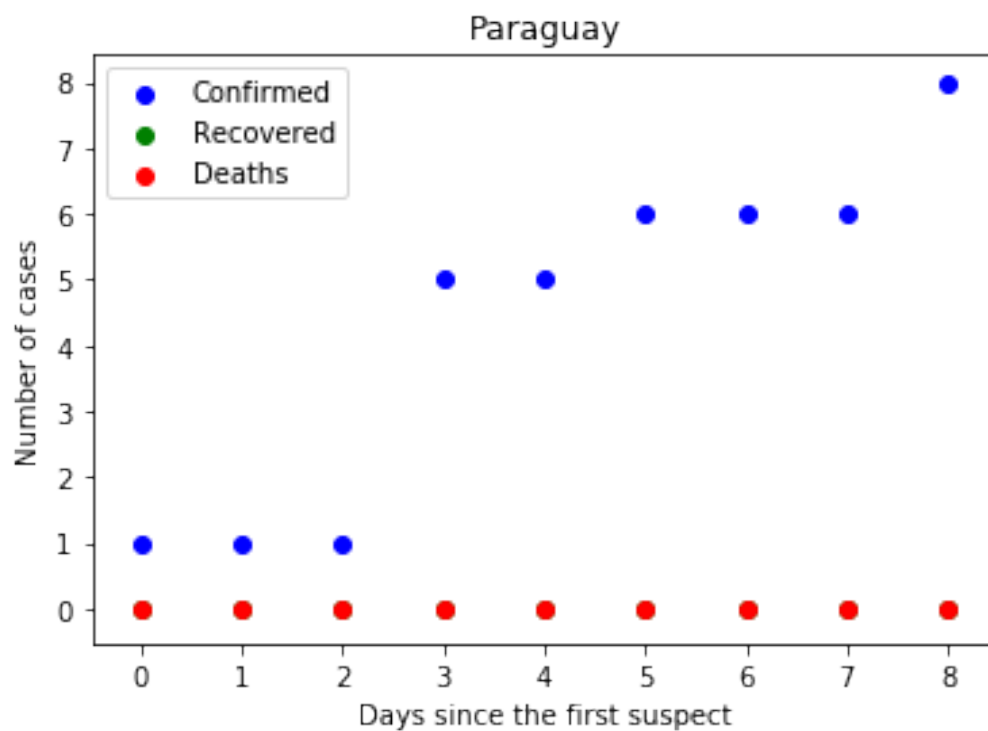
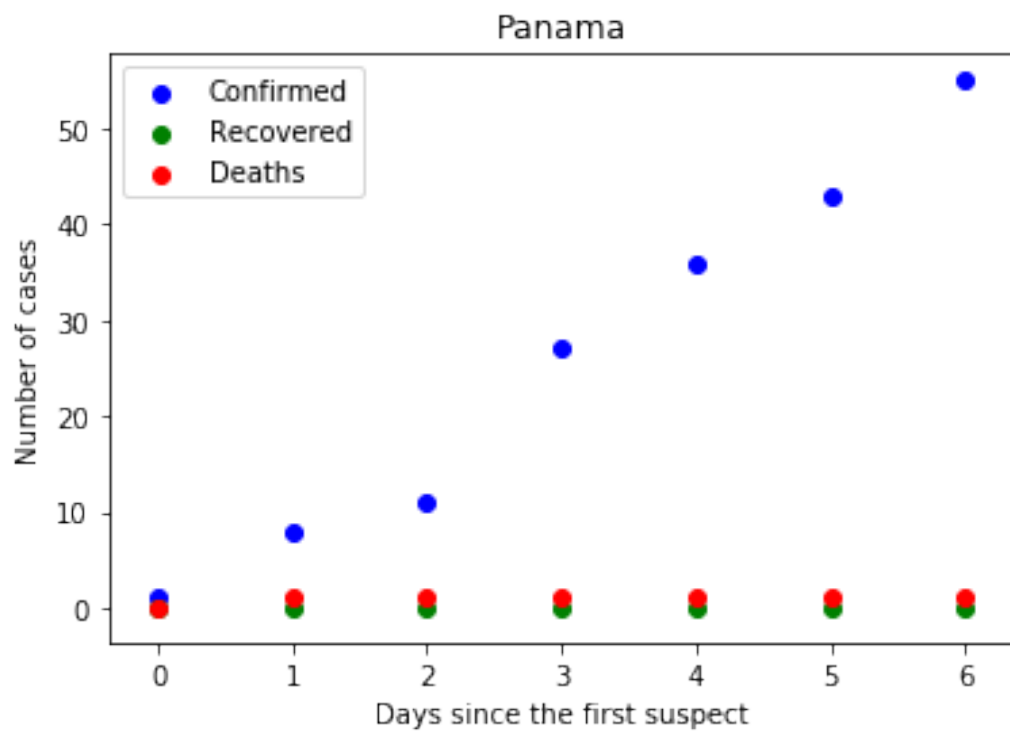


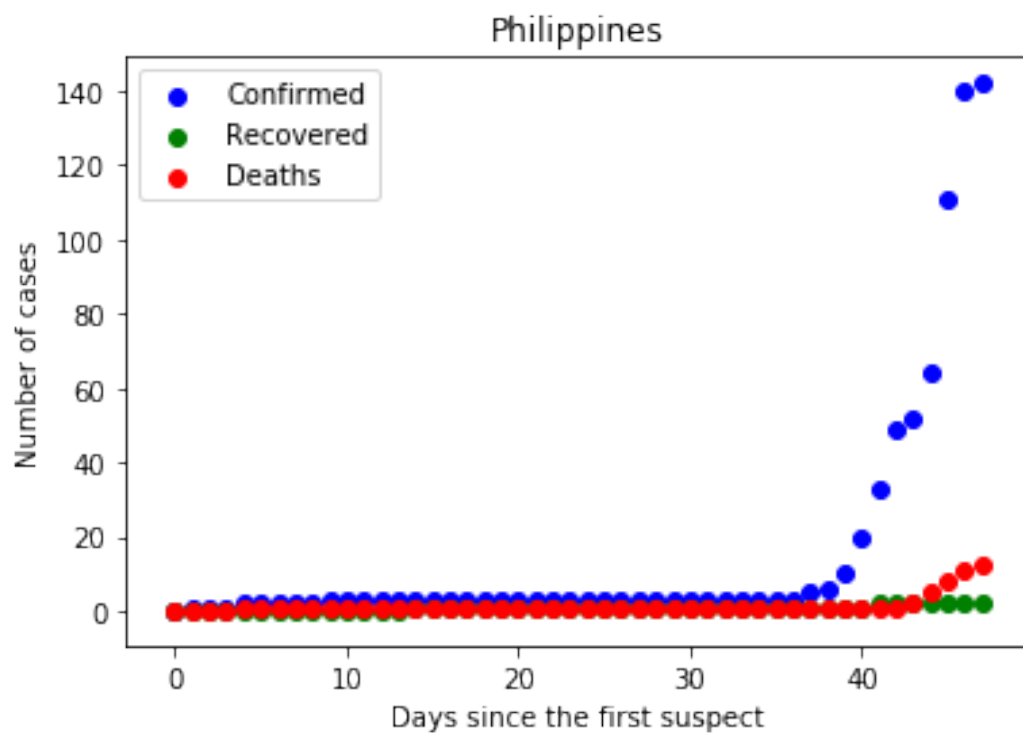
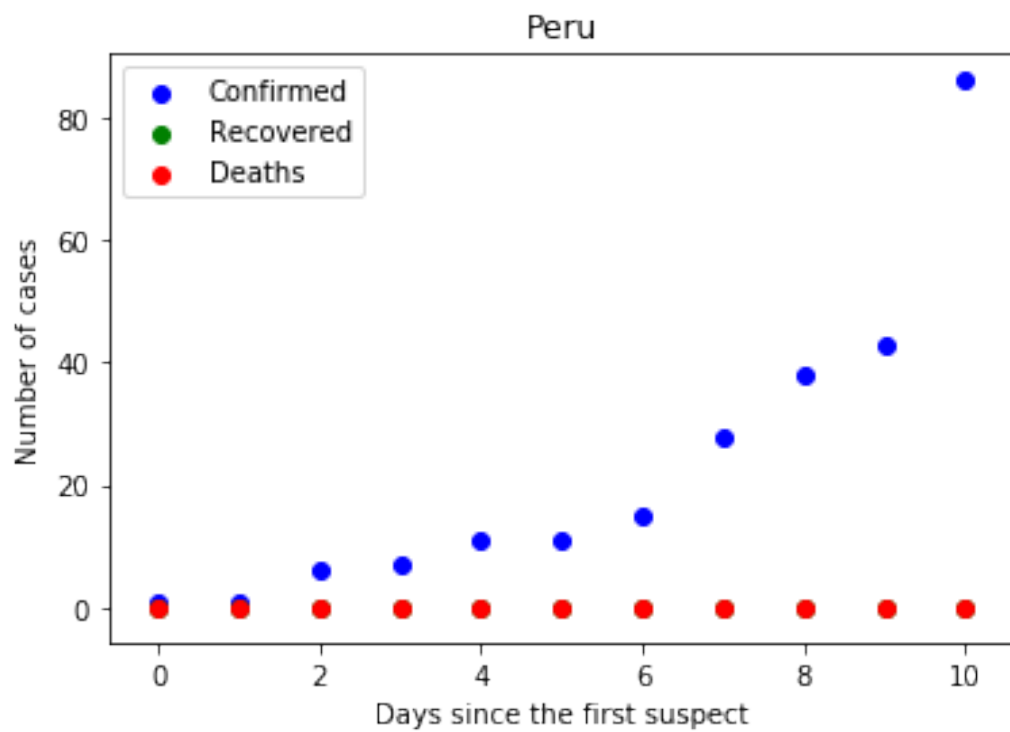


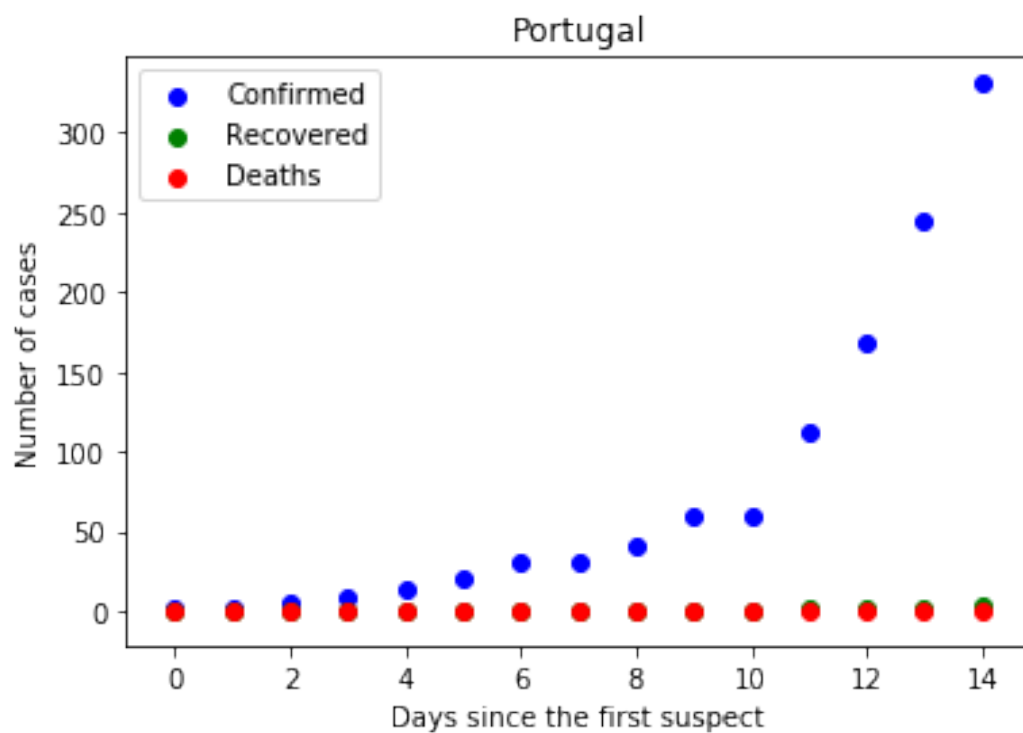
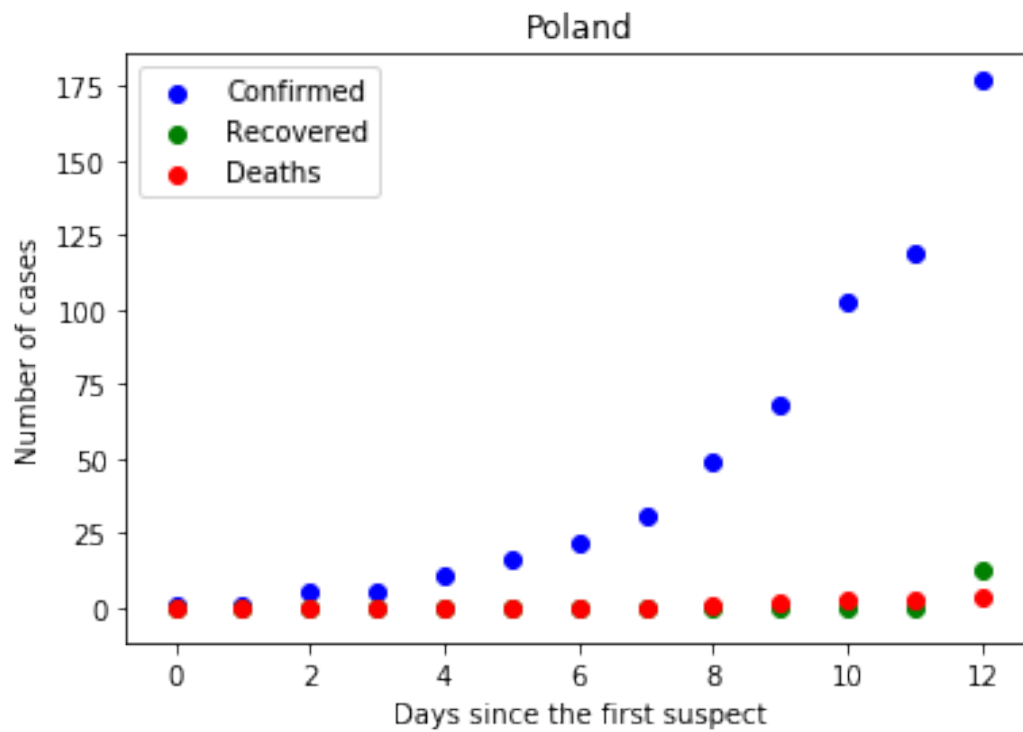




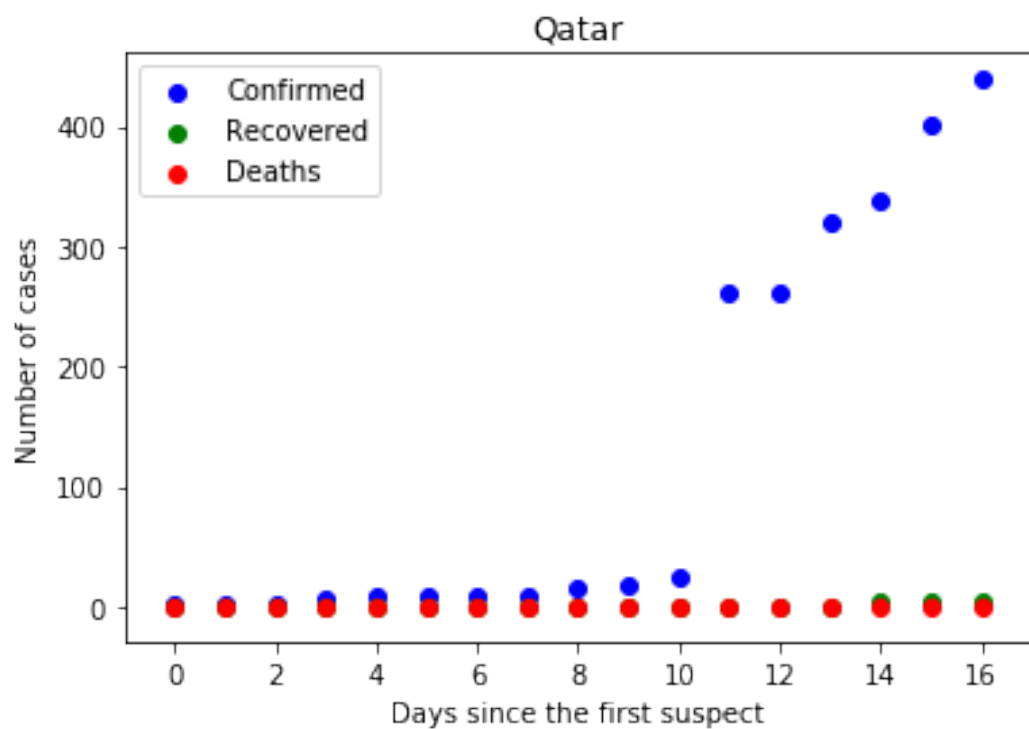
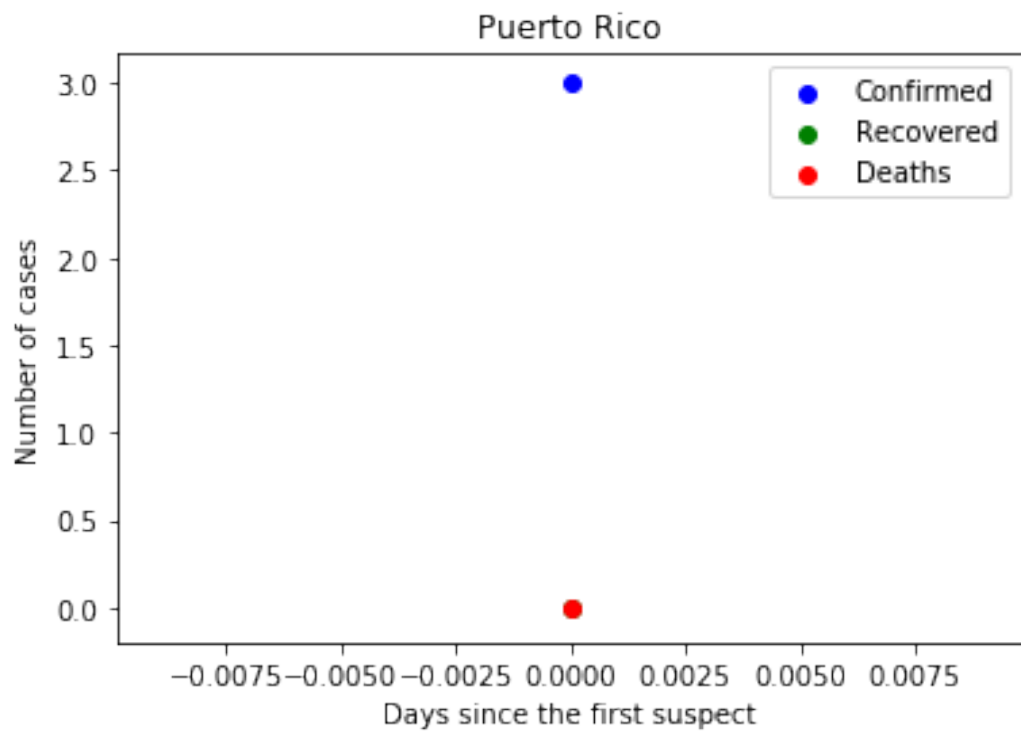


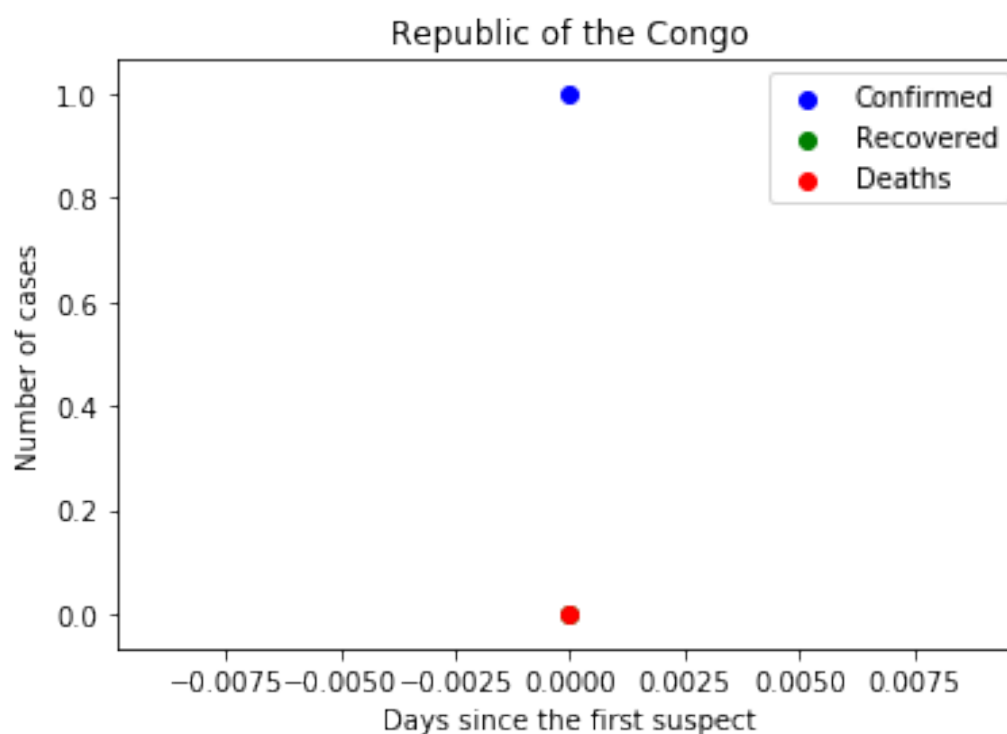
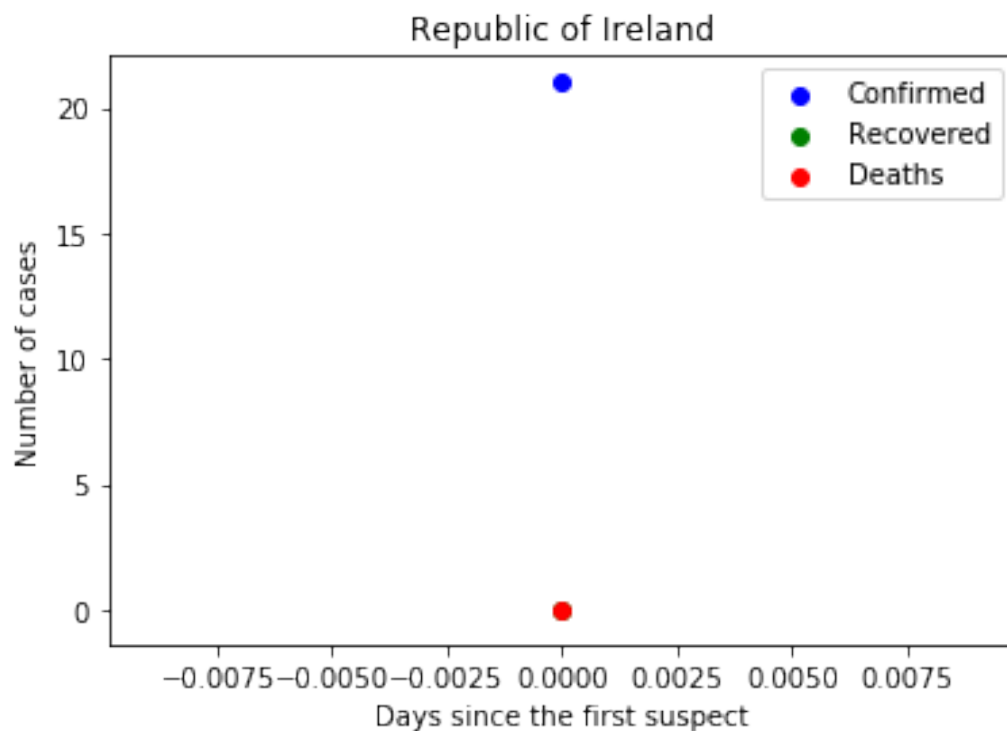


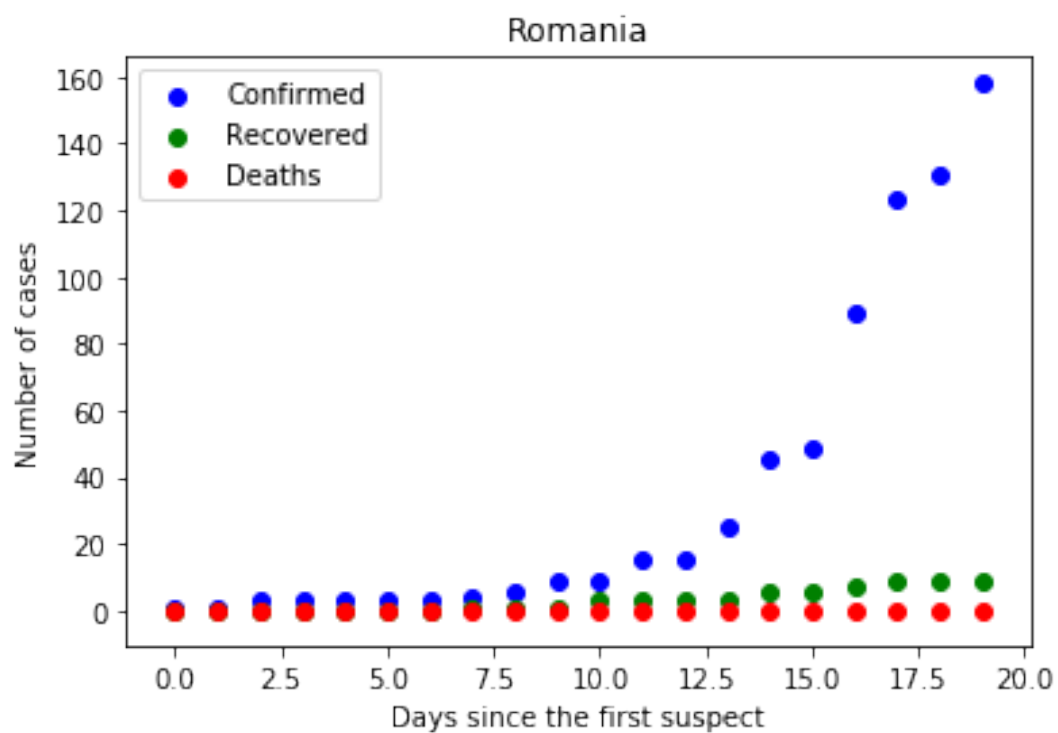
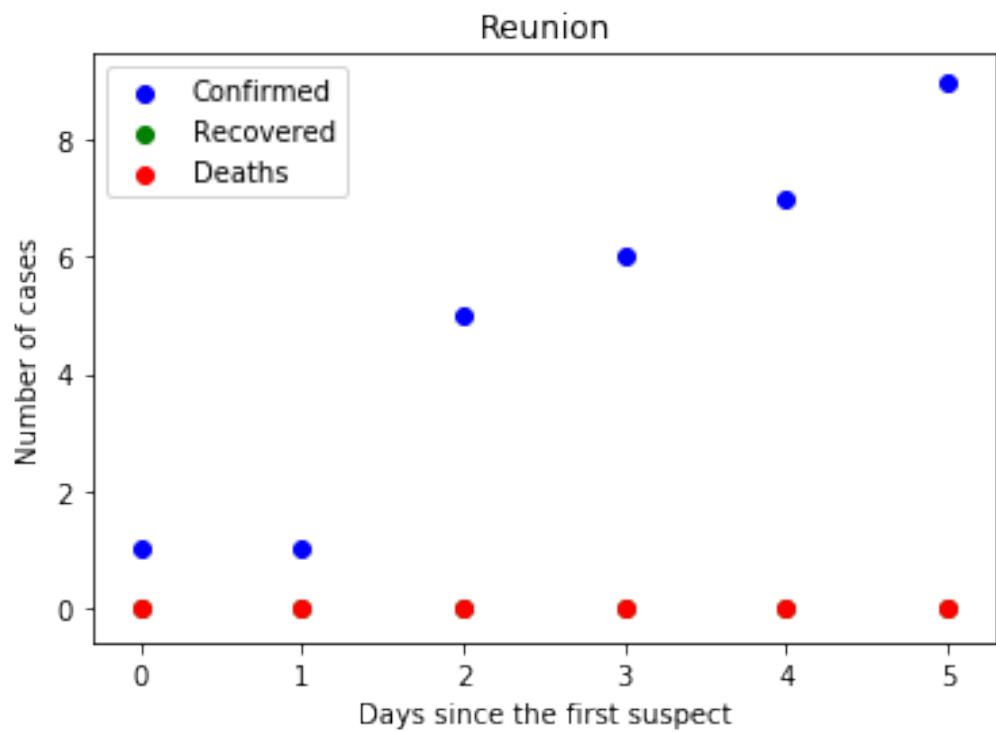


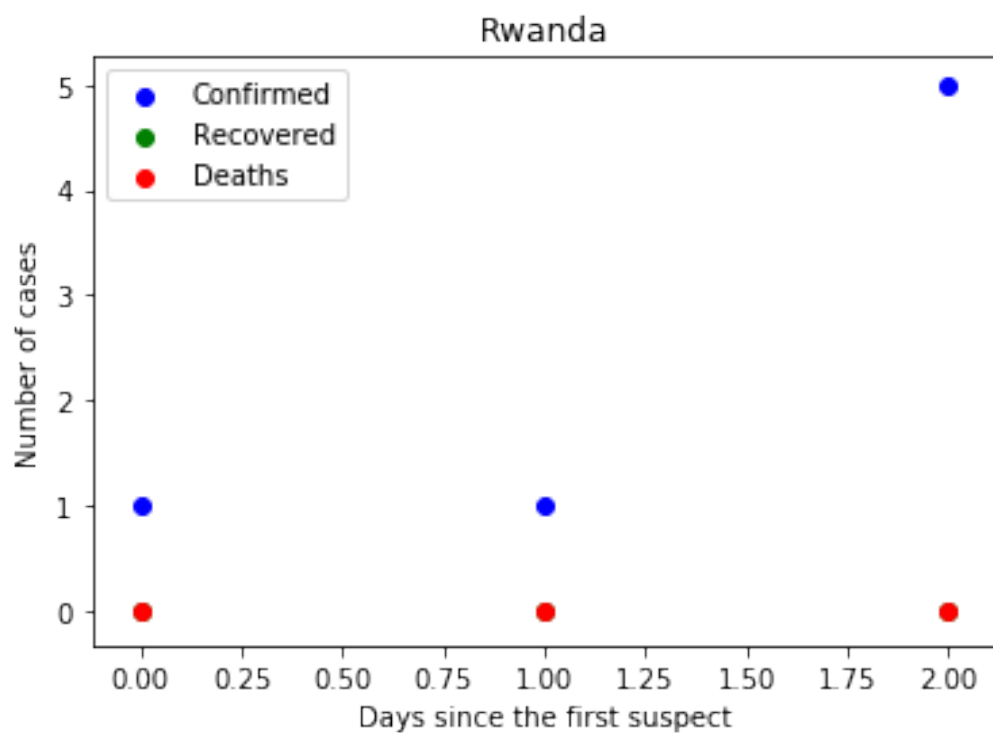
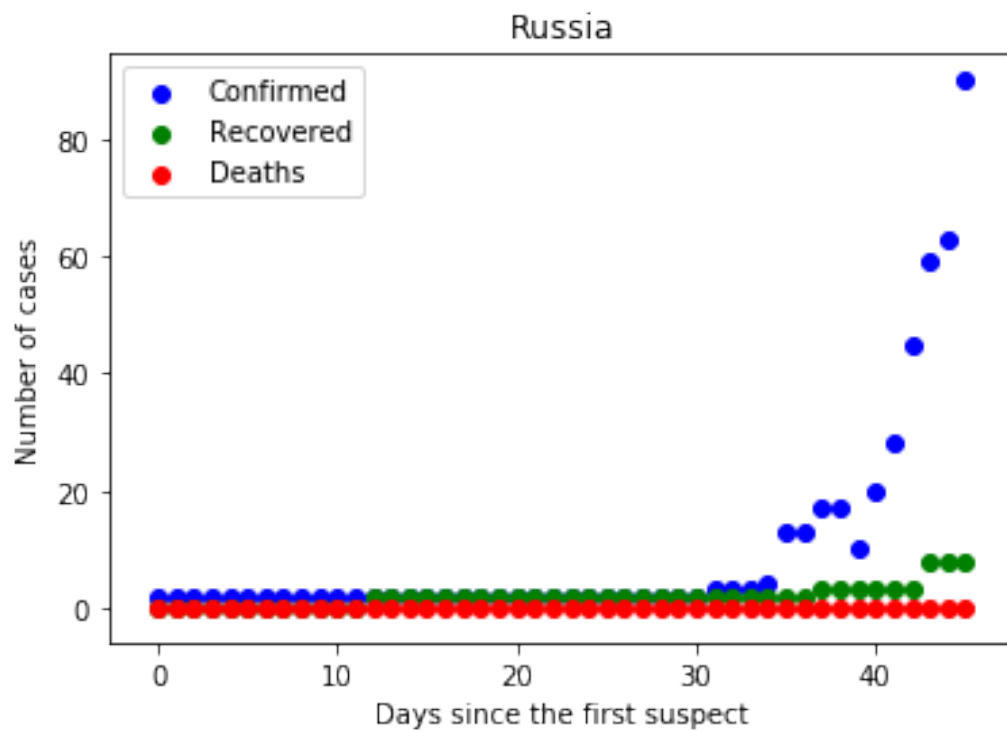


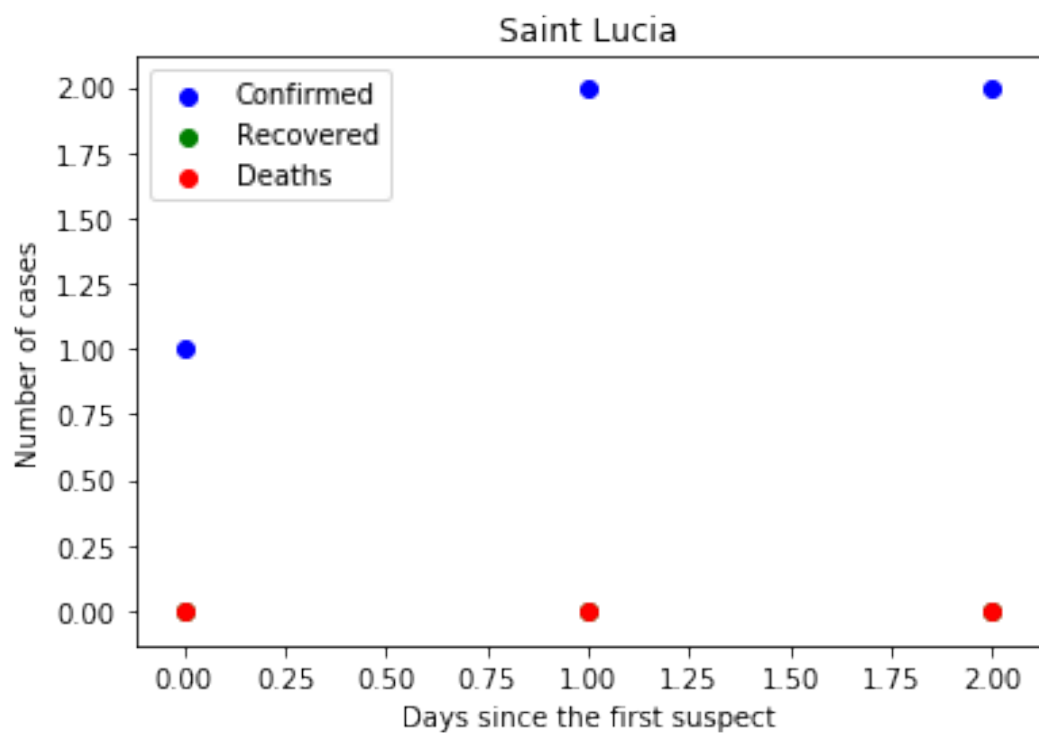
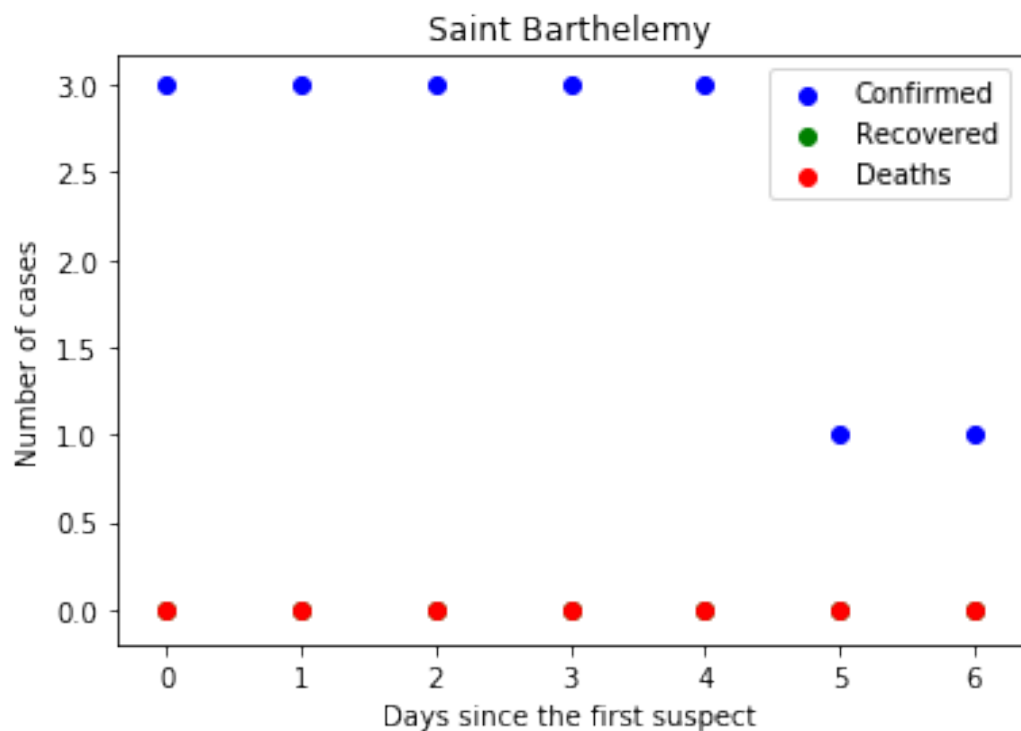


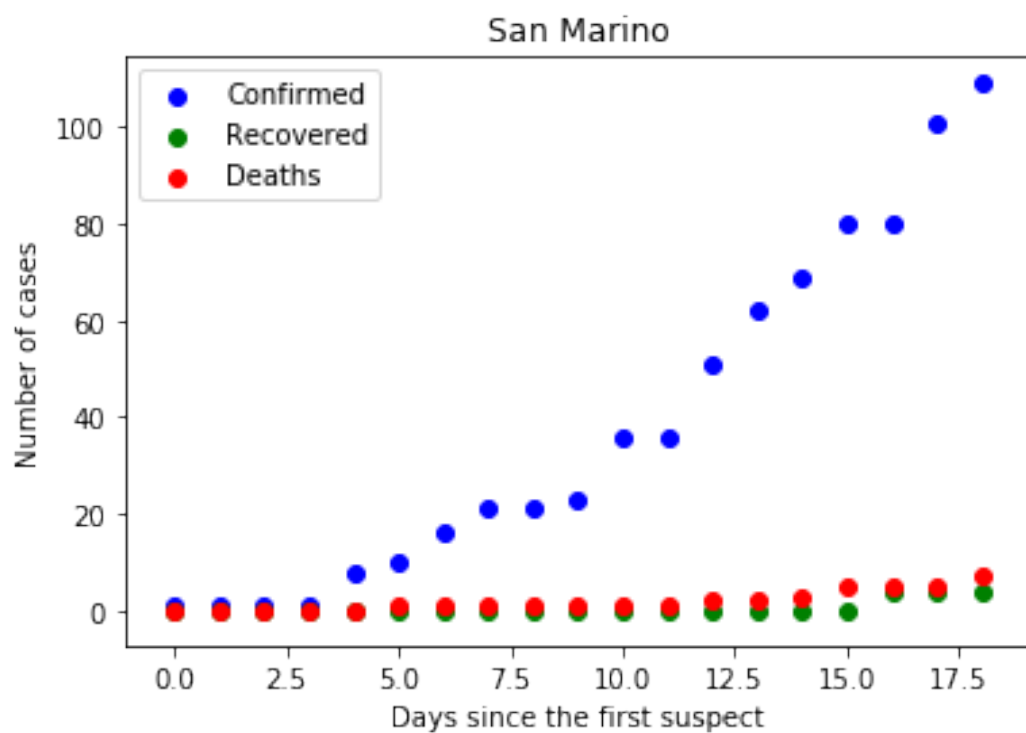
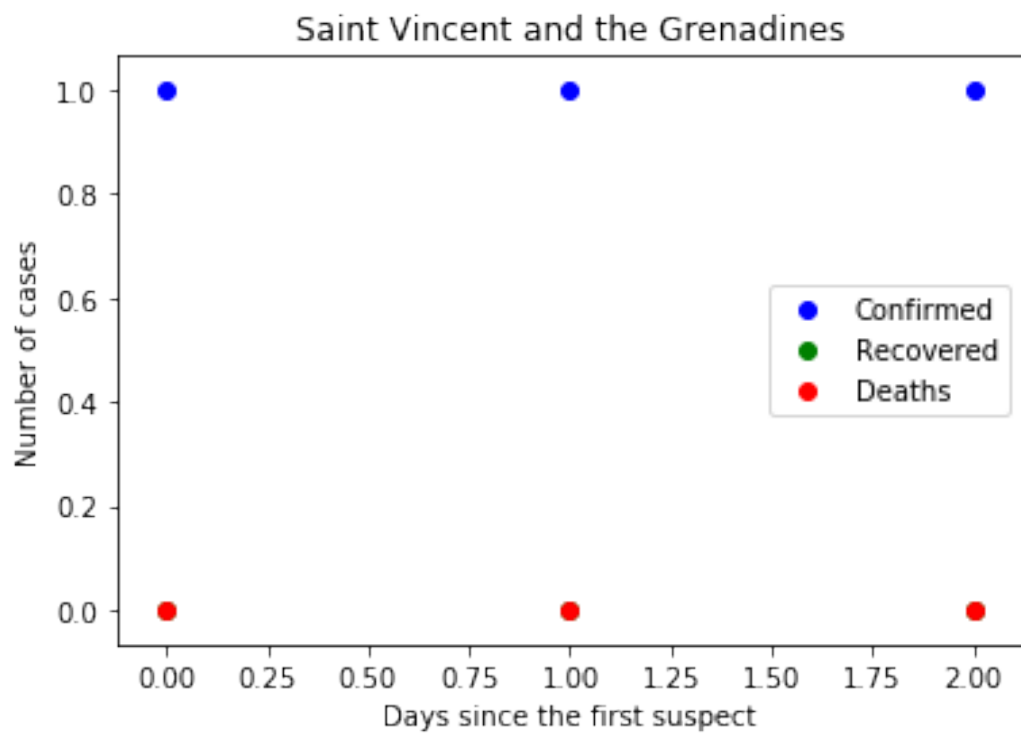


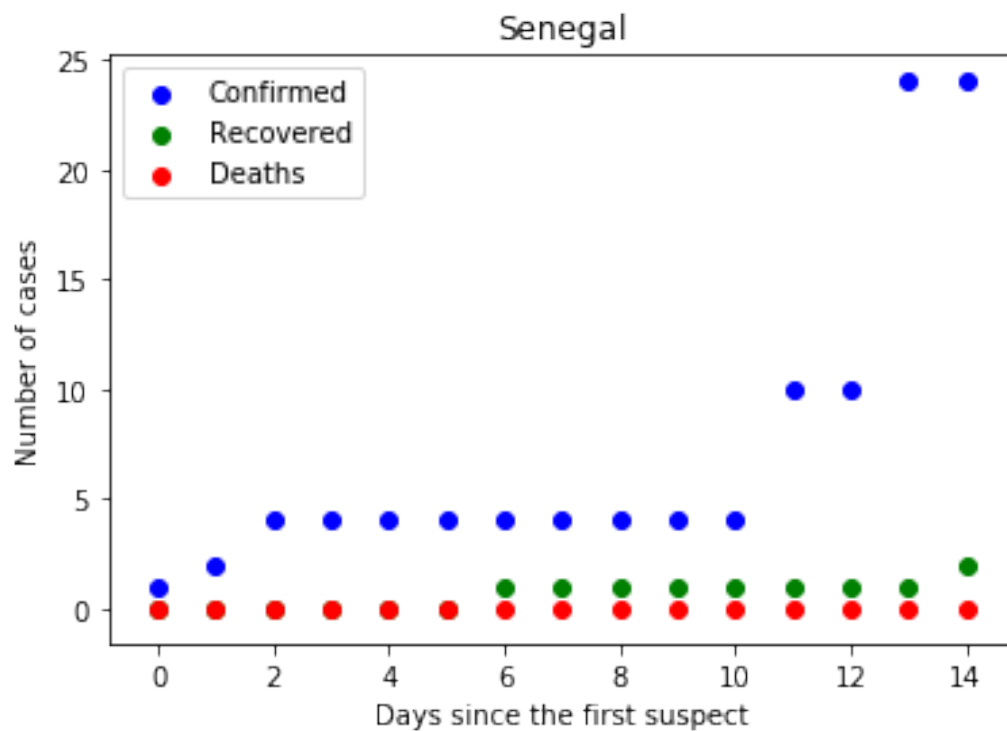
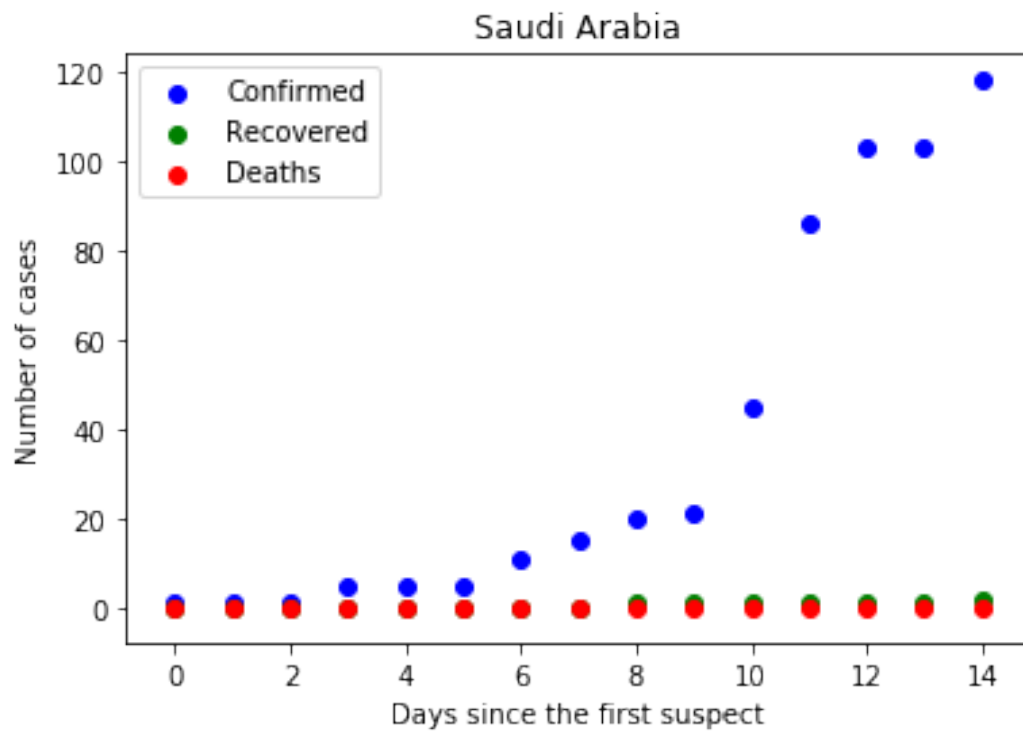


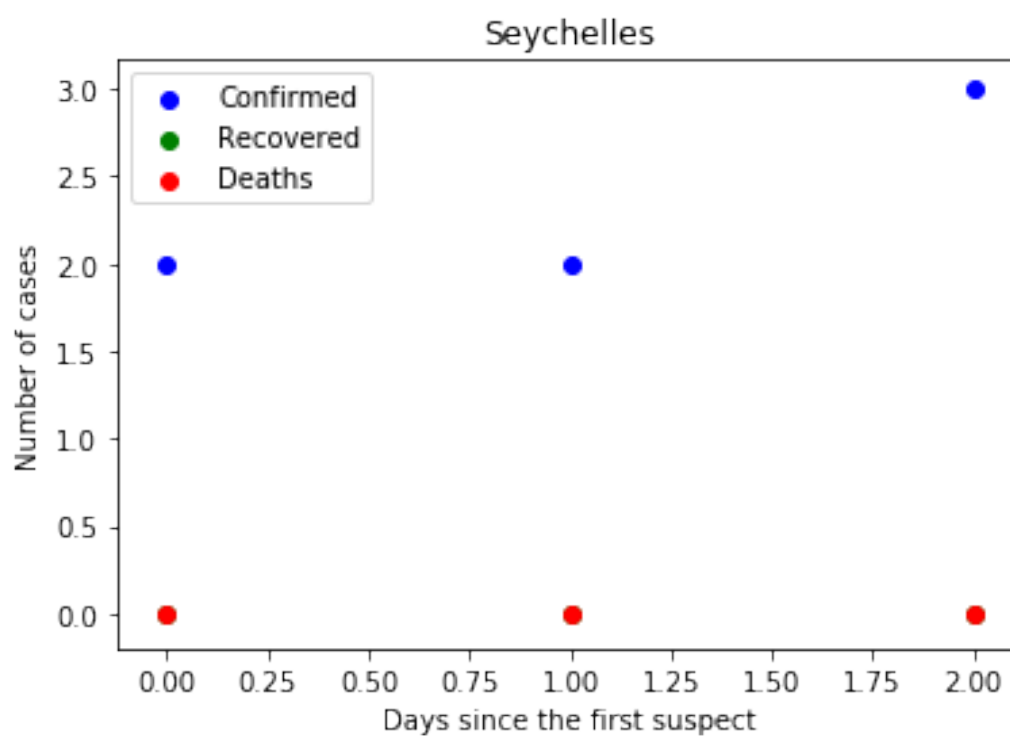
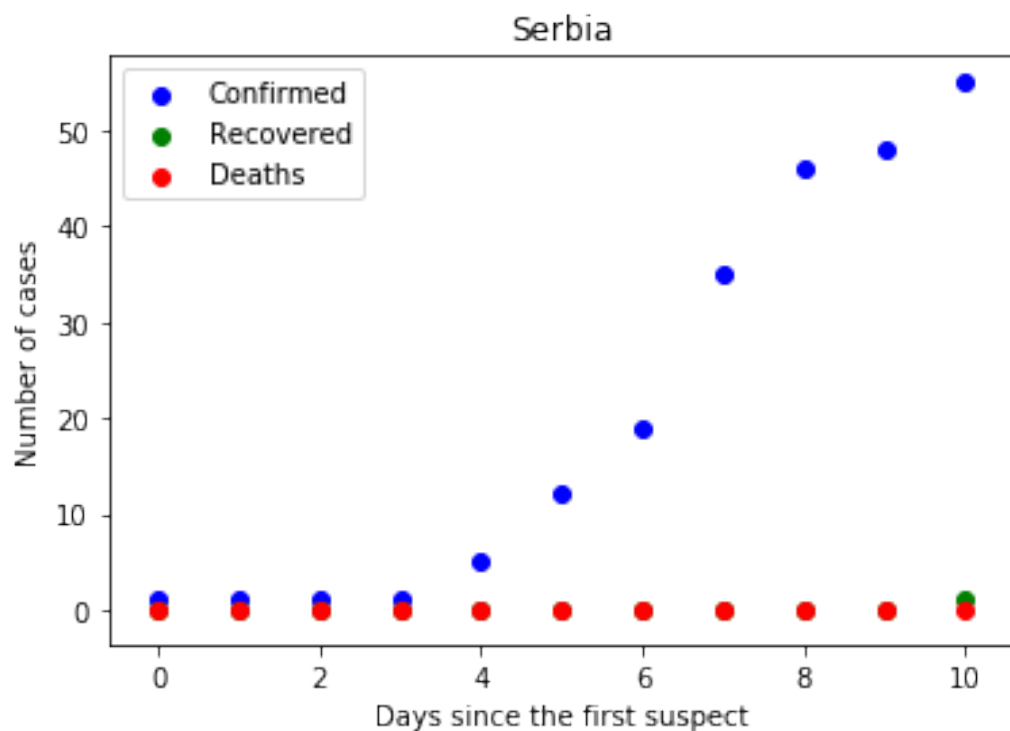




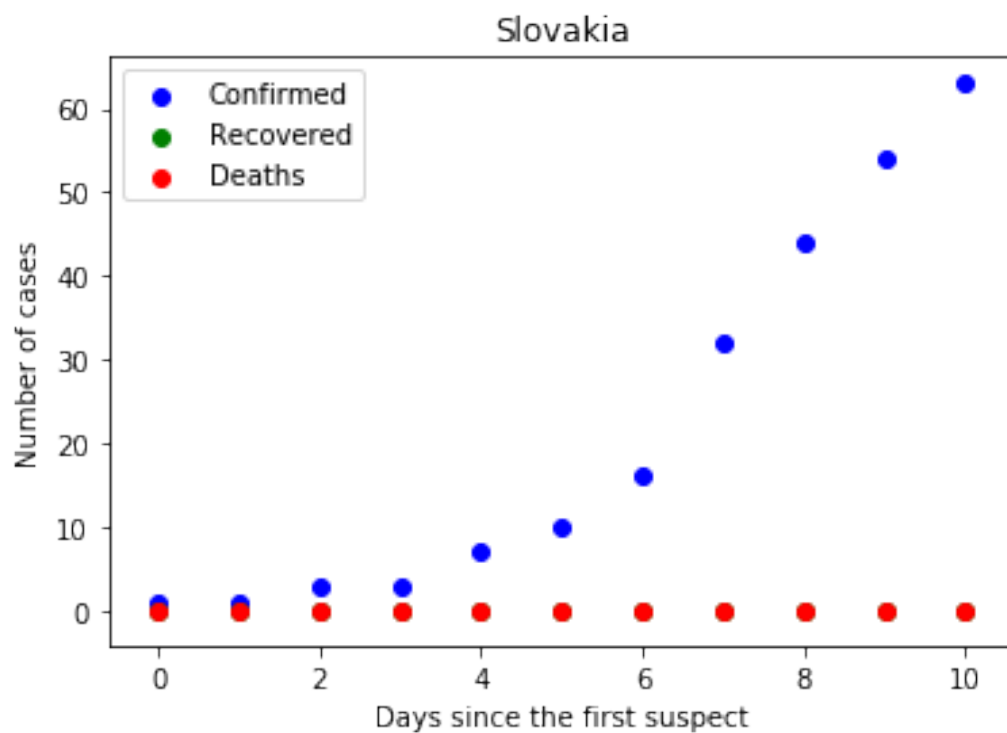
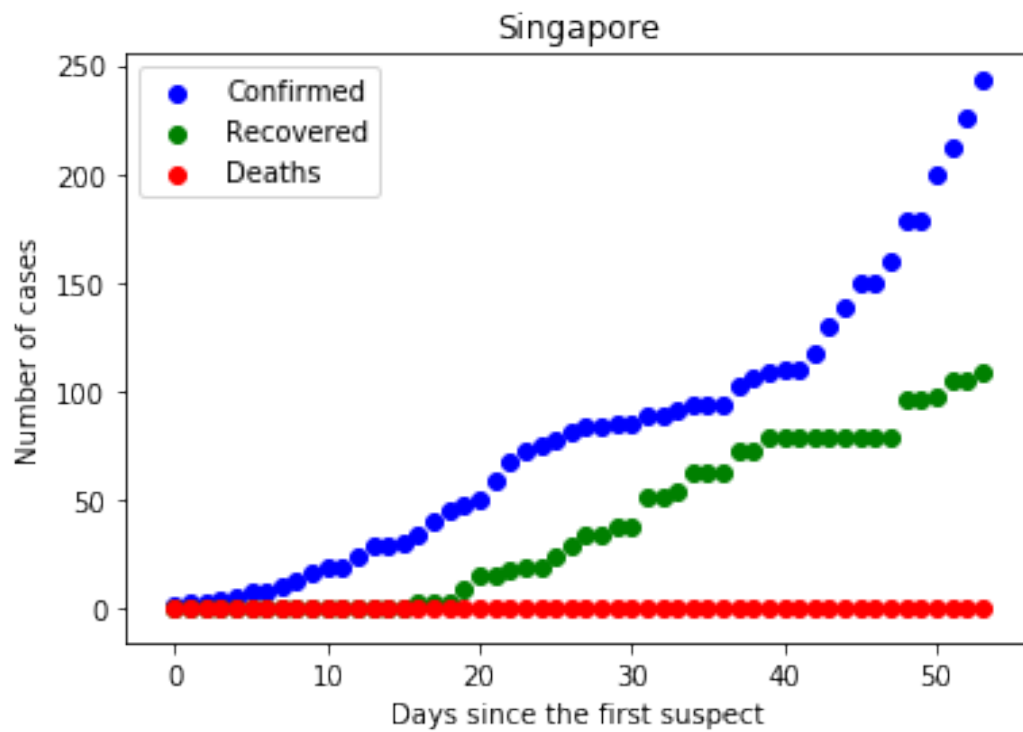


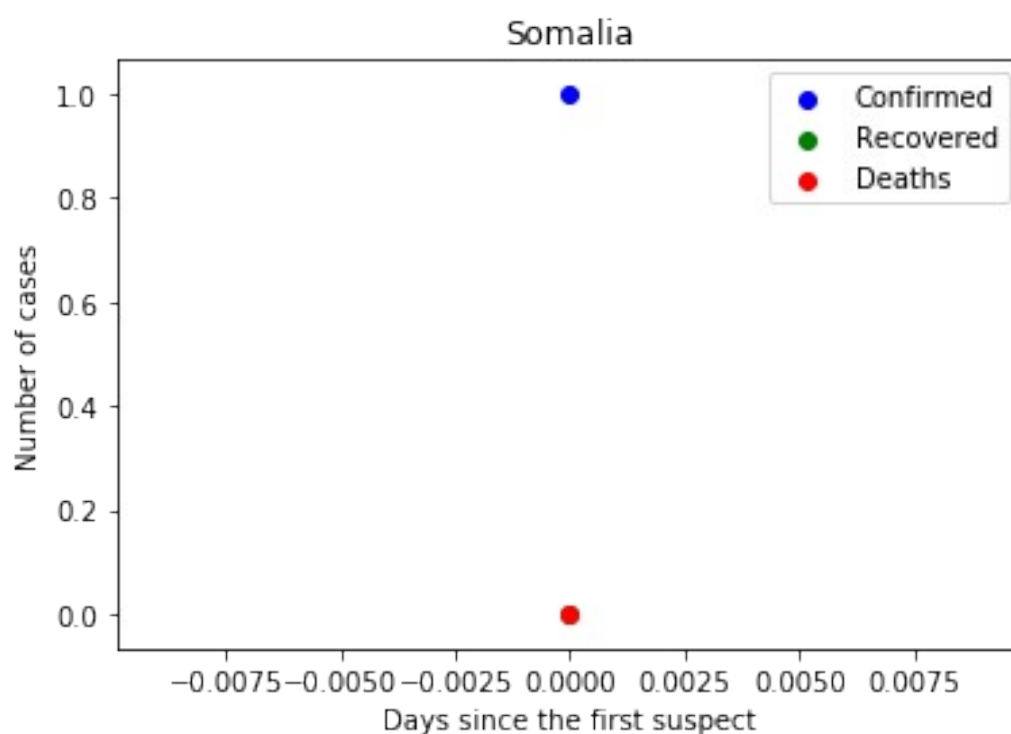
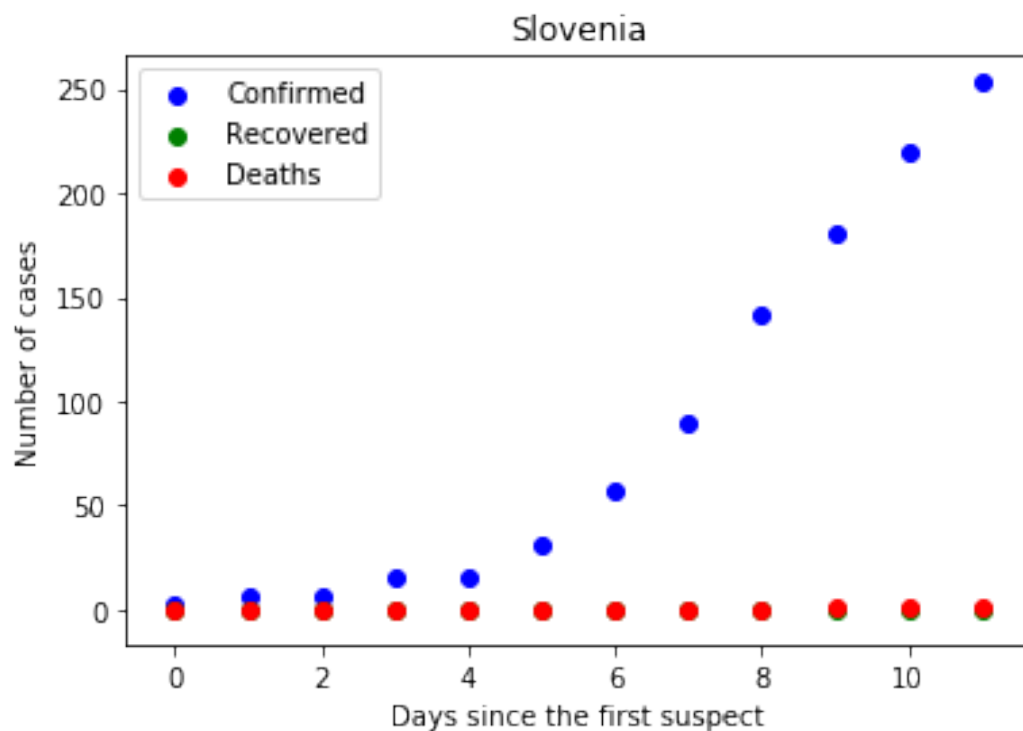


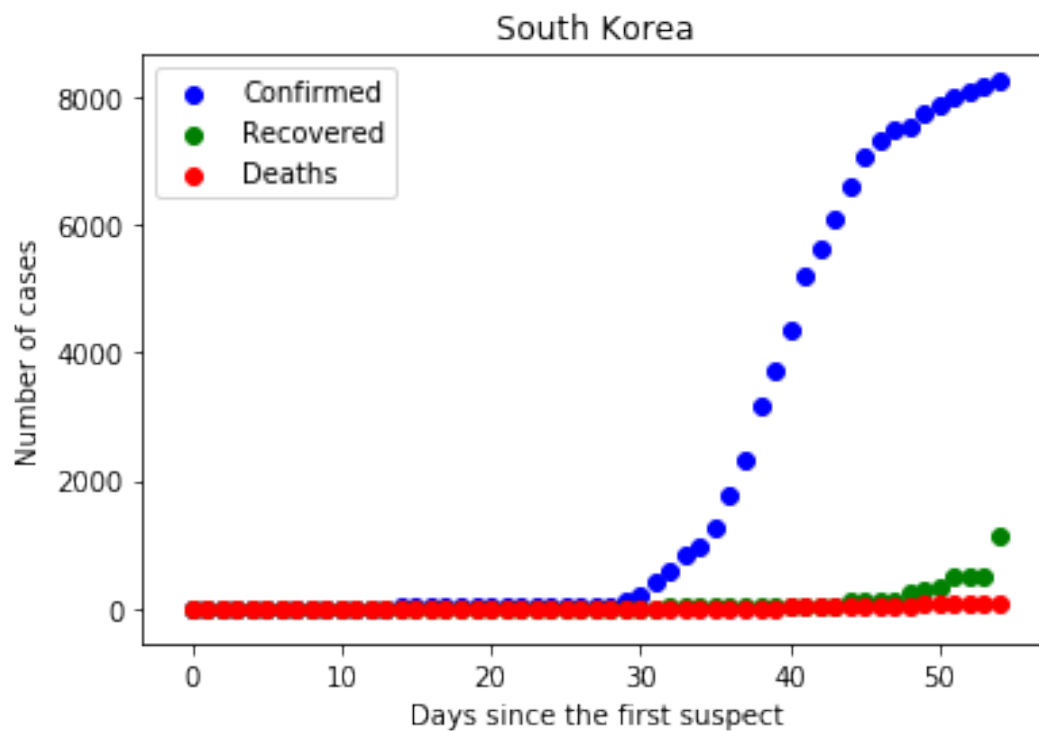
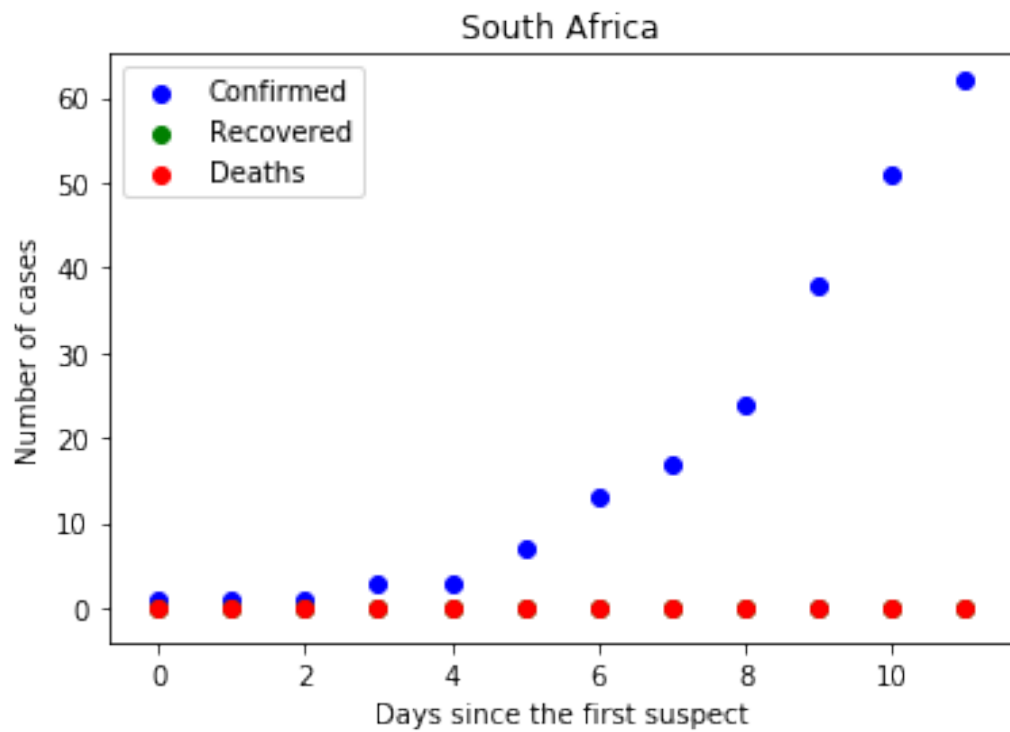


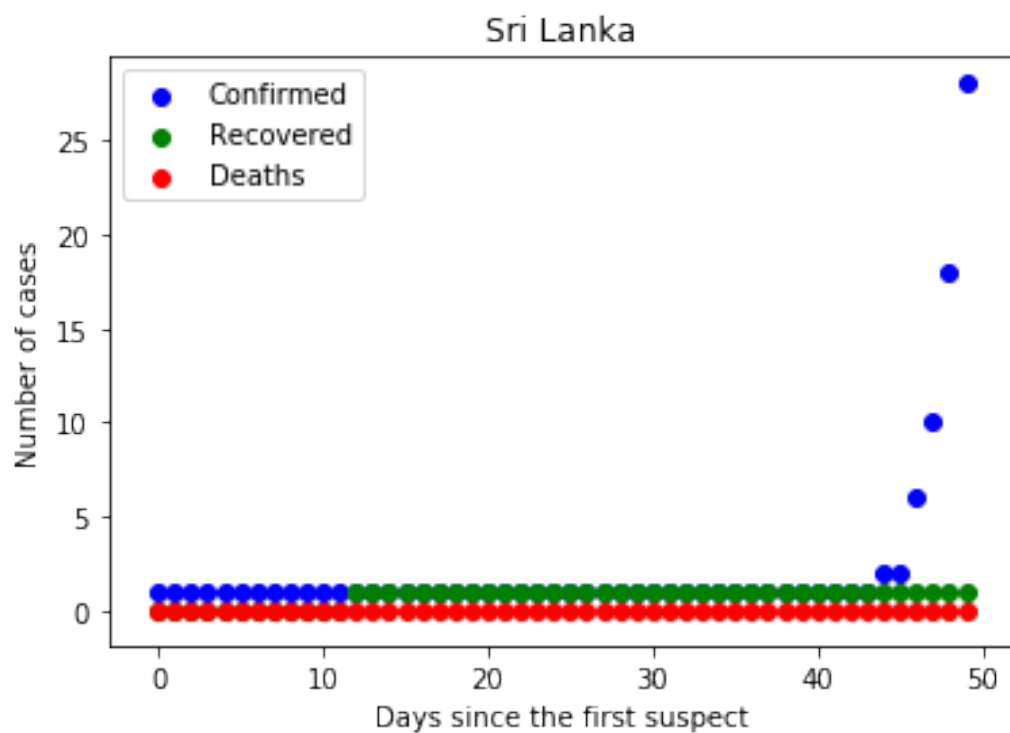
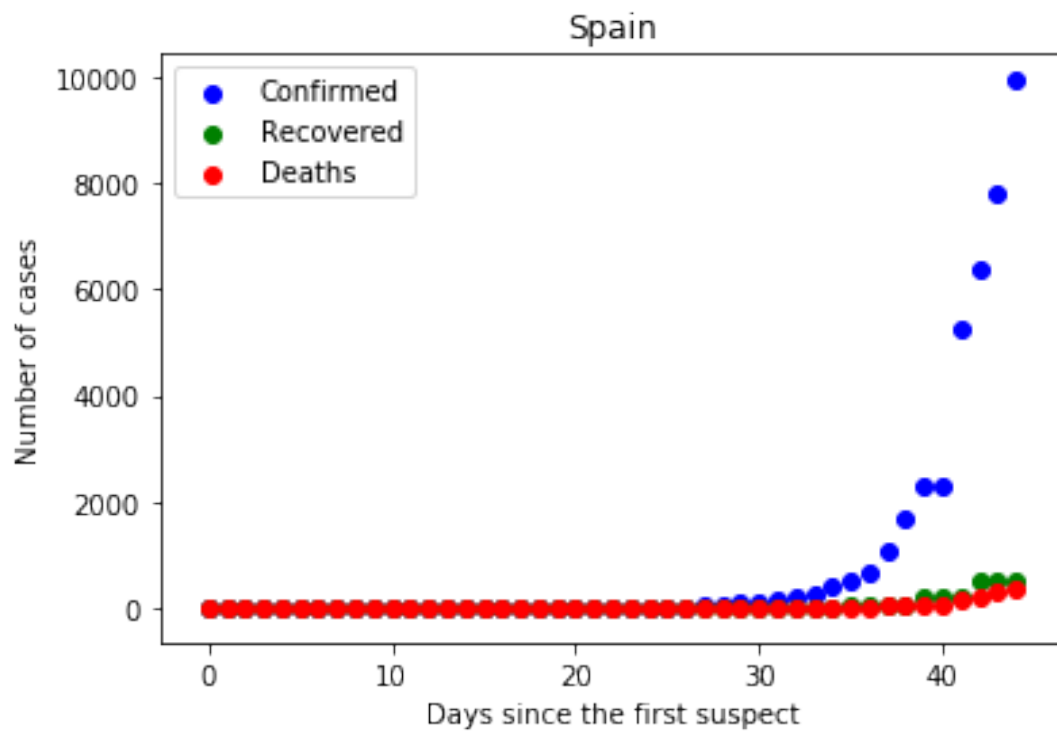


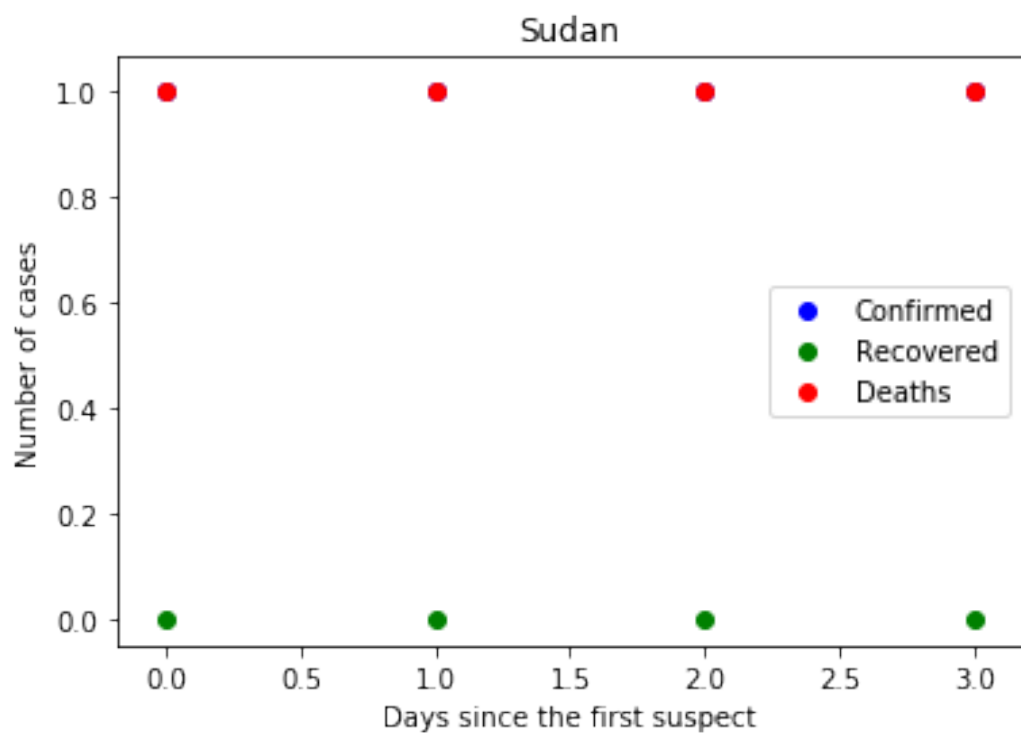
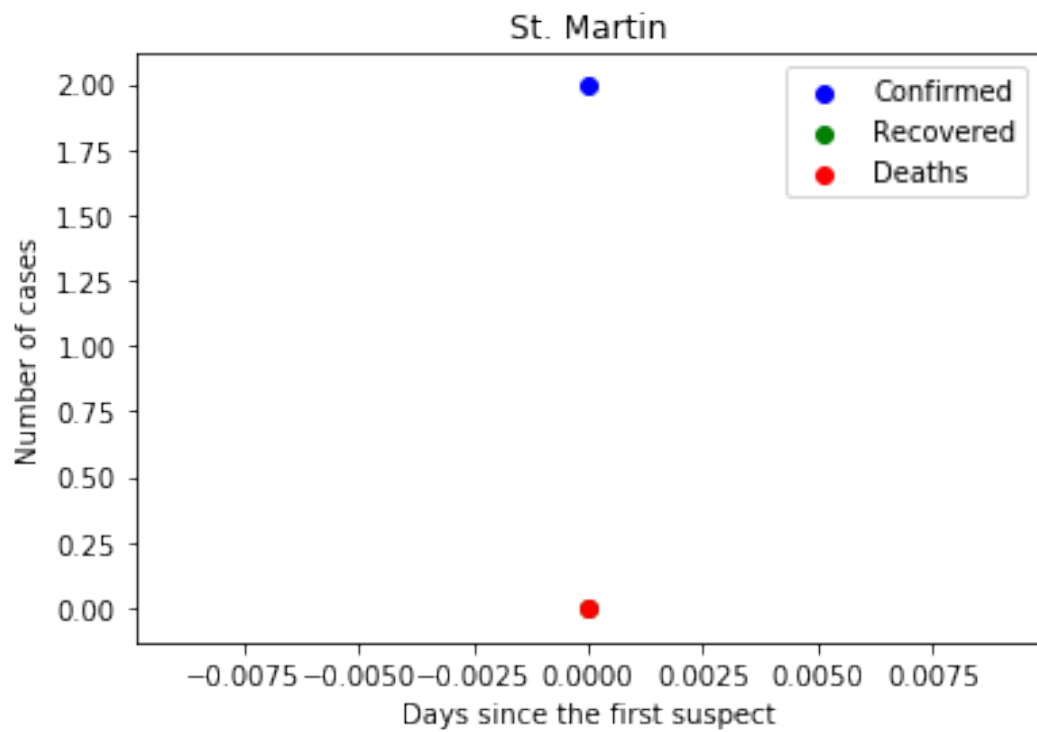


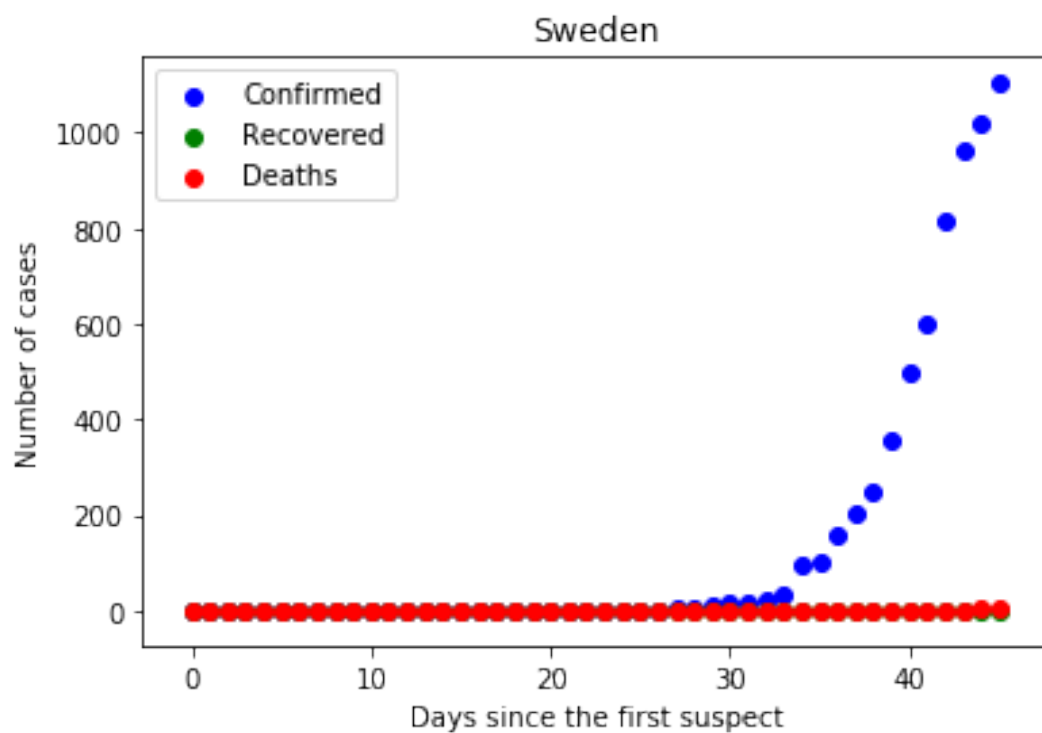
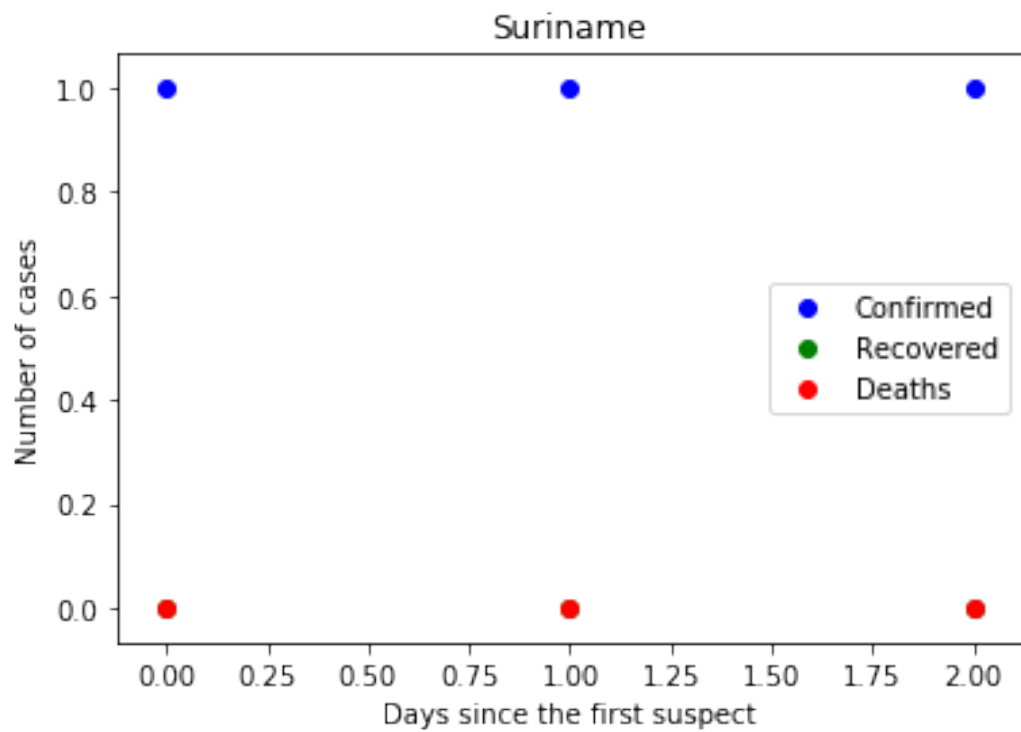


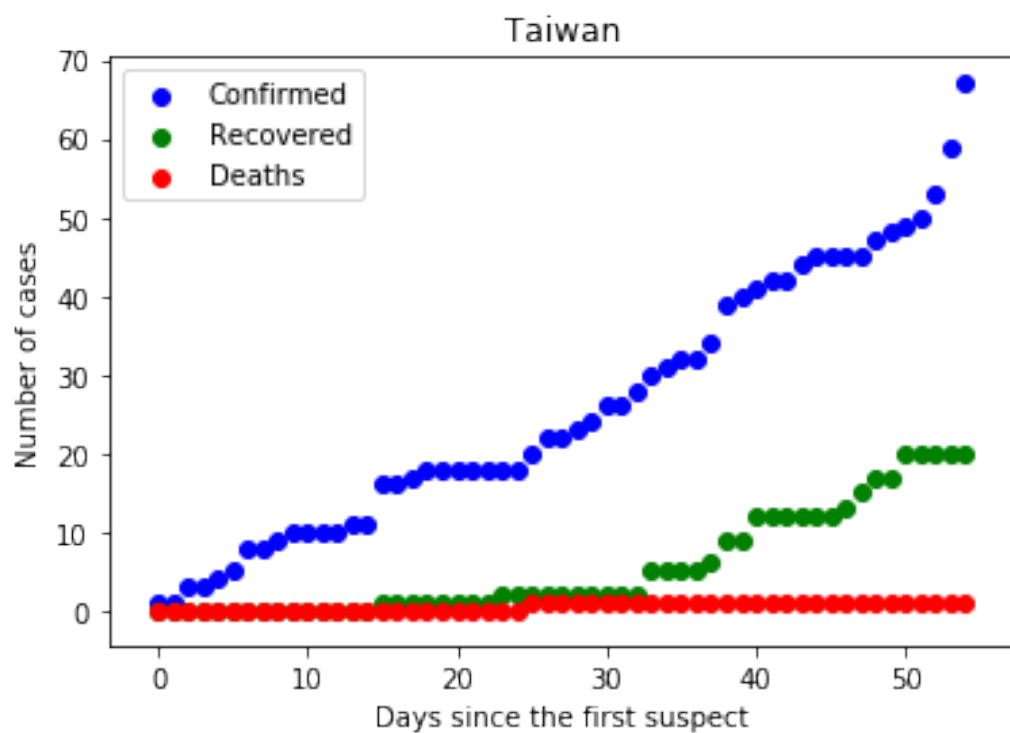
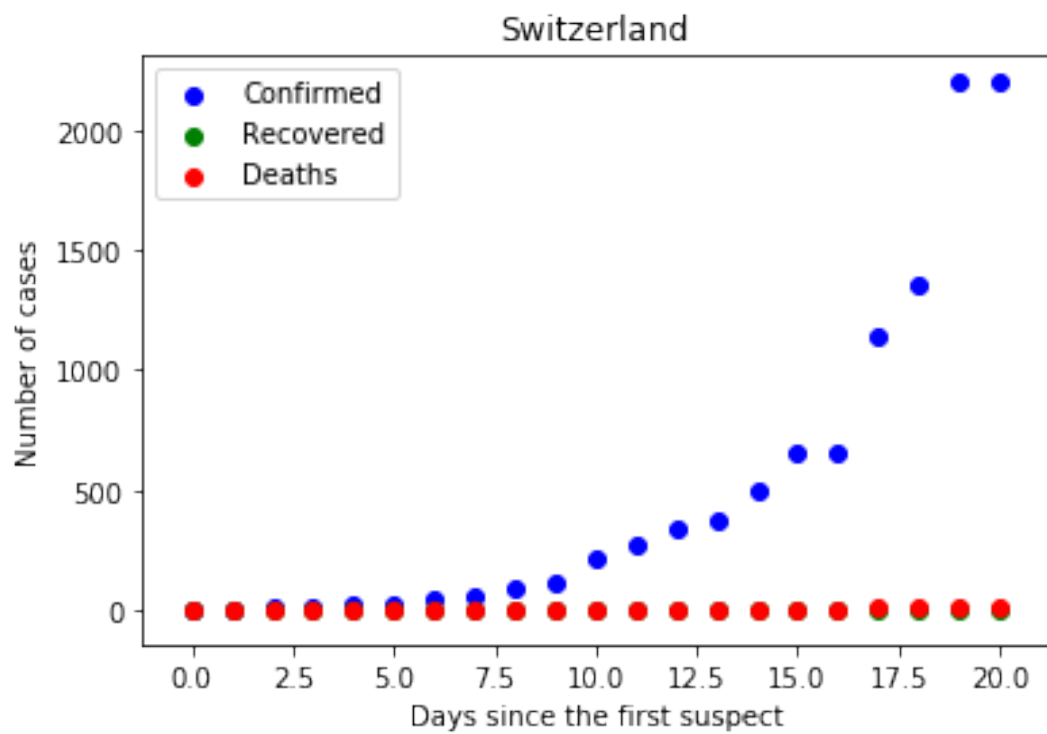


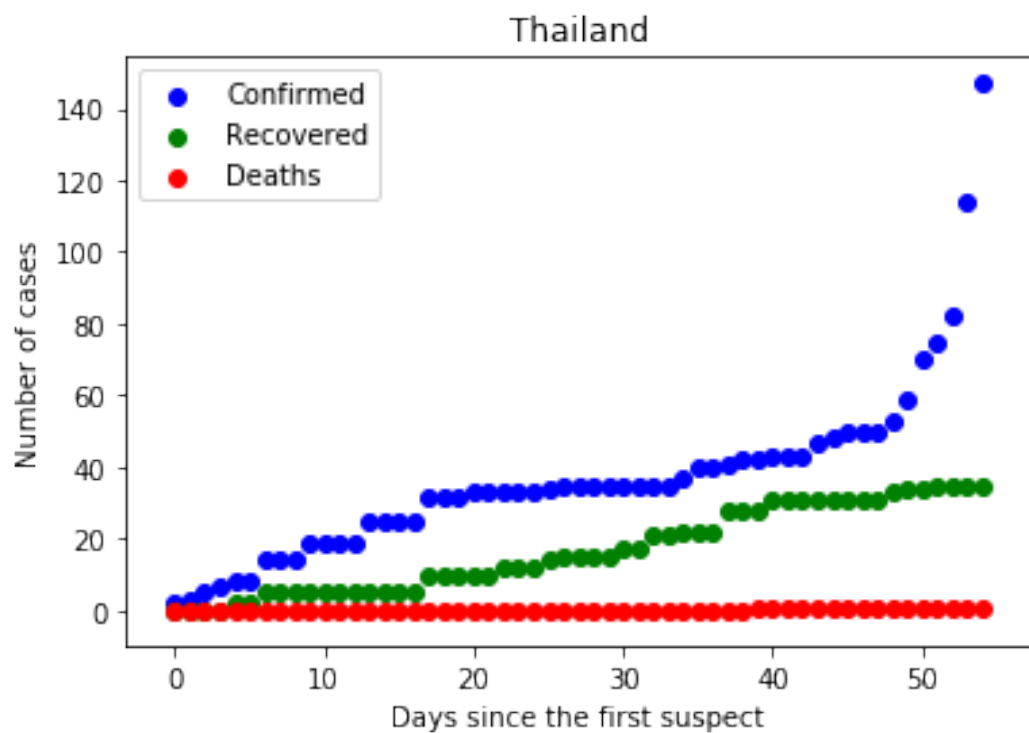
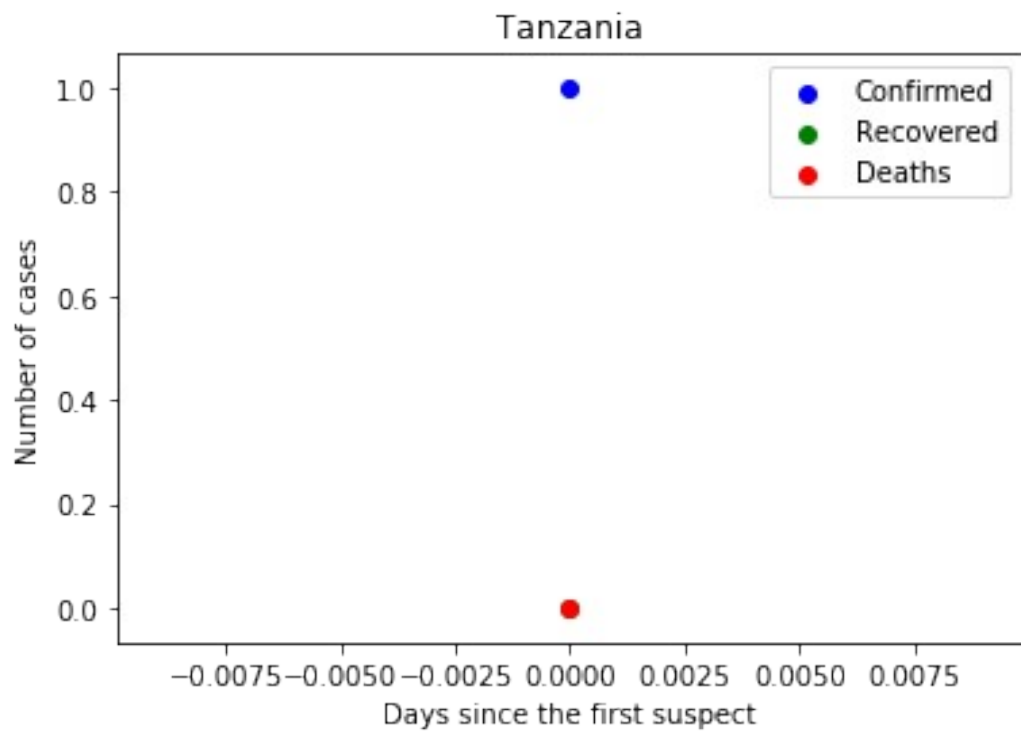




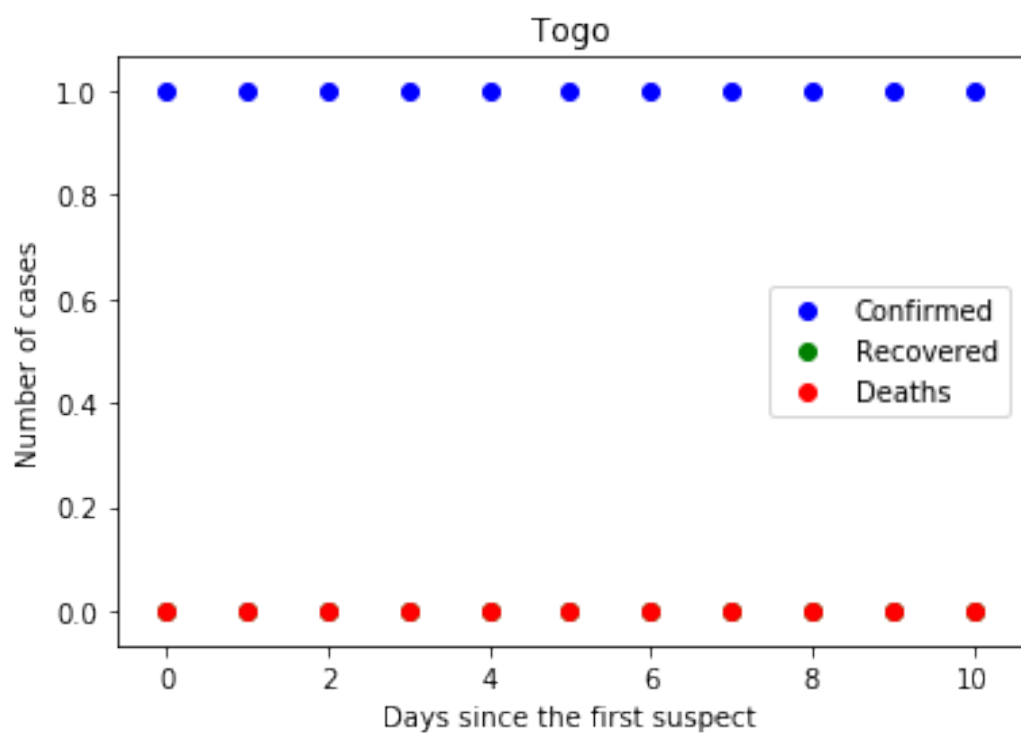
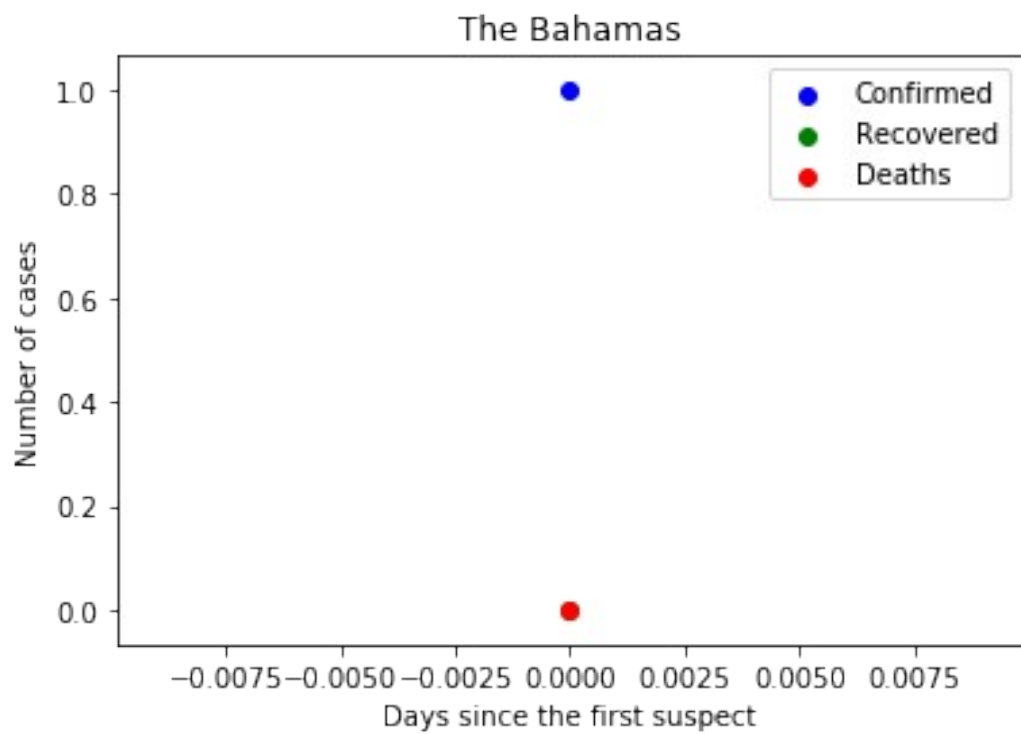


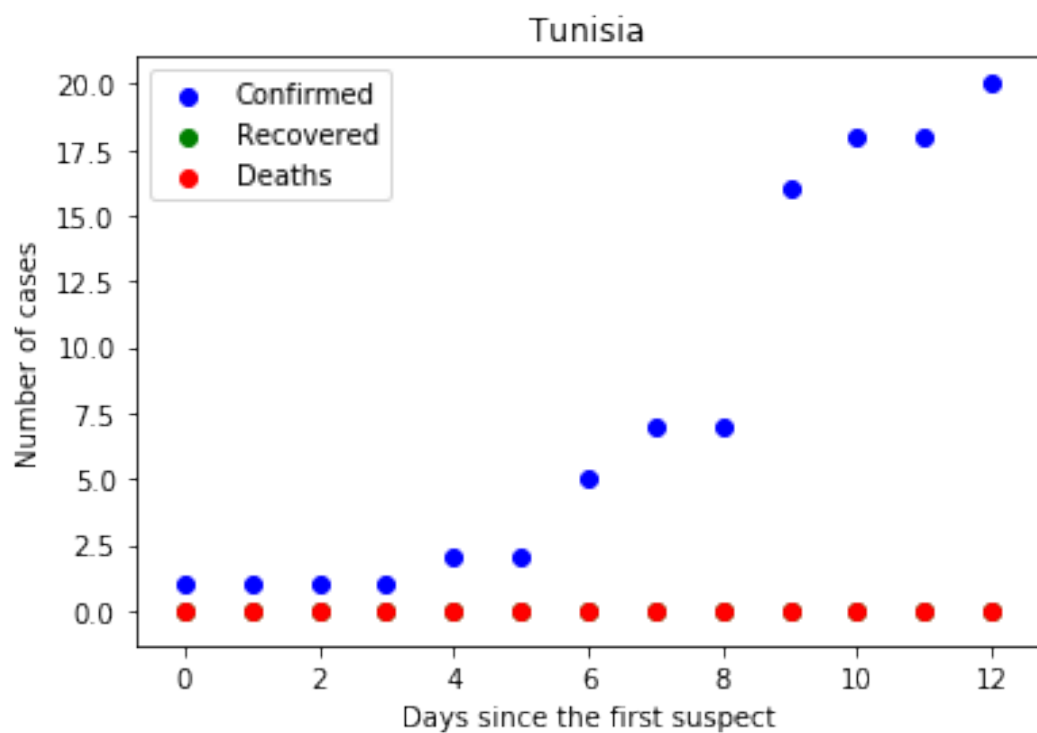
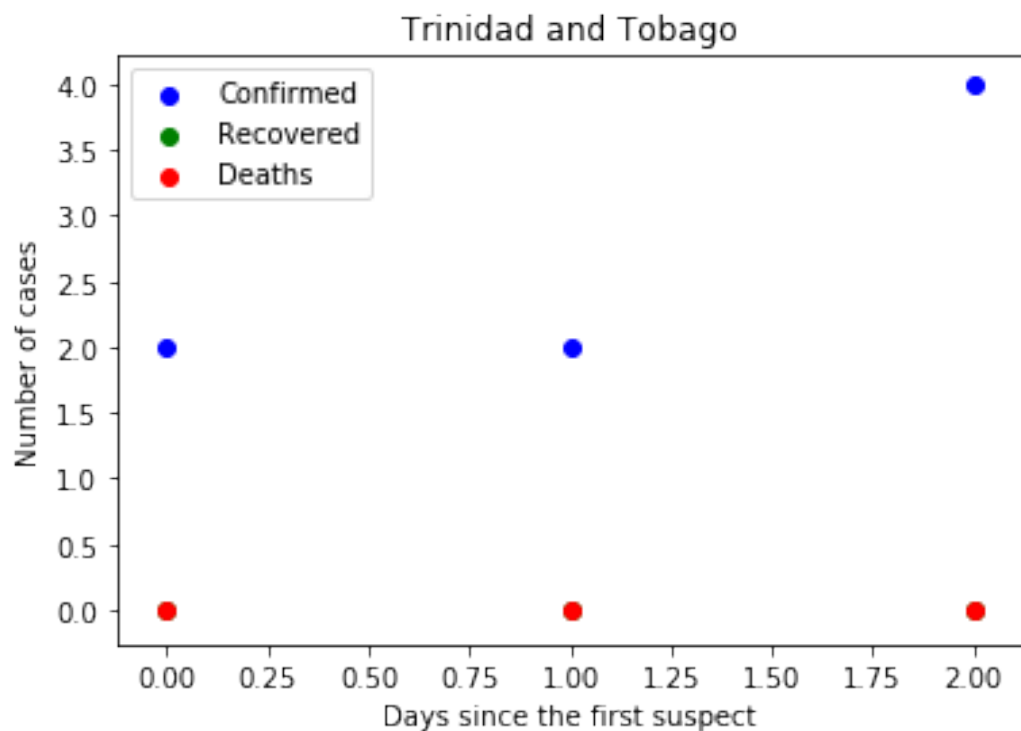


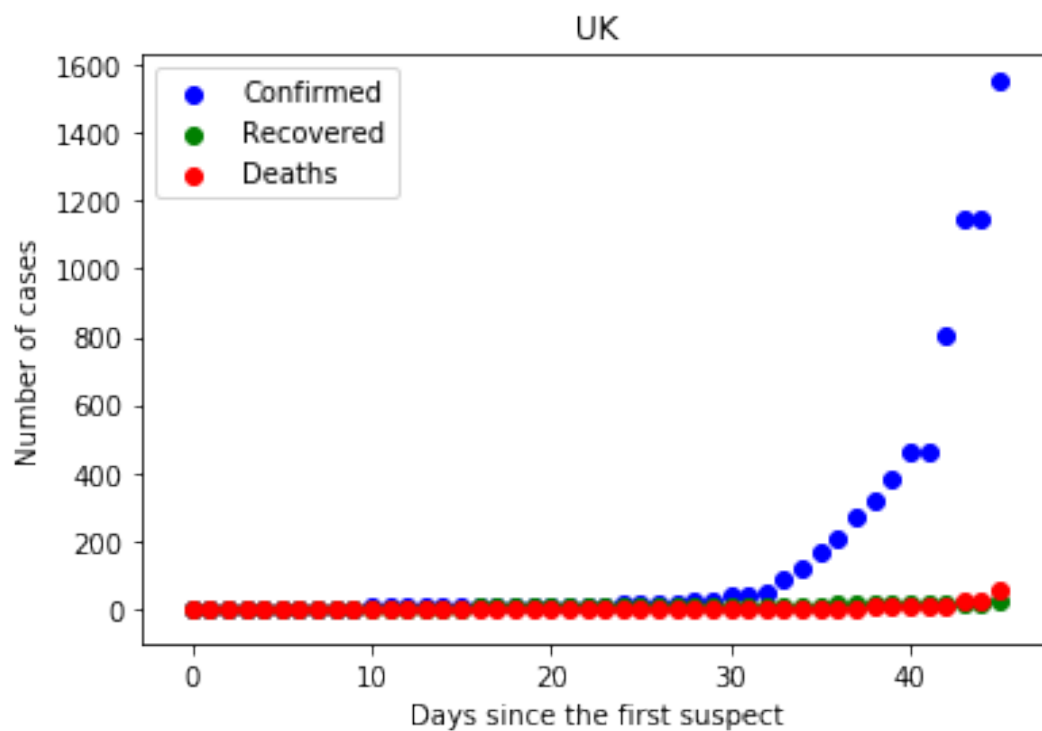
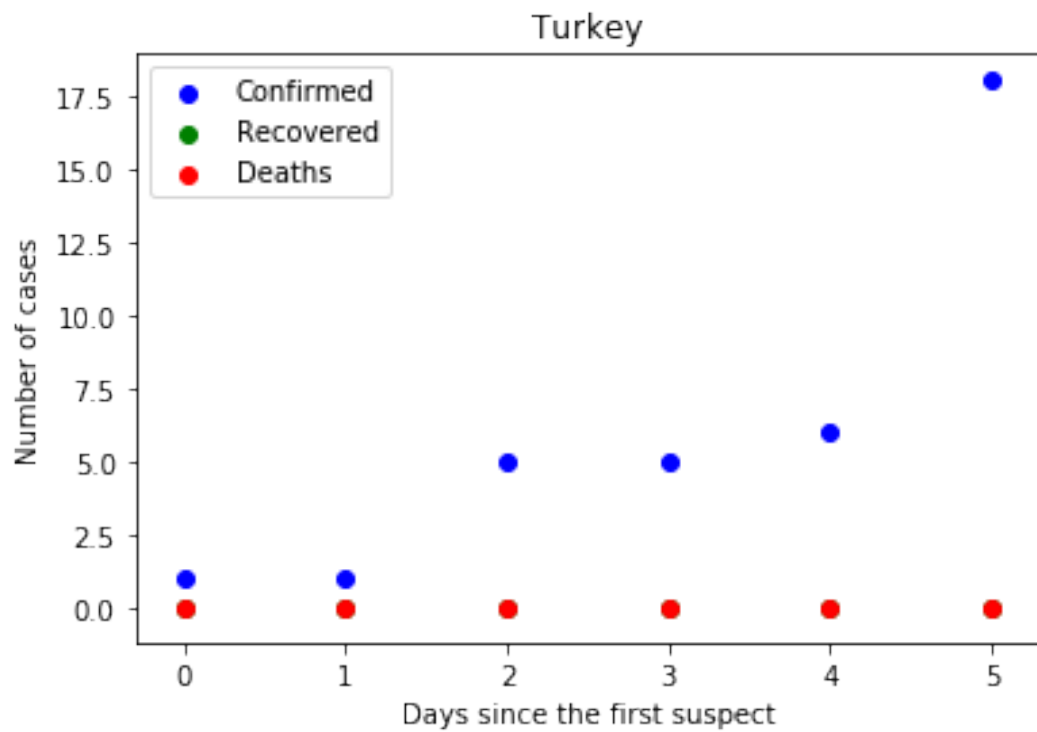


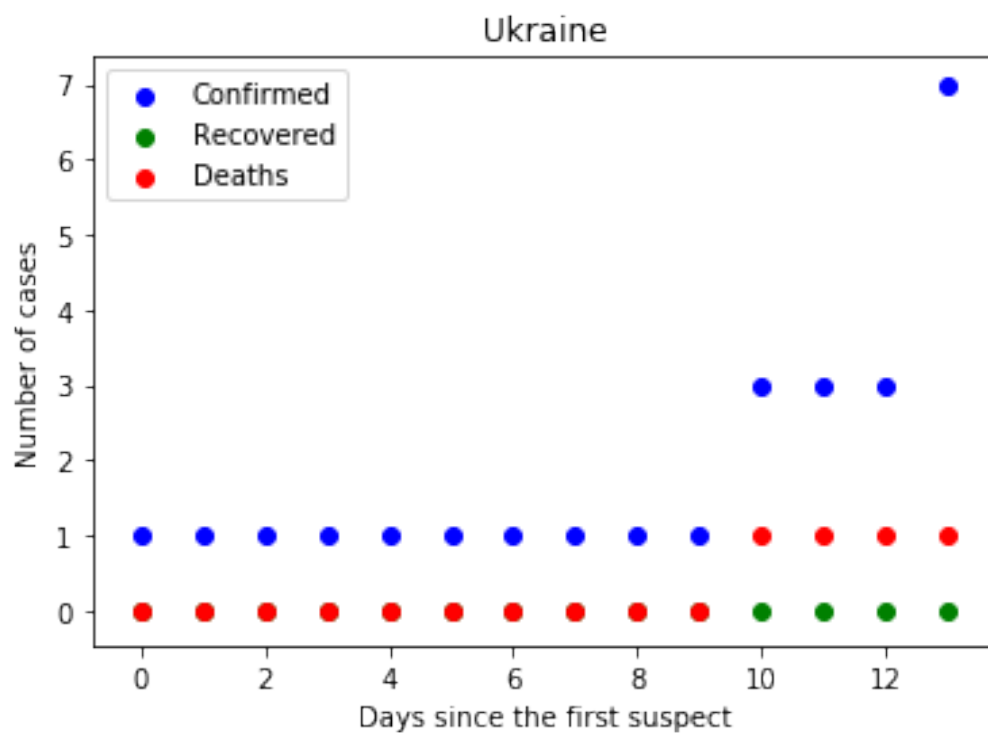
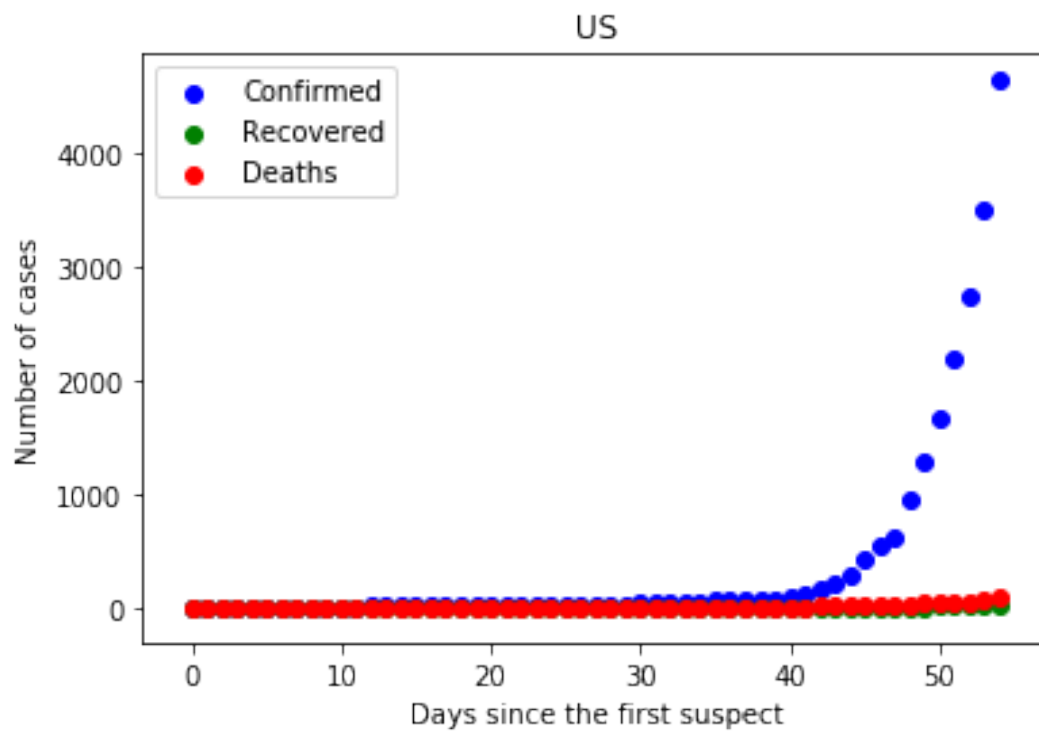


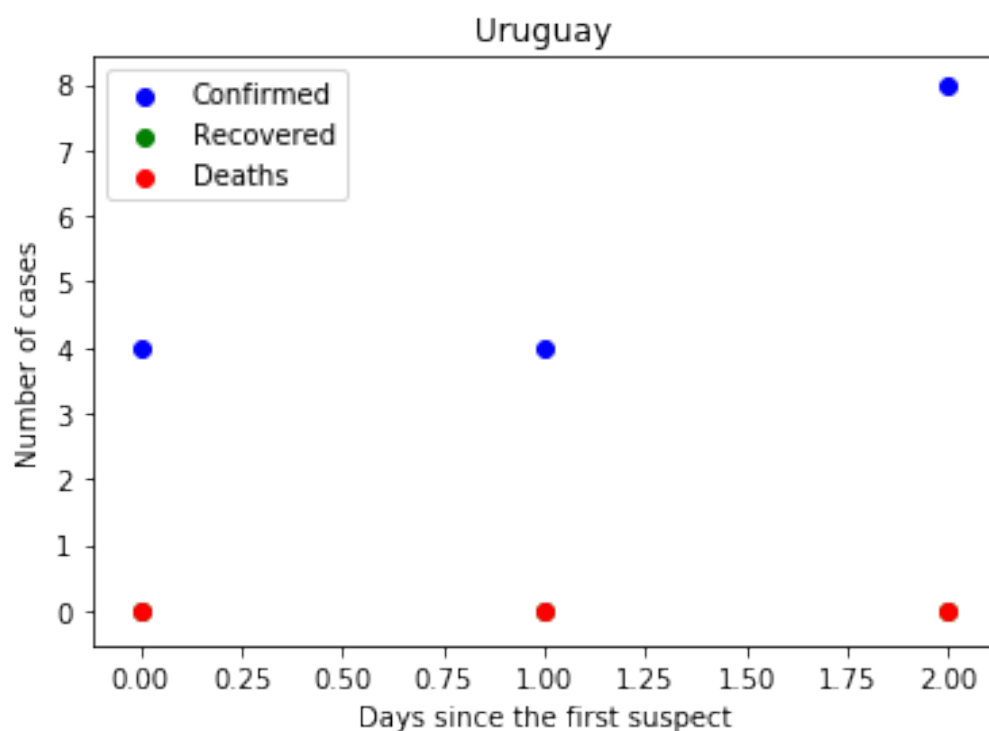
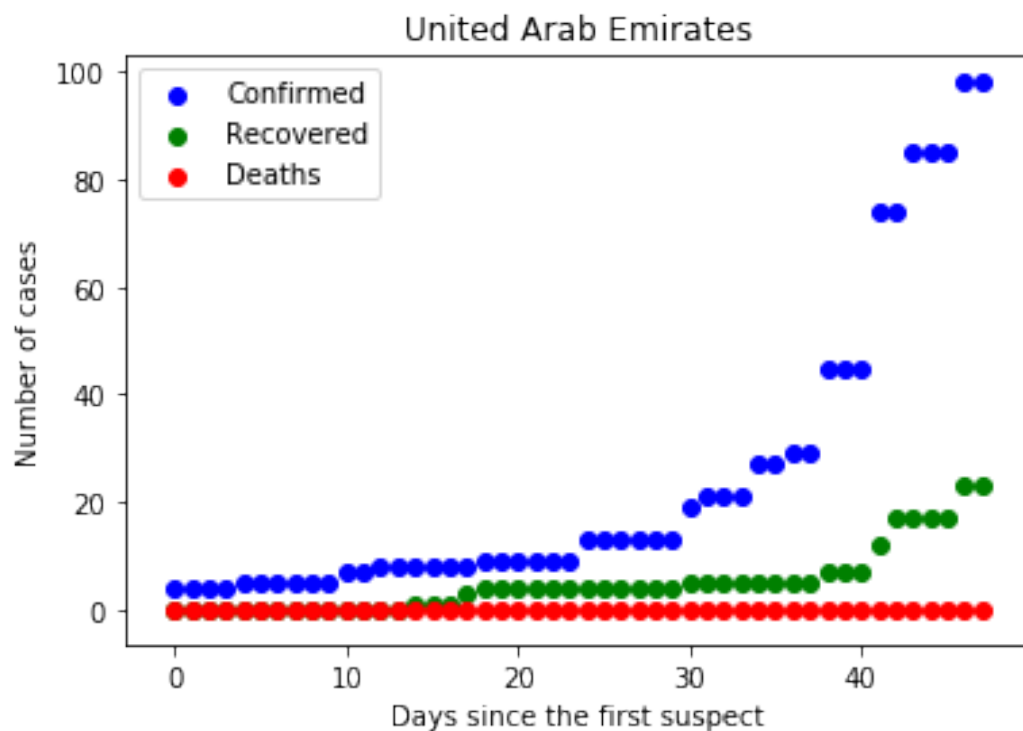


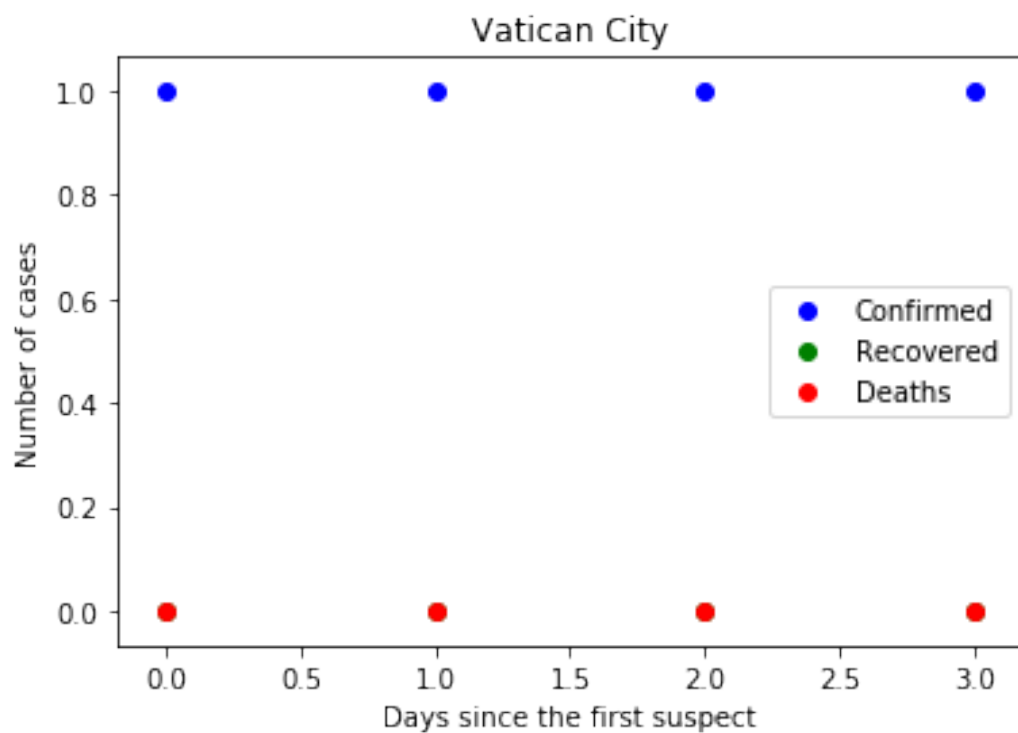
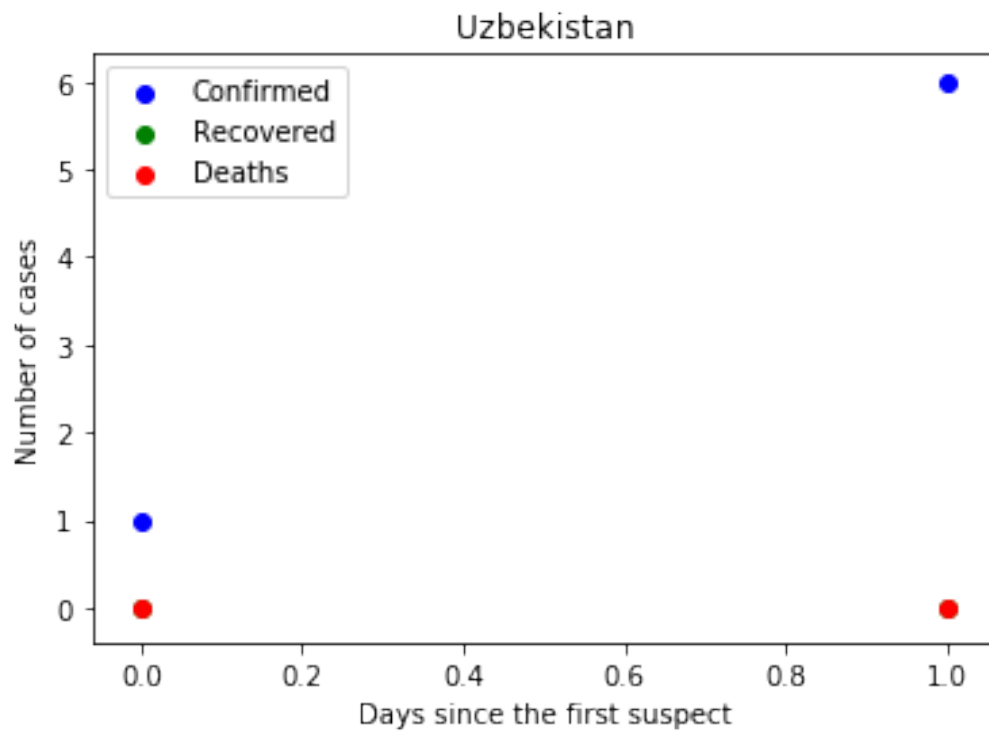


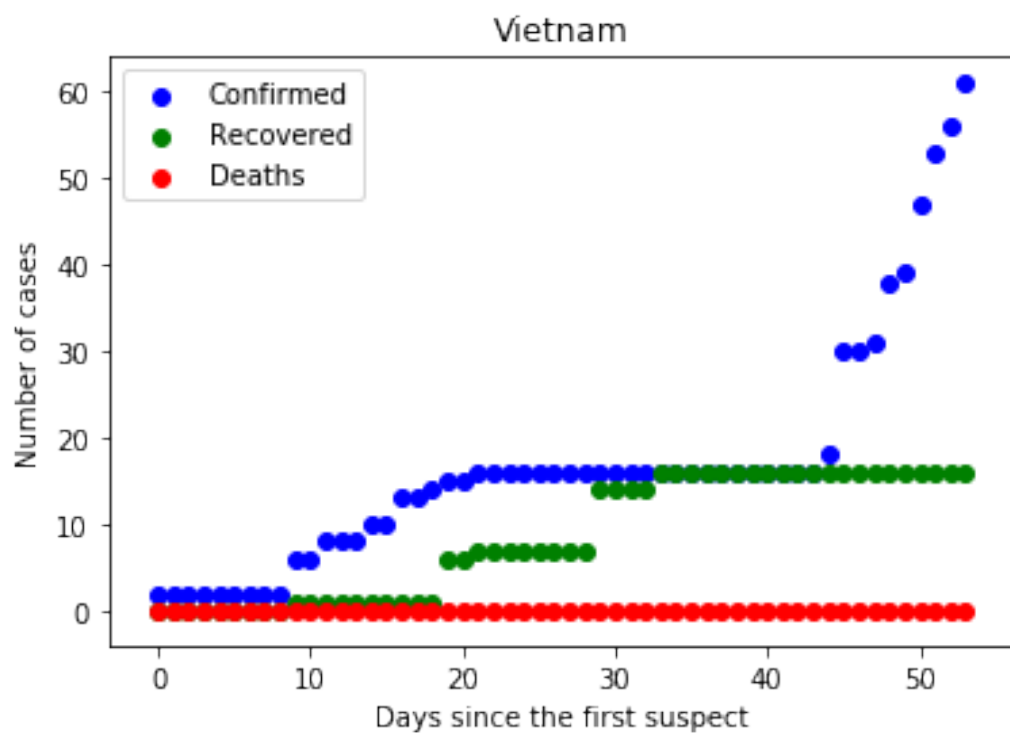
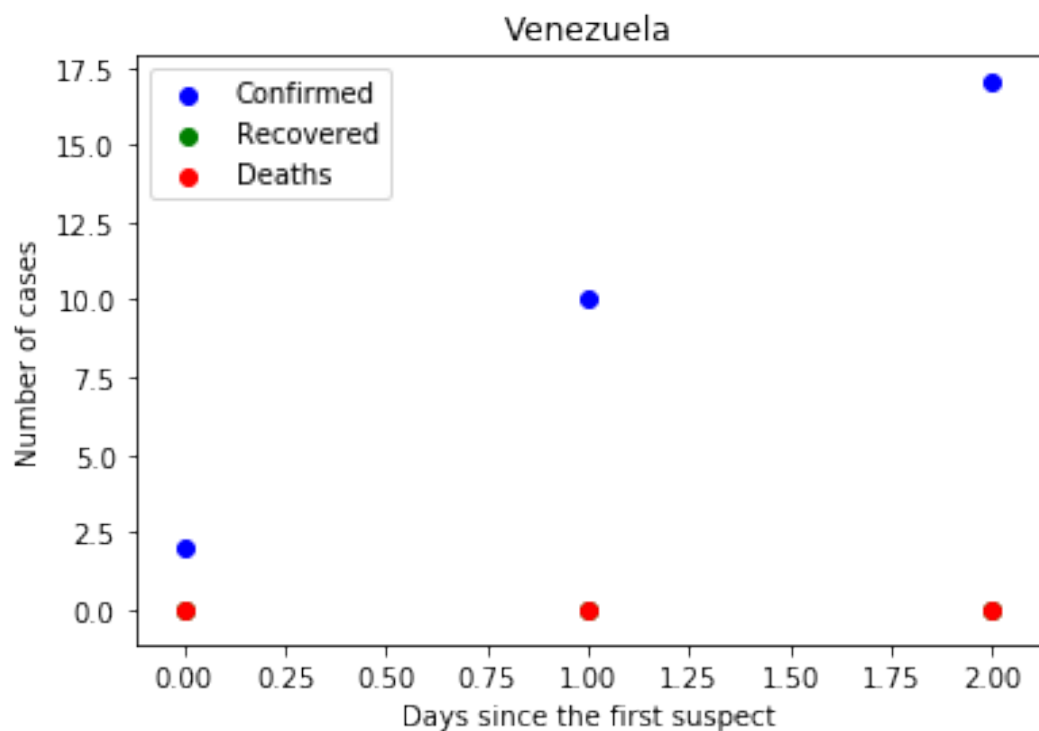


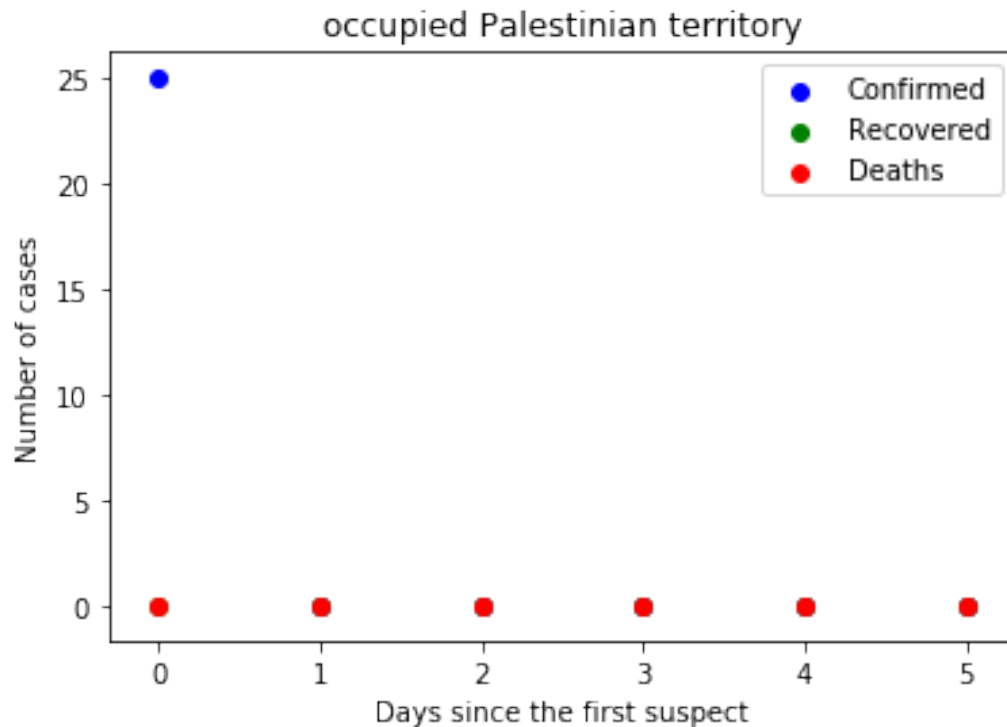












```
df4 = df3.groupby(['Date'])
[['Date', 'Confirmed', 'Deaths', 'Recovered']].sum().reset_index()

C = df4
plt.scatter(np.arange(0, len(C)), C['Confirmed'], color='blue', label='Confirmed')
plt.scatter(np.arange(0, len(C)), C['Recovered'], color='green', label='Recovered')
plt.scatter(np.arange(0, len(C)), C['Deaths'], color='red', label='Deaths')
plt.title('World')
plt.xlabel('Days since the first suspect')
plt.ylabel('Number of cases')
plt.legend()
plt.show()
```



