In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
print('Modules are imported.')
```

```
Modules are imported.
```

In [3]:

```
corona_dataset_csv = pd.read_csv('Datasets/covid19_Confirmed_dataset.csv')
corona_dataset_csv.head(10)
```

Out[3]:

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20
0	NaN	Afghanistan	33.0000	65.0000	0	0	0	0	0
1	NaN	Albania	41.1533	20.1683	0	0	0	0	0
2	NaN	Algeria	28.0339	1.6596	0	0	0	0	0
3	NaN	Andorra	42.5063	1.5218	0	0	0	0	0
4	NaN	Angola	-11.2027	17.8739	0	0	0	0	0
5	NaN	Antigua and Barbuda	17.0608	-61.7964	0	0	0	0	0
6	NaN	Argentina	-38.4161	-63.6167	0	0	0	0	0
7	NaN	Armenia	40.0691	45.0382	0	0	0	0	0
8	Australian Capital Territory	Australia	-35.4735	149.0124	0	0	0	0	0
9	New South Wales	Australia	-33.8688	151.2093	0	0	0	0	3

10 rows × 104 columns

In [4]:

```
corona_dataset_csv.shape
```

Out[4]:

```
(266, 104)
```

In [5]:

```
corona_dataset_csv.drop(['Lat','Long'],axis=1,inplace=True)
corona_dataset_csv.head(10)
```

Out[5]:

	Province/State	Country/Region	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20
0	NaN	Afghanistan	0	0	0	0	0	0	0
1	NaN	Albania	0	0	0	0	0	0	0
2	NaN	Algeria	0	0	0	0	0	0	0
3	NaN	Andorra	0	0	0	0	0	0	0
4	NaN	Angola	0	0	0	0	0	0	0
5	NaN	Antigua and Barbuda	0	0	0	0	0	0	0
6	NaN	Argentina	0	0	0	0	0	0	0
7	NaN	Armenia	0	0	0	0	0	0	0
8	Australian Capital Territory	Australia	0	0	0	0	0	0	0
9	New South Wales	Australia	0	0	0	0	3	4	4

10 rows × 102 columns

In [6]:

```
corona_dataset_aggregated = corona_dataset_csv.groupby("Country/Region").sum()
corona_dataset_aggregated.head(10)
```

Out[6]:

	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	1/
Country/Region										
Afghanistan	0	0	0	0	0	0	0	0	0	
Albania	0	0	0	0	0	0	0	0	0	
Algeria	0	0	0	0	0	0	0	0	0	
Andorra	0	0	0	0	0	0	0	0	0	
Angola	0	0	0	0	0	0	0	0	0	
Antigua and Barbuda	0	0	0	0	0	0	0	0	0	
Argentina	0	0	0	0	0	0	0	0	0	
Armenia	0	0	0	0	0	0	0	0	0	
Australia	0	0	0	0	4	5	5	6	9	
Austria	0	0	0	0	0	0	0	0	0	

10 rows × 100 columns

In [7]:

```
corona_dataset_aggregated.shape
```

Out[7]:

```
(187, 100)
```

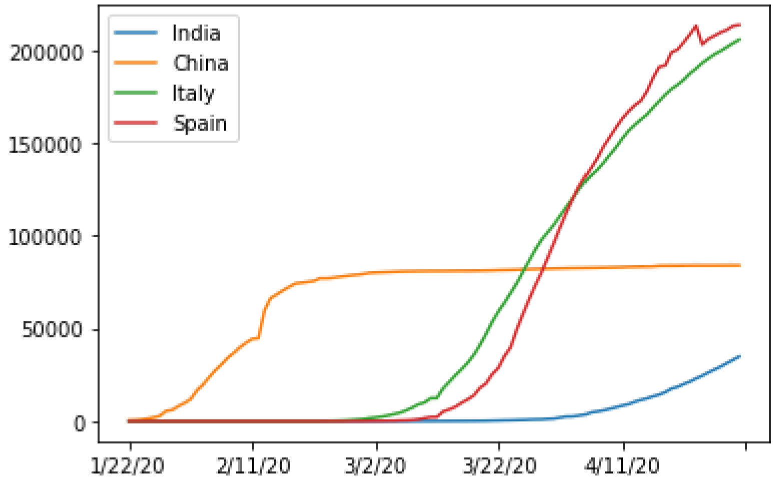
In [10]:

```
corona_dataset_aggregated.loc['India'].plot()
corona_dataset_aggregated.loc['China'].plot()
corona_dataset_aggregated.loc['Italy'].plot()
corona_dataset_aggregated.loc['Spain'].plot()

plt.legend()
```

Out[10]:

```
<matplotlib.legend.Legend at 0x16804990e10>
```

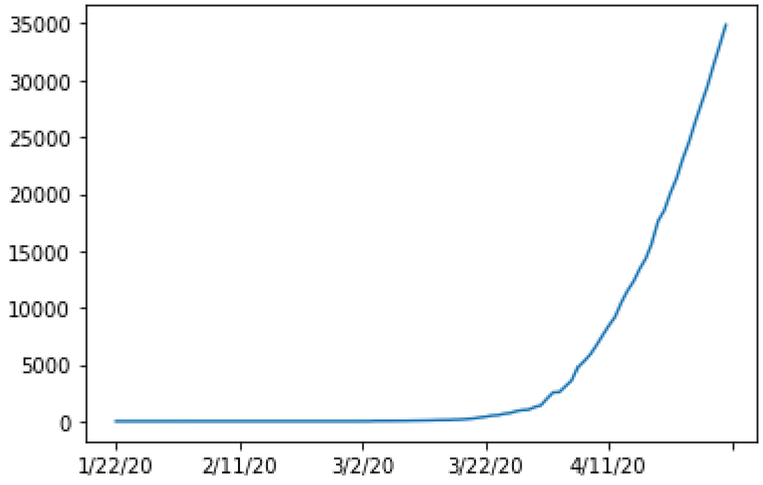


In [11]:

```
corona_dataset_aggregated.loc['India'].plot()
```

Out[11]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804a134e0>
```

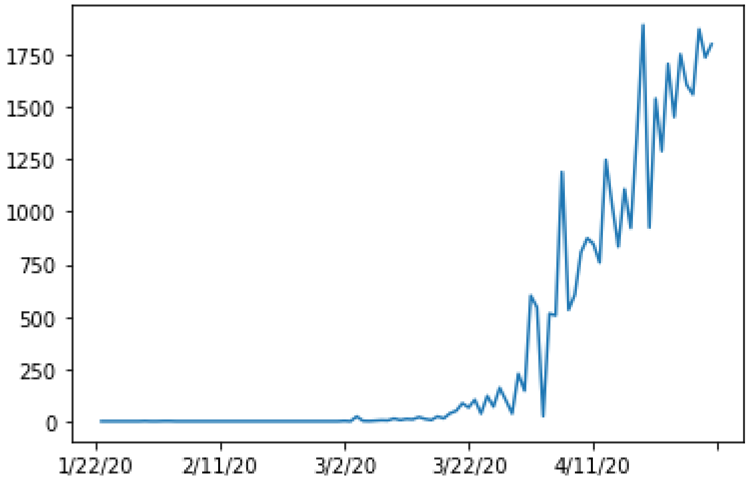


In [12]:

```
corona_dataset_aggregated.loc['India'].diff().plot()
```

Out[12]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804a75dd8>
```



In [13]:

```
corona_dataset_aggregated.loc['India'].diff().max()
```

Out[13]:

```
1893.0
```

In [14]:

```
corona_dataset_aggregated.loc['China'].diff().max()
```

Out[14]:

```
15136.0
```

In [15]:

```
corona_dataset_aggregated.loc['Italy'].diff().max()
```

Out[15]:

```
6557.0
```

In [16]:

```
corona_dataset_aggregated.loc['Spain'].diff().max()
```

Out[16]:

```
9630.0
```

In [17]:

```
countries = list(corona_dataset_aggregated.index)
max_infection_rates = []
for country in countries :
    max_infection_rates.append(corona_dataset_aggregated.loc[country].diff().max())
corona_dataset_aggregated['max infection rate'] = max_infection_rates
```

In [18]:

```
corona_dataset_aggregated.head()
```

Out[18]:

	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	1/3
Country/Region										
Afghanistan	0	0	0	0	0	0	0	0	0	
Albania	0	0	0	0	0	0	0	0	0	
Algeria	0	0	0	0	0	0	0	0	0	
Andorra	0	0	0	0	0	0	0	0	0	
Angola	0	0	0	0	0	0	0	0	0	

5 rows × 101 columns

In [20]:

```
corona_data = pd.DataFrame(corona_dataset_aggregated['max infection rate'])
corona_data.head()
```

Out[20]:

	max infection rate
Country/Region	
Afghanistan	232.0
Albania	34.0
Algeria	199.0
Andorra	43.0
Angola	5.0

In [21]:

```
world_happiness_report = pd.read_csv("Datasets/worldwide_happiness_report.csv")
world_happiness_report.head()
```

Out[21]:

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.769	1.340	1.587	0.986	0.596	0.153	0.393
1	2	Denmark	7.600	1.383	1.573	0.996	0.592	0.252	0.410
2	3	Norway	7.554	1.488	1.582	1.028	0.603	0.271	0.341
3	4	Iceland	7.494	1.380	1.624	1.026	0.591	0.354	0.118
4	5	Netherlands	7.488	1.396	1.522	0.999	0.557	0.322	0.298

In [22]:

```
world_happiness_report.shape
```

Out[22]:

```
(156, 9)
```

In [ ]:

```
columns_to_dropped = ['Overall rank','Score','Generosity','Perceptions of corruption']
world_happiness_report.drop(columns_to_dropped,axis=1 , inplace=True)
```

In [23]:

```
world_happiness_report.head()
```

Out[23]:

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.769	1.340	1.587	0.986	0.596	0.153	0.393
1	2	Denmark	7.600	1.383	1.573	0.996	0.592	0.252	0.410
2	3	Norway	7.554	1.488	1.582	1.028	0.603	0.271	0.341
3	4	Iceland	7.494	1.380	1.624	1.026	0.591	0.354	0.118
4	5	Netherlands	7.488	1.396	1.522	0.999	0.557	0.322	0.298

In [24]:

```
world_happiness_report.set_index(['Country or region'],inplace=True)
world_happiness_report.head()
```

Out[24]:

	Overall rank	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
Country or region								
Finland	1	7.769	1.340	1.587	0.986	0.596	0.153	0.393
Denmark	2	7.600	1.383	1.573	0.996	0.592	0.252	0.410
Norway	3	7.554	1.488	1.582	1.028	0.603	0.271	0.341
Iceland	4	7.494	1.380	1.624	1.026	0.591	0.354	0.118
Netherlands	5	7.488	1.396	1.522	0.999	0.557	0.322	0.298

In [25]:

```
corona_data.head()
```

Out[25]:

	max infection rate
Country/Region	
Afghanistan	232.0
Albania	34.0
Algeria	199.0
Andorra	43.0
Angola	5.0

In [26]:

```
world_happiness_report.head()
```

Out[26]:

Country or region	Overall rank	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
Finland	1	7.769	1.340	1.587	0.986	0.596	0.153	0.393
Denmark	2	7.600	1.383	1.573	0.996	0.592	0.252	0.410
Norway	3	7.554	1.488	1.582	1.028	0.603	0.271	0.341
Iceland	4	7.494	1.380	1.624	1.026	0.591	0.354	0.118
Netherlands	5	7.488	1.396	1.522	0.999	0.557	0.322	0.298

In [27]:

```
data = world_happiness_report.join(corona_data).copy()
data.head()
```

Out[27]:

Country or region	Overall rank	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	in
Finland	1	7.769	1.340	1.587	0.986	0.596	0.153	0.393	
Denmark	2	7.600	1.383	1.573	0.996	0.592	0.252	0.410	
Norway	3	7.554	1.488	1.582	1.028	0.603	0.271	0.341	
Iceland	4	7.494	1.380	1.624	1.026	0.591	0.354	0.118	
Netherlands	5	7.488	1.396	1.522	0.999	0.557	0.322	0.298	

In [28]:

```
data.corr()
```

Out[28]:

	Overall rank	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perc co
Overall rank	1.000000	-0.989096	-0.801947	-0.767465	-0.787411	-0.546606	-0.047993	-(
Score	-0.989096	1.000000	0.793883	0.777058	0.779883	0.566742	0.075824	(
GDP per capita	-0.801947	0.793883	1.000000	0.754906	0.835462	0.379079	-0.079662	(
Social support	-0.767465	0.777058	0.754906	1.000000	0.719009	0.447333	-0.048126	(
Healthy life expectancy	-0.787411	0.779883	0.835462	0.719009	1.000000	0.390395	-0.029511	(
Freedom to make life choices	-0.546606	0.566742	0.379079	0.447333	0.390395	1.000000	0.269742	(
Generosity	-0.047993	0.075824	-0.079662	-0.048126	-0.029511	0.269742	1.000000	(
Perceptions of corruption	-0.351959	0.385613	0.298920	0.181899	0.295283	0.438843	0.326538	1
max infection rate	-0.244777	0.228783	0.250118	0.191958	0.289263	0.078196	-0.114635	(

In [29]:

```
data.head()
```

Out[29]:

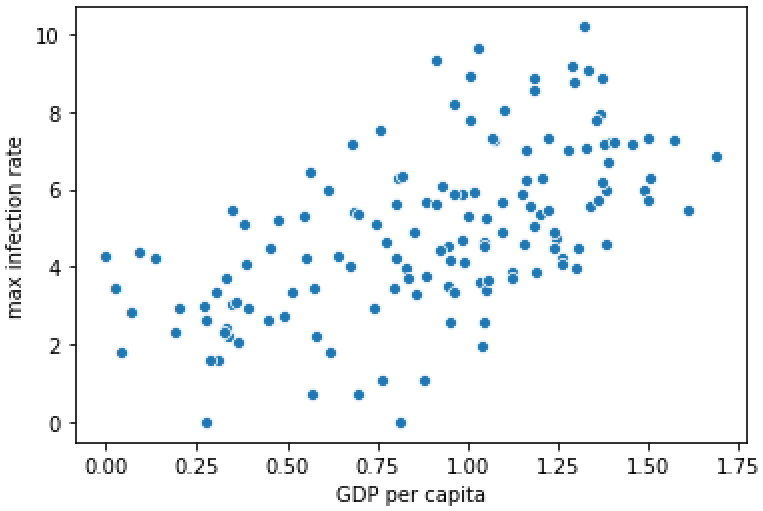
	Overall rank	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	in
Country or region									
Finland	1	7.769	1.340	1.587	0.986	0.596	0.153	0.393	
Denmark	2	7.600	1.383	1.573	0.996	0.592	0.252	0.410	
Norway	3	7.554	1.488	1.582	1.028	0.603	0.271	0.341	
Iceland	4	7.494	1.380	1.624	1.026	0.591	0.354	0.118	
Netherlands	5	7.488	1.396	1.522	0.999	0.557	0.322	0.298	

In [30]:

```
x = data['GDP per capita']
y = data['max infection rate']
sns.scatterplot(x,np.log(y))
```

Out[30]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804b0c588>
```

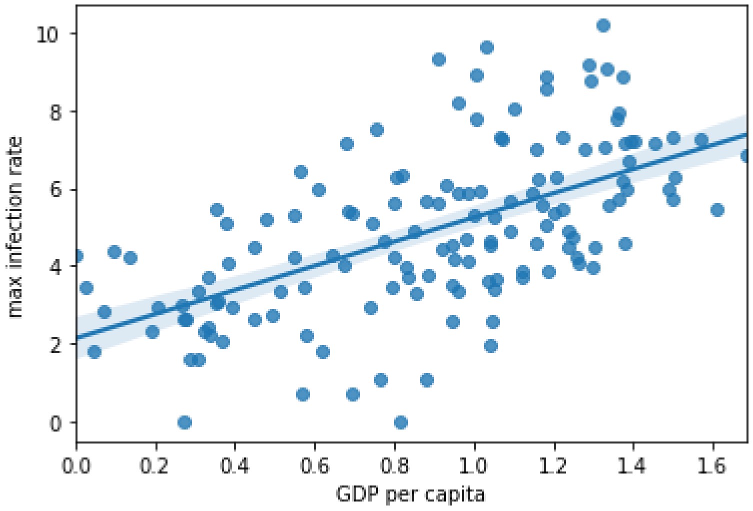


In [31]:

```
sns.regplot(x,np.log(y))
```

Out[31]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804b80048>
```

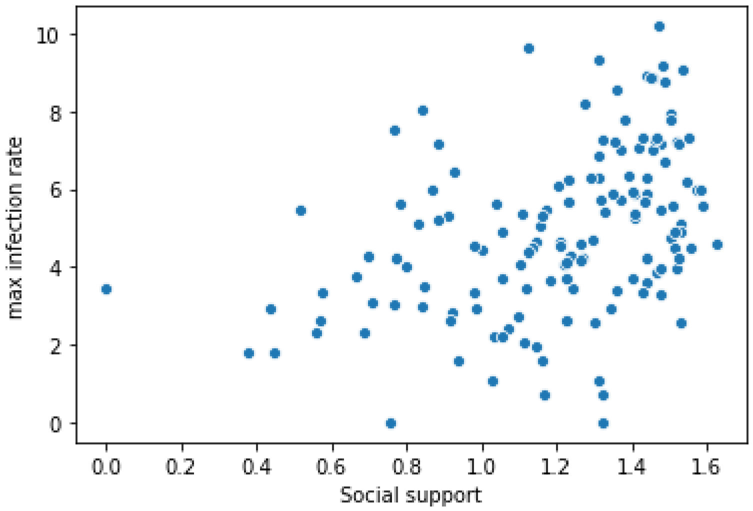


In [32]:

```
x = data['Social support']
y = data['max infection rate']
sns.scatterplot(x,np.log(y))
```

Out[32]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804bf12b0>
```

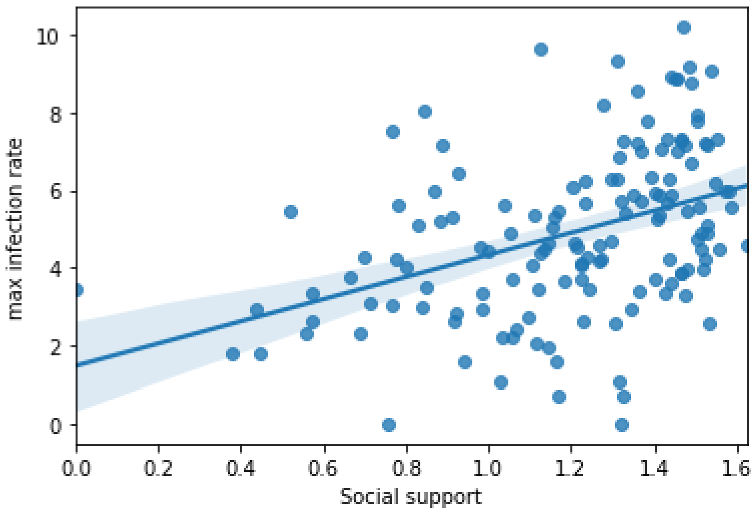


In [33]:

```
sns.regplot(x,np.log(y))
```

Out[33]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804bdf630>
```

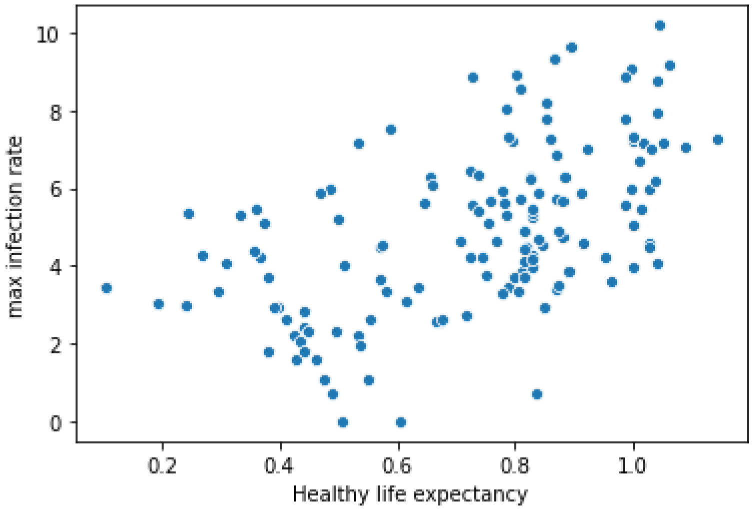


In [34]:

```
x = data['Healthy life expectancy']
y = data['max infection rate']
sns.scatterplot(x,np.log(y))
```

Out[34]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804c9eef0>
```

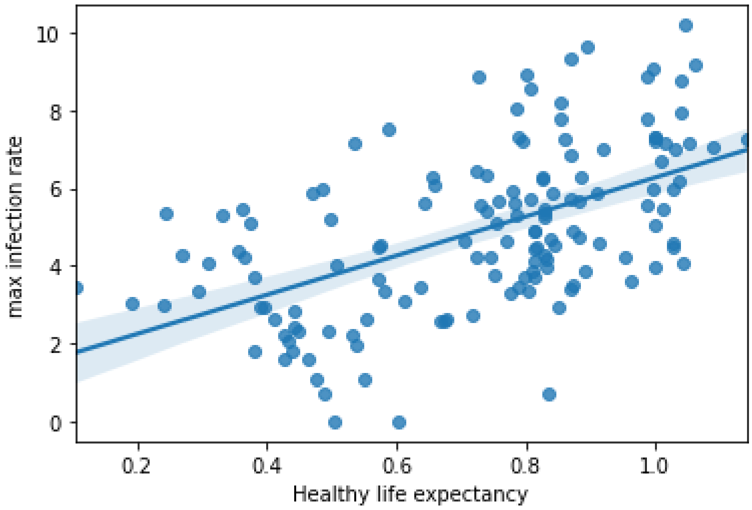


In [35]:

```
sns.regplot(x,np.log(y))
```

Out[35]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804cfe0f0>
```

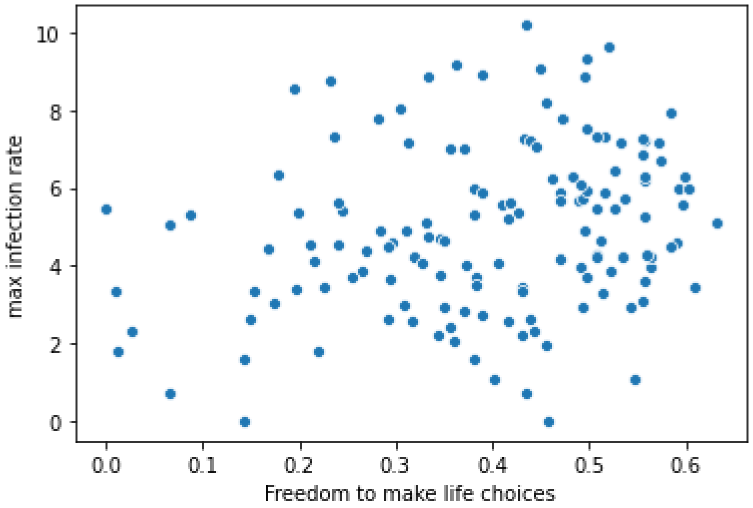


In [36]:

```
x = data['Freedom to make life choices']
y = data['max infection rate']
sns.scatterplot(x,np.log(y))
```

Out[36]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804d5ae80>
```

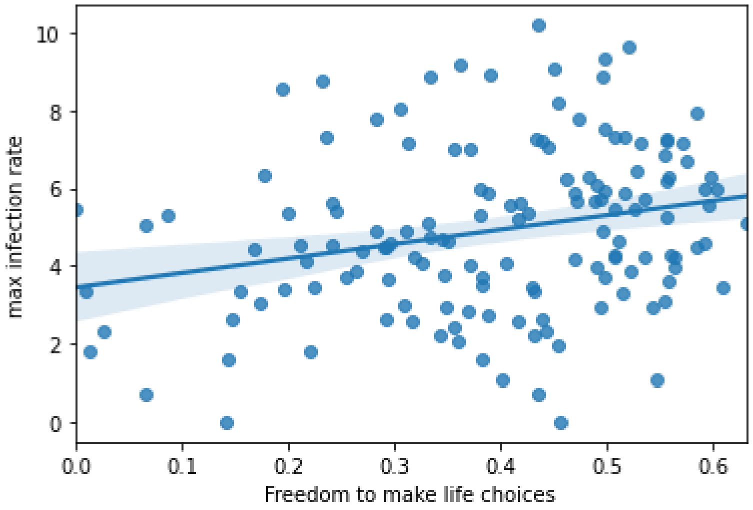


In [37]:

```
sns.regplot(x,np.log(y))
```

Out[37]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x16804d9efd0>
```



In [ ]: