

Covid Ecuador

Librerias

```
In [7]: import pandas as pd
import numpy as np
from datetime import datetime, timedelta
from sklearn.metrics import mean_squared_error
from scipy.optimize import curve_fit
from scipy.optimize import fsolve
from sklearn import linear_model
import matplotlib.pyplot as plt
%matplotlib inline
```

Cargar datos estadisticos

```
In [8]: url = 'covid.csv'
df = pd.read_csv(url)
df
```

Out[8]:

	Date_reported	Country_code	Country	WHO_region	New_cases	Cumulative_cases	New_deaths	Cui
0	2020-01-03	AF	Afghanistan	EMRO	0	0	0	
1	2020-01-04	AF	Afghanistan	EMRO	0	0	0	
2	2020-01-05	AF	Afghanistan	EMRO	0	0	0	
3	2020-01-06	AF	Afghanistan	EMRO	0	0	0	
4	2020-01-07	AF	Afghanistan	EMRO	0	0	0	
...
71670	2020-10-29	ZW	Zimbabwe	AFRO	5	8320	0	
71671	2020-10-30	ZW	Zimbabwe	AFRO	1	8321	0	
71672	2020-10-31	ZW	Zimbabwe	AFRO	41	8362	0	
71673	2020-11-01	ZW	Zimbabwe	AFRO	5	8367	1	
71674	2020-11-02	ZW	Zimbabwe	AFRO	7	8374	0	

71675 rows × 8 columns

Grafica de personas infectadas de COVID19 Ecuador

```

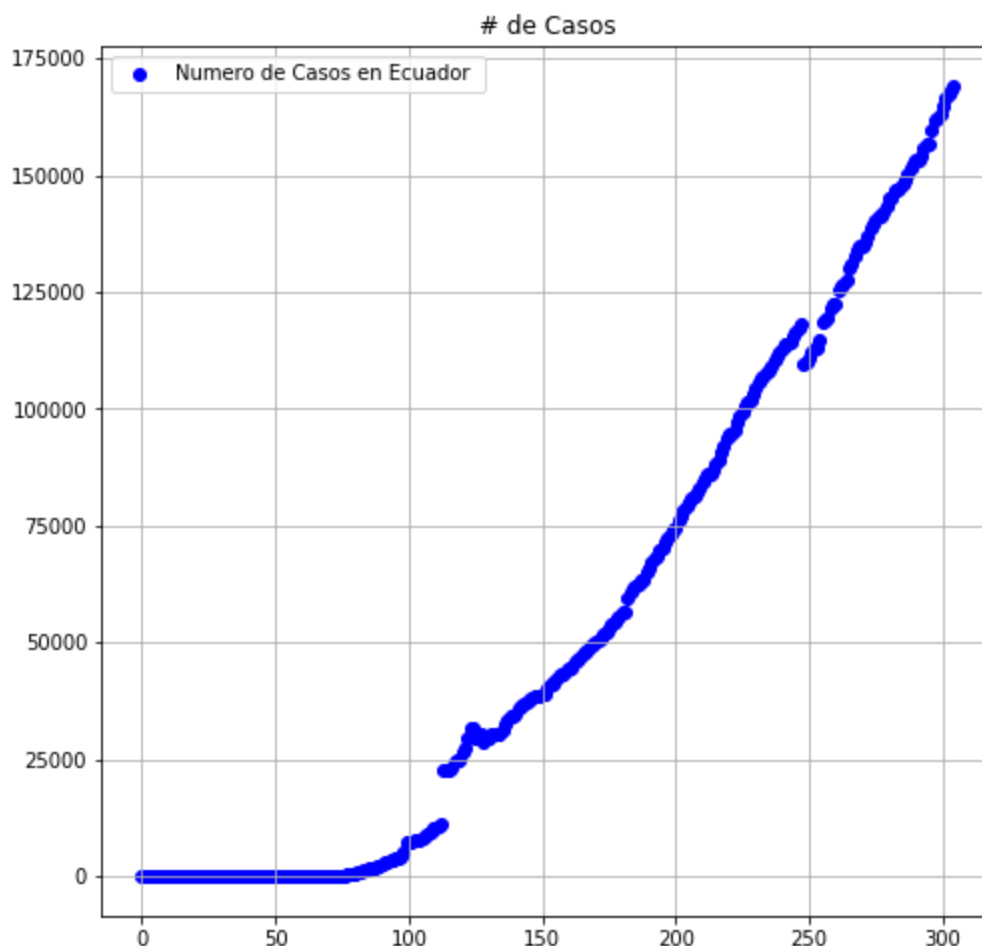
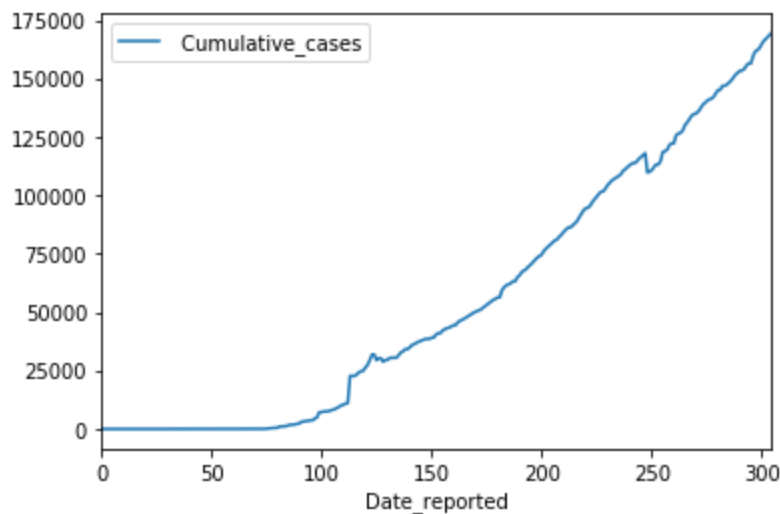
In [10]: url = 'covid.csv'
df = pd.read_csv(url)

df = df[df['Country'].isin(['Ecuador'])]
df = df.loc[:, ['Date_reported', 'Cumulative_cases']]
FMT = '%Y-%m-%d'
date = df['Date_reported']
df['Date_reported'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-03-01", FMT)).days)
df.plot(x = 'Date_reported', y = 'Cumulative_cases')

x = np.array(df.values[:,0])
y = np.array(df.values[:,1])

plt.figure(figsize=(8, 8))
plt.scatter(x,y,label='Numero de Casos en Ecuador ', color='blue')
plt.grid(True)
plt.legend()
plt.title('# de Casos');

```

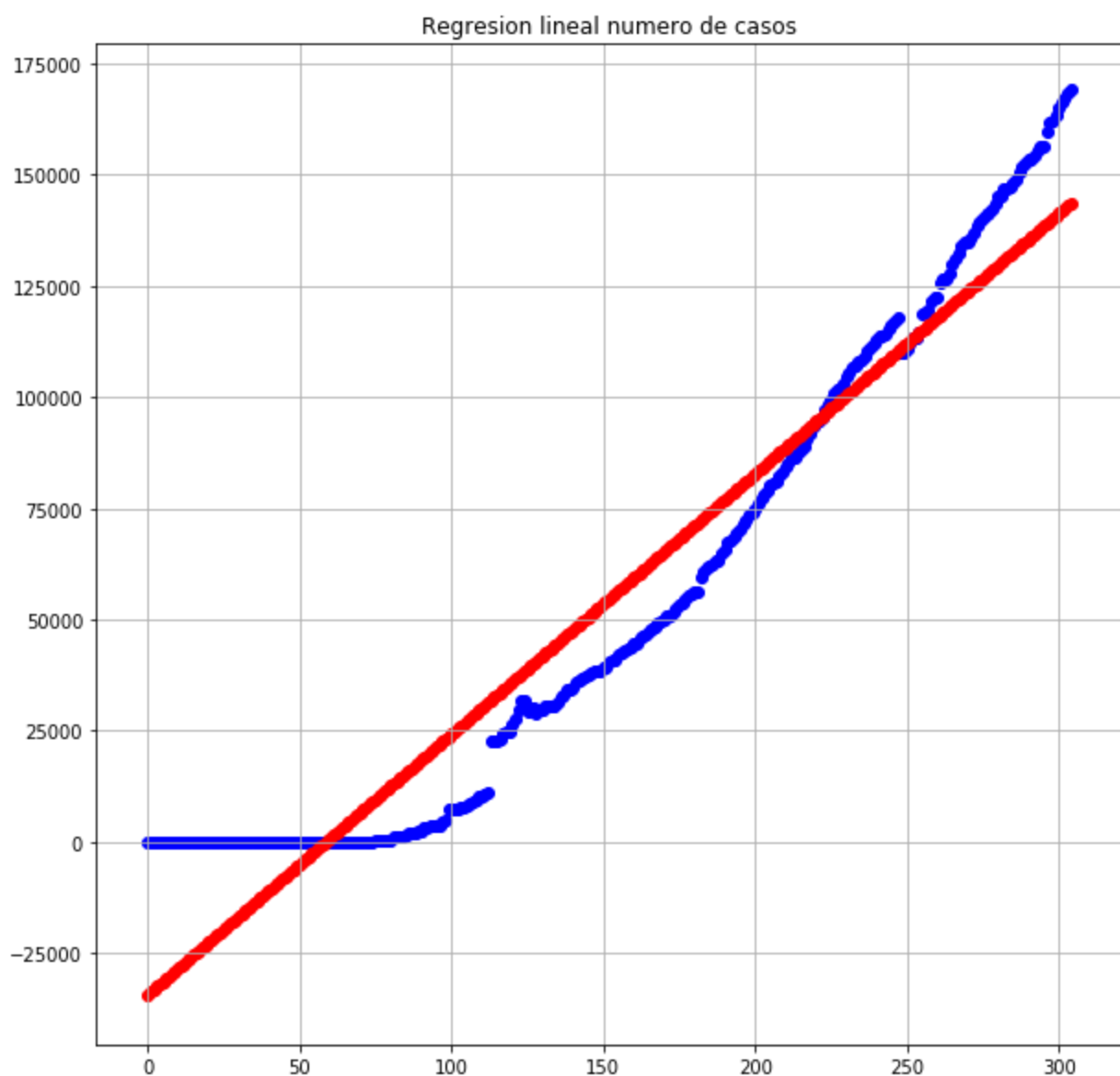


```
In [4]: import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model

modelo = linear_model.LinearRegression()
modelo.fit(x.reshape((-1,1)), y)
y_pred = modelo.predict(x.reshape((-1,1)))

plt.figure(figsize=(10, 10))
plt.scatter(x,y,color='blue')
plt.title("Regresion lineal numero de casos")
plt.scatter(x,y_pred,color='red')
plt.grid(True)
plt.show()

if (modelo.intercept_ < 0):
    ecua='y = {}x {}'
else:
    ecua='y = {}x + {}'
print("La ecuacion es la siguiente: ", ecua.format(modelo.coef_[0],modelo.intercept_))
```



La ecuacion es la siguiente: $y = 585.1653030841328x - 34266.57524911606$

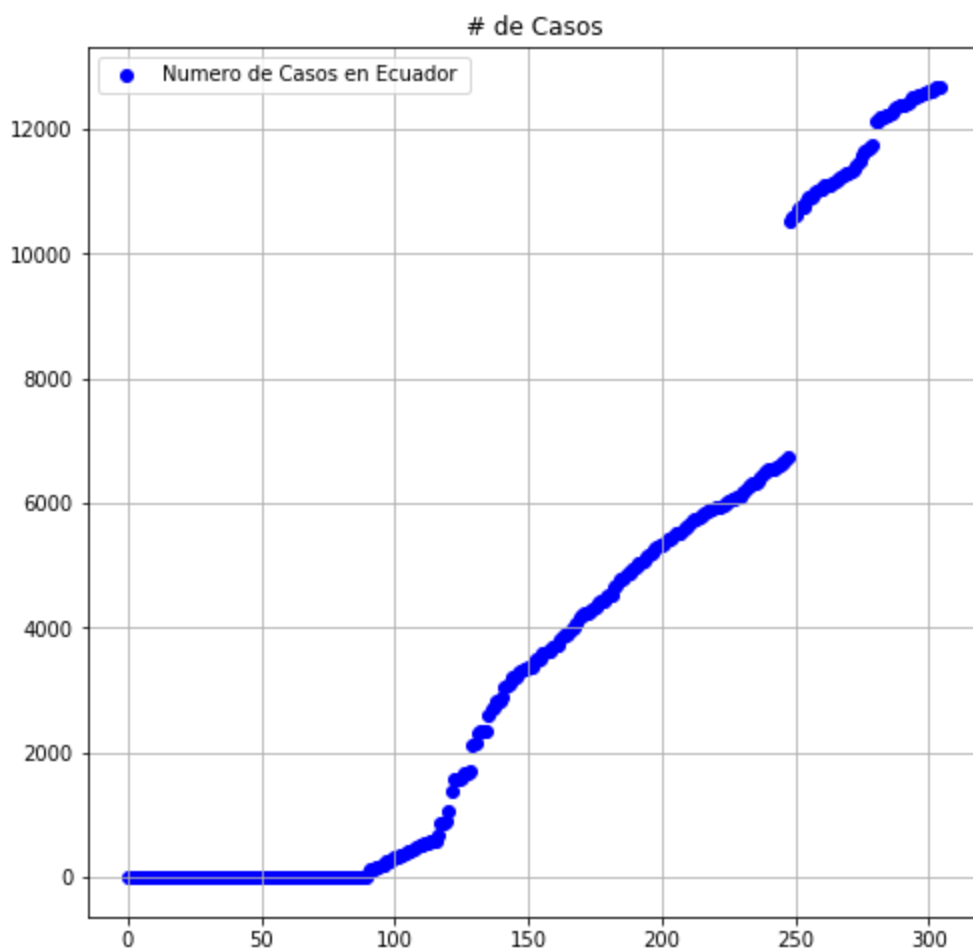
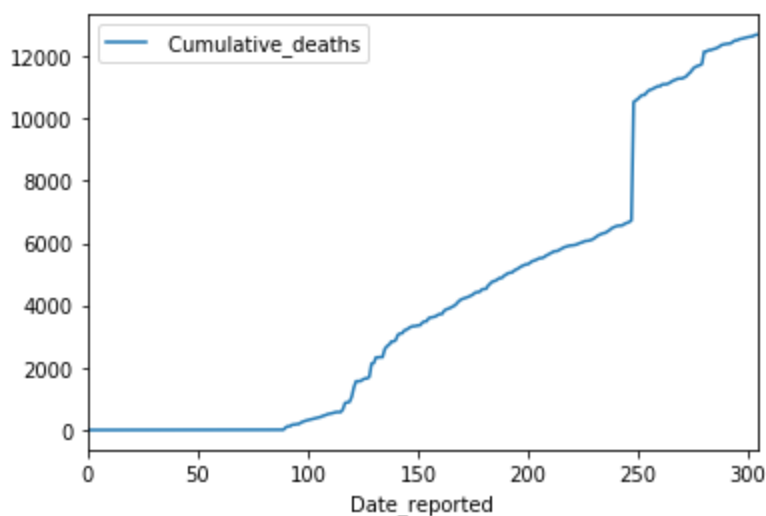
Grafica de personas muertas por COVID19 Ecuador

```
In [5]: url = 'covid.csv'
df = pd.read_csv(url)

df = df[df['Country'].isin(['Ecuador'])]
df = df.loc[:, ['Date_reported', 'Cumulative_deaths']]
FMT = '%Y-%m-%d'
date = df['Date_reported']
df['Date_reported'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-03-01", FMT)).days)
df
df.plot(x = 'Date_reported', y='Cumulative_deaths')

x = np.array(df.values[:,0])
y = np.array(df.values[:,1])

plt.figure(figsize=(8, 8))
plt.scatter(x,y,label='Numero de Casos en Ecuador ', color='blue')
#plt.scatter(x,y1,label='# de Muertes', color='red')
plt.grid(True)
plt.legend()
plt.title('# de Casos');
```

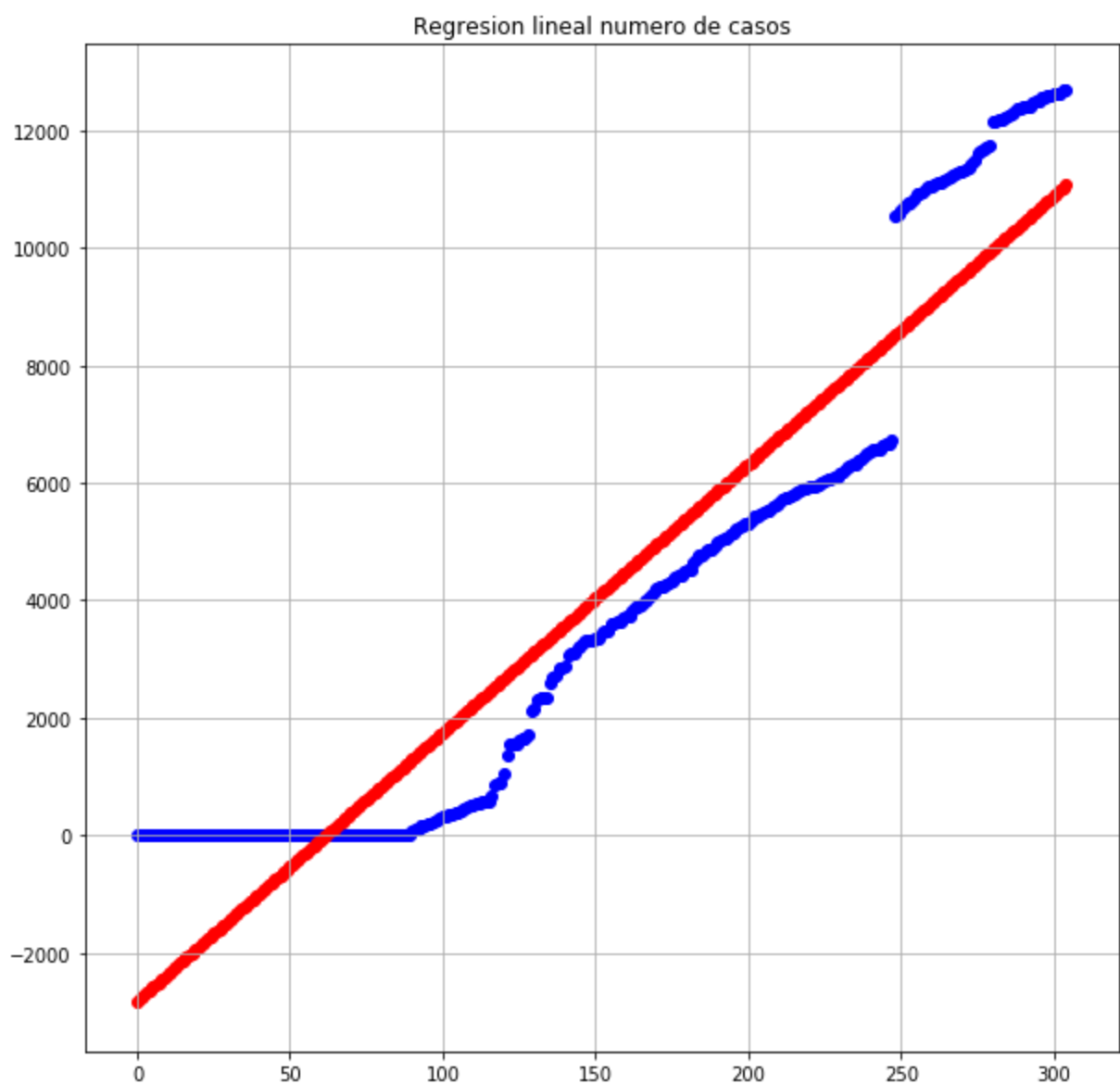


```
In [6]: import numpy as np #Librería numérica
import matplotlib.pyplot as plt
from sklearn import linear_model

modelo = linear_model.LinearRegression()
modelo.fit(x.reshape((-1,1)), y)
y_pred = modelo.predict(x.reshape((-1,1)))

plt.figure(figsize=(10, 10))
plt.scatter(x,y,color='blue')
plt.title("Regresion lineal numero de casos")
plt.scatter(x,y_pred,color='red')
plt.grid(True)
plt.show()

if (modelo.intercept_ < 0):
    ecua='y = {}x {}'
else:
    ecua='y = {}x + {}'
print("La ecuacion es la siguiente: ", ecua.format(modelo.coef_[0],modelo.intercept_))
```



La ecuacion es la siguiente: $y = 45.6448451166489x - 2812.6754741240757$

Conclusion

Gracias a los datos proporcionados por el estado se puede tener una muestra exacta de las personas infectadas por covid 19 en Ecuador