

MODELO PROBABILISTICO

AUTOR: CARLOS MOROCHO

Implementacion de un modelo probabilistico de infección por el virus Covid-19

Se realiza un análisis probabilistico simple del crecimiento de la infección en Python y el modelos para comprender mejor la evolución de la infección.

Se crea modelos de series temporales del número total de personas infectadas hasta la fecha (es decir, las personas realmente infectadas más las personas que han sido infectadas). Estos modelos tienen parámetros , que se estimarán por ajuste de probabilidad.

```
In [1]: # Importar las librerias para el analisis
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
```

```
In [2]: # Carga del dataset
df = pd.read_csv('datacovid.csv')
df
```

```
Out[2]:
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	to
0	ABW	North America	Aruba	2020-03-13	2.0	2.0	NaN	

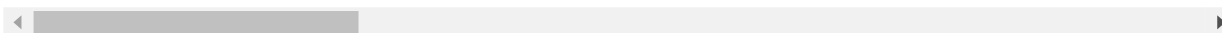
	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	to
1	ABW	North America	Aruba	2020-03-19	NaN	NaN	0.286	
2	ABW	North America	Aruba	2020-03-20	4.0	2.0	0.286	
3	ABW	North America	Aruba	2020-03-21	NaN	NaN	0.286	
4	ABW	North America	Aruba	2020-03-22	NaN	NaN	0.286	
5	ABW	North America	Aruba	2020-03-23	NaN	NaN	0.286	
6	ABW	North America	Aruba	2020-03-24	12.0	8.0	1.429	
7	ABW	North America	Aruba	2020-03-25	17.0	5.0	2.143	
8	ABW	North America	Aruba	2020-03-26	19.0	2.0	2.429	
9	ABW	North America	Aruba	2020-03-27	28.0	9.0	3.429	
10	ABW	North America	Aruba	2020-03-28	28.0	0.0	3.429	
11	ABW	North America	Aruba	2020-03-29	28.0	0.0	3.429	
12	ABW	North America	Aruba	2020-03-30	50.0	22.0	6.571	
13	ABW	North America	Aruba	2020-03-31	NaN	NaN	5.429	
14	ABW	North America	Aruba	2020-04-01	55.0	5.0	5.429	
15	ABW	North America	Aruba	2020-04-02	55.0	0.0	5.143	
16	ABW	North America	Aruba	2020-04-03	60.0	5.0	4.571	

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	to
17	ABW	North America	Aruba	2020-04-04	62.0	2.0	4.857	
18	ABW	North America	Aruba	2020-04-05	64.0	2.0	5.143	
19	ABW	North America	Aruba	2020-04-06	64.0	0.0	2.000	
20	ABW	North America	Aruba	2020-04-07	71.0	7.0	3.000	
21	ABW	North America	Aruba	2020-04-08	74.0	3.0	2.714	
22	ABW	North America	Aruba	2020-04-09	77.0	3.0	3.143	
23	ABW	North America	Aruba	2020-04-10	82.0	5.0	3.143	
24	ABW	North America	Aruba	2020-04-11	86.0	4.0	3.429	
25	ABW	North America	Aruba	2020-04-12	92.0	6.0	4.000	
26	ABW	North America	Aruba	2020-04-13	92.0	0.0	4.000	
27	ABW	North America	Aruba	2020-04-14	92.0	0.0	3.000	
28	ABW	North America	Aruba	2020-04-15	92.0	0.0	2.571	
29	ABW	North America	Aruba	2020-04-16	93.0	1.0	2.286	
...
54360	NaN	NaN	International	2020-10-05	696.0	NaN	NaN	
54361	NaN	NaN	International	2020-10-06	696.0	NaN	NaN	

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	to
54362	NaN	NaN	International	2020-10-07	696.0	NaN	NaN	
54363	NaN	NaN	International	2020-10-08	696.0	NaN	NaN	
54364	NaN	NaN	International	2020-10-09	696.0	NaN	NaN	
54365	NaN	NaN	International	2020-10-10	696.0	NaN	NaN	
54366	NaN	NaN	International	2020-10-11	696.0	NaN	NaN	
54367	NaN	NaN	International	2020-10-12	696.0	NaN	NaN	
54368	NaN	NaN	International	2020-10-13	696.0	NaN	NaN	
54369	NaN	NaN	International	2020-10-14	696.0	NaN	NaN	
54370	NaN	NaN	International	2020-10-15	696.0	NaN	NaN	
54371	NaN	NaN	International	2020-10-16	696.0	NaN	NaN	
54372	NaN	NaN	International	2020-10-17	696.0	NaN	NaN	
54373	NaN	NaN	International	2020-10-18	696.0	NaN	NaN	
54374	NaN	NaN	International	2020-10-19	696.0	NaN	NaN	
54375	NaN	NaN	International	2020-10-20	696.0	NaN	NaN	
54376	NaN	NaN	International	2020-10-21	696.0	NaN	NaN	
54377	NaN	NaN	International	2020-10-22	696.0	NaN	NaN	

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	to
54378	NaN	NaN	International	2020-10-23	696.0	NaN	NaN	
54379	NaN	NaN	International	2020-10-24	696.0	NaN	NaN	
54380	NaN	NaN	International	2020-10-25	696.0	NaN	NaN	
54381	NaN	NaN	International	2020-10-26	696.0	NaN	NaN	
54382	NaN	NaN	International	2020-10-27	696.0	NaN	NaN	
54383	NaN	NaN	International	2020-10-28	696.0	NaN	NaN	
54384	NaN	NaN	International	2020-10-29	696.0	NaN	NaN	
54385	NaN	NaN	International	2020-10-30	696.0	NaN	NaN	
54386	NaN	NaN	International	2020-10-31	696.0	NaN	NaN	
54387	NaN	NaN	International	2020-11-01	696.0	NaN	NaN	
54388	NaN	NaN	International	2020-11-02	696.0	NaN	NaN	
54389	NaN	NaN	International	2020-11-03	696.0	NaN	NaN	

54390 rows × 49 columns



Imprimos los resultados y agregamos el numero del dia

```
In [3]: df = df[df['location'].isin(['Ecuador'])] #Filtro la Informacion solo p
ara Ecuador
df = df[(df.total_cases >= 1)]
```

```
df = df.loc[:,['date', 'new_cases', 'total_cases']] #Selecciono las columnas de analisis
FMT = '%Y-%m-%d'
date = df['date']
df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-01-01", FMT)).days)
df
```

Out[3]:

	date	new_cases	total_cases
14430	60	1.0	1.0
14431	61	5.0	6.0
14432	62	1.0	7.0
14434	64	3.0	10.0
14435	65	3.0	13.0
14438	68	1.0	14.0
14439	69	1.0	15.0
14440	70	2.0	17.0
14443	73	6.0	23.0
14444	74	5.0	28.0
14445	75	9.0	37.0
14446	76	21.0	58.0
14447	77	53.0	111.0
14448	78	57.0	168.0
14449	79	31.0	199.0
14450	80	227.0	426.0
14451	81	106.0	532.0
14452	82	257.0	789.0
14453	83	192.0	981.0
14454	84	101.0	1082.0

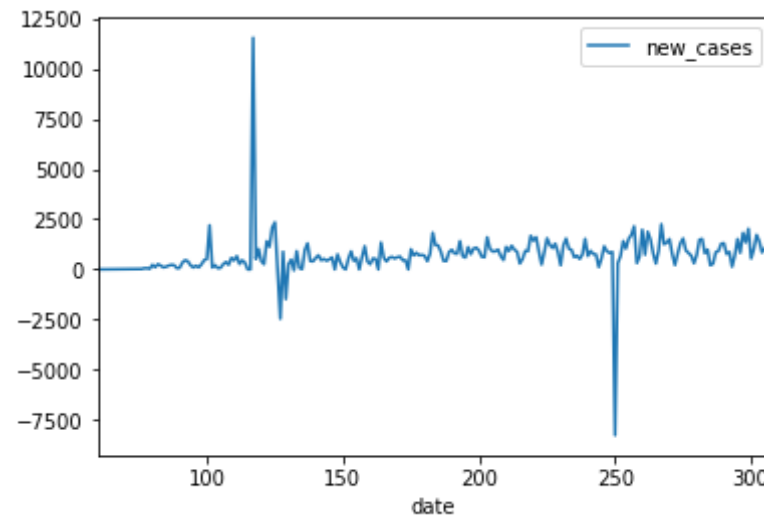
14454	84	129.0	1211.0
	date	new_cases	total_cases
14455	85	129.0	1211.0
14456	86	192.0	1403.0
14457	87	224.0	1627.0
14458	88	208.0	1835.0
14459	89	55.0	1890.0
14460	90	76.0	1966.0
14461	91	336.0	2302.0
14462	92	456.0	2758.0
14463	93	405.0	3163.0
14464	94	205.0	3368.0
...
14648	278	683.0	141034.0
14649	279	305.0	141339.0
14650	280	717.0	142056.0
14651	281	1475.0	143531.0
14652	282	1514.0	145045.0
14653	283	803.0	145848.0
14654	284	980.0	146828.0
14655	285	205.0	147033.0
14656	286	282.0	147315.0
14657	287	856.0	148171.0
14658	288	912.0	149083.0
14659	289	1277.0	150360.0
14660	290	1299.0	151659.0

14661	291	763.0	152422.0
	date	new_cases	total_cases
14662	292	867.0	153289.0
14663	293	134.0	153423.0
14664	294	692.0	154115.0
14665	295	1510.0	155625.0
14666	296	826.0	156451.0
14667	297	1819.0	158270.0
14668	298	1344.0	159614.0
14669	299	2021.0	161635.0
14670	300	543.0	162178.0
14671	301	1014.0	163192.0
14672	302	1716.0	164908.0
14673	303	1394.0	166302.0
14674	304	845.0	167147.0
14675	305	1045.0	168192.0
14676	306	1002.0	169194.0
14677	307	368.0	169562.0

243 rows × 3 columns

In [4]: `df.plot(x='date', y='new_cases')`

Out[4]: `<matplotlib.axes._subplots.AxesSubplot at 0x18484959160>`



Ahora podemos analizar un modelo probabilístico para el examen.

El modelo basado en probabilidad

Para realizar una estimación del factor de crecimiento de los casos de Covid 19 en Ecuador calculamos la mediana, con esto obtenemos el valor medio de crecimiento de un conjunto de datos, con esto podemos obtener un factor de crecimiento o tasa de crecimiento de los nuevos casos.

```
In [5]: filtro = df["new_cases"] # Filtro los datos que se empezó a tener casos
#Obtenemos la mediana
media = filtro.mean()
mediana = filtro.median()
print(mediana)
print(media)
```

```
660.0
697.7860082304527
```

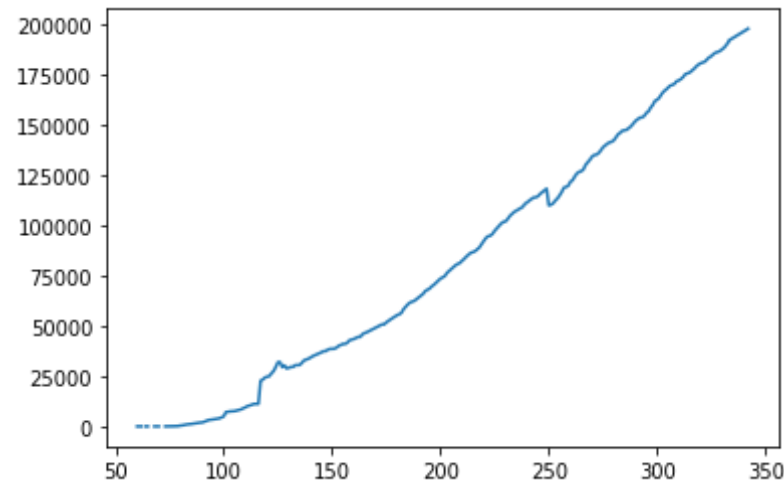
De la ecuación de la recta $y = mX + b$ nuestra pendiente «m» es el coeficiente y el término

independiente «b»

```
In [6]: #Vamos a comprobar:
# según la media y la mediana podemos obtener la tasa de crecimiento y
# predecir su comportamiento.
# Cargamos los datos de total de casos
url = 'https://covid.ourworldindata.org/data/ecdc/total_cases.csv'
df_t = pd.read_csv(url)
FMT = '%Y-%m-%d'
date = df_t['date']
df_t['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-01-01", FMT)).days)
df_t = df_t.loc[:, ['date', 'Ecuador']] #Selecciono las columnas de analisis
y = list(df_t.iloc[:, 1]) # Total casos
x = list(df_t.iloc[:, 0]) # Dias
#Realizamos un ejemplo de prediccion
prediccion_siguiente = int(y[-1] + mediana)
print(prediccion_siguiente)
```

191569

```
In [7]: # Quiero predecir cuántos "Casos" voy a obtener de aqui a 10 dias.
for i in range(x[-1], x[-1]+10):
    x.append(i)
    y.append(int(y[-1] + mediana))
plt.plot(x[61:], y[61:])
plt.show()
```



Practica

1. Comparar el modelo de predicción matemático vs probabilidad.
2. Retroceder un semana y comparar el modelo matemático vs probabilidad vs reales. Solo cargan los datos para generar los modelos menos 7 días.

Puntos extras: Investigas sobre la correlación de variables y aplicar el cálculo en base a los datos del Ecuador.

1. Comparar el modelo de predicción matemático vs probabilidad.

```
In [8]: #Implementar
# predicción probabilidad
x = list(df.iloc[:, 0]) # Dias
y = list(df.iloc[:, 2]) # Total casos
#Realizamos un ejemplo de predicción
predicción_siguiente = int(y[-1] + mediana)
```

```

plt.plot(x[-1],y[-1],'o',label='actual')

#prediccion a una semana
for i in range(x[-1], x[-1]+8):
    x.append(i)
    y.append(int(y[-1] + mediana))

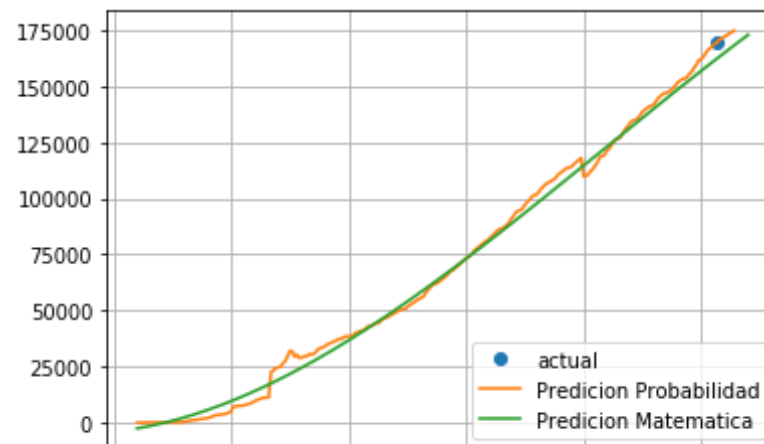
#prediccion matematica
x1 = np.array(x)
y1 = np.array(y)
def func_polinomial(x, a, b, c, d):
    return a*x**4 + b*x**3 + c*x**2 + d*x + 1

popt1, pcov1 = curve_fit(func_polinomial, x1, y1)
pred_x = list(range(min(x1),max(x1)+7))

pred_x = np.array(pred_x, dtype=float)
#grficamos los dos metodos para analizar

plt.plot(x, y,label='Prediccion Probabilidad')
plt.plot(pred_x,func_polinomial(pred_x,*popt1),label='Prediccion Matematica')
plt.legend(loc='lower right')
plt.grid(True)
plt.show()

```



2. Retroceder un semana y comparar el modelo matematico vs probabilidad vs reales. Solo cargan los datos para generar los modelos menos 7 días.

```
In [9]: df2 = pd.read_csv('datacovid.csv')

df2 = df2[df2['location'].isin(['Ecuador'])] #Filtro la Informacion solo para Ecuador
df2 = df2[(df2.total_cases >= 1)]
df2 = df2.loc[:, ['date', 'new_cases', 'total_cases']] #Selecciono las columnas de analisis
FMT = '%Y-%m-%d'
date = df2['date']

df2['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-01-01", FMT)).days)

data = df2[:-7]

#Modelo probablistico
filtrol = data["new_cases"] # Filtro los datos que se empezo a tener casos
#Obtenemos la mediana
medial = filtrol.mean()
mediana1 = filtrol.median()

y1 = list(data.iloc[:, 2]) # Total casos
x1 = list(data.iloc[:, 0]) # Dias

#Realizamos un ejemplo de prediccion
prediccion_siguietel = int(y1[-1] + mediana1)
print(prediccion_siguietel)

for i in range(x1[-1], x1[-1]+8):
    x1.append(i)
```

```

        y1.append(int(y1[-1] + mediana))

#modelo matematico
x1 = np.array(x1, dtype=float)
y1 = np.array(y1, dtype=float)
def func_polinomial(x, a, b, c, d):
    return a*x**4 + b*x**3 + c*x**2 + d*x + 1

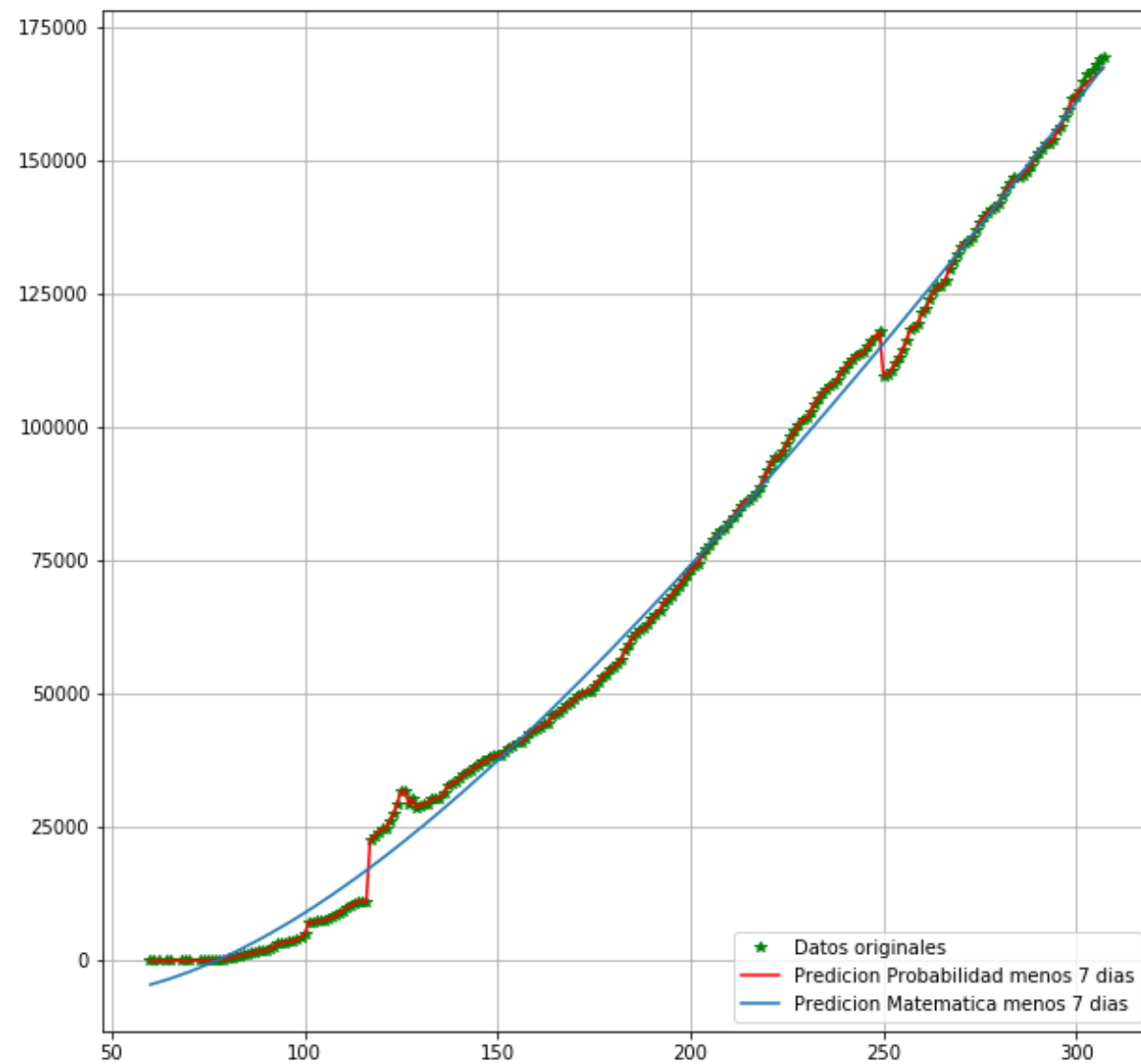
popt1, pcov1 = curve_fit(func_polinomial, x1, y1)

x2 = df2.date
y2 = df2.total_cases

plt.figure(figsize=(10,10))
plt.plot(x2, y2, 'g*', label='Datos originales')
plt.plot(x1, y1, color='r', label='Prediccion Probabilidad menos 7 dias')
plt.plot(x1, func_polinomial(x1, *popt1), label='Prediccion Matematica menos 7 dias')
plt.legend(loc='lower right')
plt.grid(True)
plt.show()

```

162816



Analisis

En el segundo punto, al cargar la información con menos 7 días, de los datos originales y luego hacer uso de los dos métodos tanto matemático como el de probabilidad y prediciendo 7 días más podemos observar que ambos nos dan una aproximación similar en datos y al comparar y graficar los datos originales se observa que con la proyección se obtiene un número más bajo de infectados que son los que se tiene en la actualidad, el número de infectados es de 162816 mil casos.

Conclusiones

Estos modelos deben ser usados como referencia y con mas datos se puede lograr una predicción mucho mayor, se recomienda el investigar sobre nuevas técnicas o combinaciones que mejoren los resultados del modelo.

Criterio personal (Político, económico y social de la situación)

Debido a todo el caos que provoco el virus en todo el mundo, estos modelos de prediccon que si bien no son del todo exactos pueden ayudarnos a solventar algunas respuestas que todo la sociedad esta buscando, existen millones de empresas que de seguro estan desarrollando estos sistemas debido a la contingencia pero creo que ningunoo veneficia a la sociedad como la informacion como tal.