Covid-19 infección en Ecuador. Modelos probabilisticos

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 analasis
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]:

```
# Actualizar los datos (URL)
url = 'https://covid.ourworldindata.org/data/ecdc/new_cases.csv'
df = pd.read_csv(url)
df
```

Out[2]:

	date	World	Afghanistan	Albania	Algeria	Andorra	Angola	Anguilla	Antigua and Barbuda	Argent
0	2019-12-31	27	0.0	NaN	0.0	NaN	NaN	NaN	NaN	N
1	2020-01-01	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	N
2	2020-01-02	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	N
3	2020-01-03	17	0.0	NaN	0.0	NaN	NaN	NaN	NaN	N
4	2020-01-04	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	N
...	...	...	...	...	...	...	...	...	...	
321	2020-11-16	504530	163.0	597.0	860.0	147.0	146.0	0.0	0.0	564
322	2020-11-17	557581	65.0	602.0	910.0	42.0	77.0	0.0	0.0	789
323	2020-11-18	580506	383.0	694.0	1002.0	37.0	164.0	0.0	0.0	1062
324	2020-11-19	623239	0.0	711.0	1038.0	67.0	203.0	0.0	5.0	1033
325	2020-11-20	594842	282.0	786.0	1023.0	48.0	104.0	0.0	0.0	1011

326 rows × 216 columns

Imprimos los resultados y agregamos el numero del dia

In [3]:

```
df = df.loc[:,['date','Ecuador']] #Selecciono las columnas de analasis
# Expresar las fechas en numero de dias desde el 01 Enero
FMT = '%Y-%m-%d'
date = df['date']
df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-
df
```

Out[3]:

	date	Ecuador
0	-1	0.0
1	0	0.0
2	1	0.0
3	2	0.0
4	3	0.0
...	...	...
321	320	668.0
322	321	381.0
323	322	428.0
324	323	1146.0
325	324	996.0

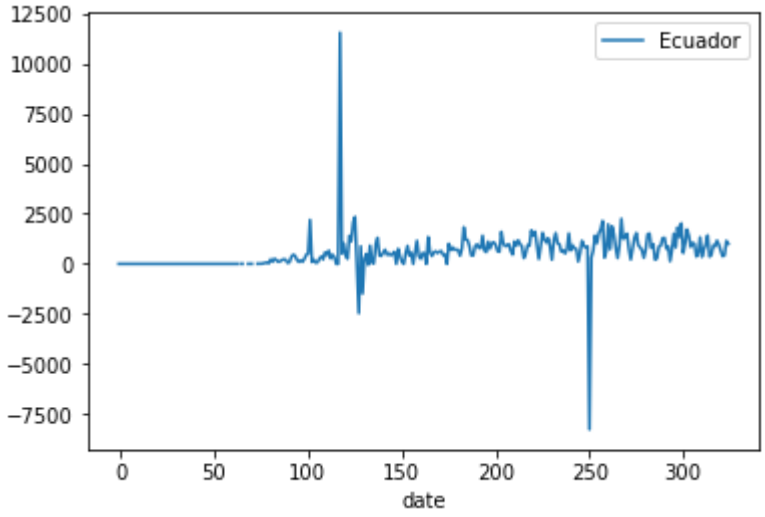
326 rows × 2 columns

In [4]:

```
1 df.plot(x ='date', y='Ecuador')
```

Out[4]:

```
<AxesSubplot:xlabel='date'>
```



Ahora podemos analizar un modelo probabilisto para el examen.

El modelo basado en probabilidad

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

In [5]:

```
1 filtro = df["Ecuador"][61:] # Filtro los datos que se empezo a tener casos
2 #Obtenemos la mediana
3 media = filtro.mean()
4 mediana = filtro.median()
5 print(mediana)
6 print(media)
7
```

```
666.5
704.7923076923076
```

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

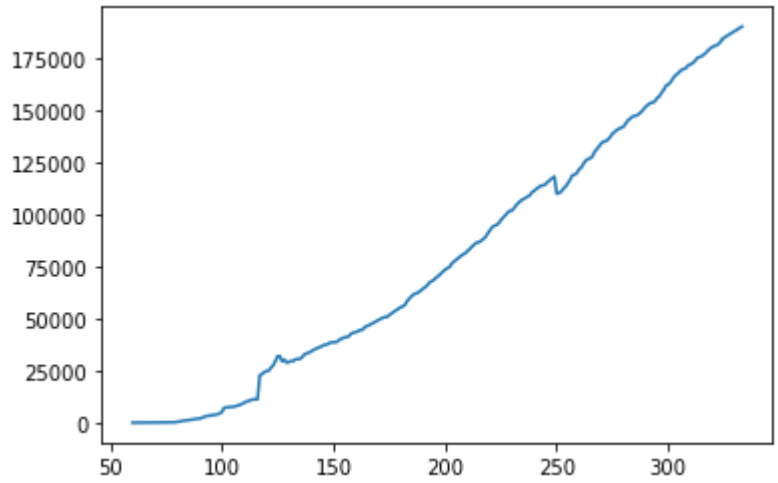
In [6]:

```
#Vamos a comprobar:
# según la media y la mediana podemos obtener la taza de crecieminto y predicir su comp
# Cargamos los datos de total de casos
url = 'https://covid.ourworldindata.org/data/ecdc/total_cases.csv'
df_t = pd.read_csv(url).fillna(0)
FMT = '%Y-%m-%d'
date = df_t['date']
df_t['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("202(
df_t = df_t.loc[:,['date','Ecuador']] #Selecciono las columnas de analasis
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)
```

```
183912
```

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. COMPARACION DEL MODELO DE PREDICCION MATEMÁTICO VS PROBABILIDAD

Para la comparacion se utilizara el modelo polinomial

In [8]:

```
1 import numpy as np #Librería numérica
2 import matplotlib.pyplot as plt
```

In [33]:

```
df = pd.read_csv('covidDatos.csv').fillna(0) # poniendo datos nan a cero
ndf= df.loc[(df['location'] == 'Ecuador') & (df['total_cases'] != 0)] # filtrando por p
ndf1=ndf[['date','total_cases']]
x=np.arange(1,len(ndf1)+1,1, dtype='float') # arreglo de x lo creo para simular el nume
y=np.array(ndf1.values[:,1], dtype='float')


funcion = np.poly1d(np.polyfit(x,y, 4))

y_pred = funcion(x)
print(funcion)
for i , a in zip(x ,y_pred):
    print('dia: ', i ,' --> prediccion: ',int(a))

plt.figure(figsize=(8, 8))
plt.scatter(x, y, color = "yellow")
plt.plot(x, y_pred, color='green')
plt.show()
```

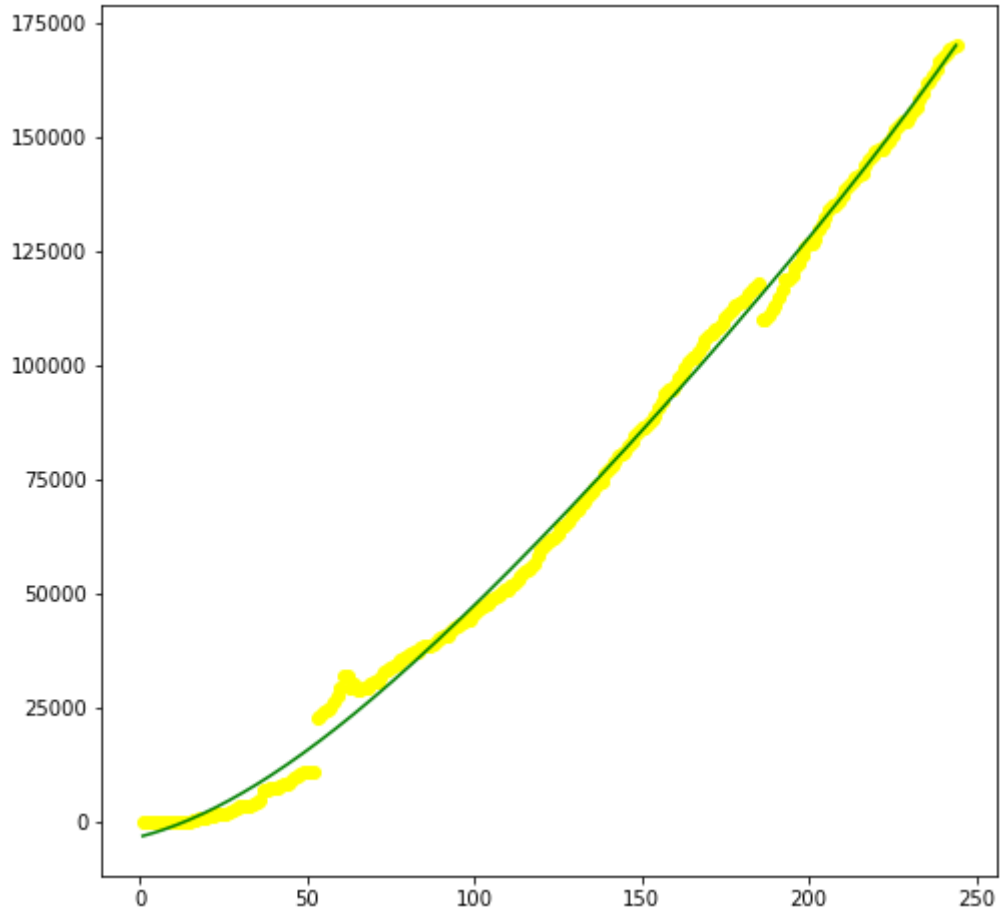
```
          4           3         2
2.806e-05 x - 0.01701 x + 4.631 x + 185 x - 3223
dia:  1.0  --> prediccion:  -3033
dia:  2.0  --> prediccion:  -2834
dia:  3.0  --> prediccion:  -2626
dia:  4.0  --> prediccion:  -2409
dia:  5.0  --> prediccion:  -2184
dia:  6.0  --> prediccion:  -1949
dia:  7.0  --> prediccion:  -1706
dia:  8.0  --> prediccion:  -1455
dia:  9.0  --> prediccion:  -1195
dia:  10.0  --> prediccion:  -926
dia:  11.0  --> prediccion:  -649
dia:  12.0  --> prediccion:  -364
dia:  13.0  --> prediccion:  -71
dia:  14.0  --> prediccion:  228
dia:  15.0  --> prediccion:  537
dia:  16.0  --> prediccion:  854
dia:  17.0  --> prediccion:  1178
dia:  18.0  --> prediccion:  1510
dia:  19.0  --> prediccion:  1850
dia:  20.0  --> prediccion:  2197
dia:  21.0  --> prediccion:  2551
dia:  22.0  --> prediccion:  2913
dia:  23.0  --> prediccion:  3282
dia:  24.0  --> prediccion:  3658
dia:  25.0  --> prediccion:  4041
dia:  26.0  --> prediccion:  4430
dia:  27.0  --> prediccion:  4827
dia:  28.0  --> prediccion:  5230
dia:  29.0  --> prediccion:  5641
dia:  30.0  --> prediccion:  6057
dia:  31.0  --> prediccion:  6480
dia:  32.0  --> prediccion:  6910
dia:  33.0  --> prediccion:  7346
dia:  34.0  --> prediccion:  7788
dia:  35.0  --> prediccion:  8236
dia:  36.0  --> prediccion:  8691
```

```
dia:  37.0  --> prediccion:  9151
dia:  38.0  --> prediccion:  9618
dia:  39.0  --> prediccion:  10090
dia:  40.0  --> prediccion:  10568
dia:  41.0  --> prediccion:  11052
dia:  42.0  --> prediccion:  11541
dia:  43.0  --> prediccion:  12037
dia:  44.0  --> prediccion:  12537
dia:  45.0  --> prediccion:  13043
dia:  46.0  --> prediccion:  13554
dia:  47.0  --> prediccion:  14071
dia:  48.0  --> prediccion:  14593
dia:  49.0  --> prediccion:  15120
dia:  50.0  --> prediccion:  15652
dia:  51.0  --> prediccion:  16189
dia:  52.0  --> prediccion:  16730
dia:  53.0  --> prediccion:  17277
dia:  54.0  --> prediccion:  17829
dia:  55.0  --> prediccion:  18385
dia:  56.0  --> prediccion:  18946
dia:  57.0  --> prediccion:  19512
dia:  58.0  --> prediccion:  20082
dia:  59.0  --> prediccion:  20656
dia:  60.0  --> prediccion:  21235
dia:  61.0  --> prediccion:  21819
dia:  62.0  --> prediccion:  22406
dia:  63.0  --> prediccion:  22998
dia:  64.0  --> prediccion:  23594
dia:  65.0  --> prediccion:  24195
dia:  66.0  --> prediccion:  24799
dia:  67.0  --> prediccion:  25407
dia:  68.0  --> prediccion:  26019
dia:  69.0  --> prediccion:  26635
dia:  70.0  --> prediccion:  27255
dia:  71.0  --> prediccion:  27878
dia:  72.0  --> prediccion:  28506
dia:  73.0  --> prediccion:  29137
dia:  74.0  --> prediccion:  29771
dia:  75.0  --> prediccion:  30409
dia:  76.0  --> prediccion:  31051
dia:  77.0  --> prediccion:  31696
dia:  78.0  --> prediccion:  32345
dia:  79.0  --> prediccion:  32996
dia:  80.0  --> prediccion:  33651
dia:  81.0  --> prediccion:  34310
dia:  82.0  --> prediccion:  34971
dia:  83.0  --> prediccion:  35636
dia:  84.0  --> prediccion:  36304
dia:  85.0  --> prediccion:  36975
dia:  86.0  --> prediccion:  37649
dia:  87.0  --> prediccion:  38326
dia:  88.0  --> prediccion:  39005
dia:  89.0  --> prediccion:  39688
dia:  90.0  --> prediccion:  40374
dia:  91.0  --> prediccion:  41062
dia:  92.0  --> prediccion:  41753
dia:  93.0  --> prediccion:  42447
dia:  94.0  --> prediccion:  43144
dia:  95.0  --> prediccion:  43843
dia:  96.0  --> prediccion:  44544
dia:  97.0  --> prediccion:  45249
```

```
dia:  98.0  --> prediccion:  45956
dia:  99.0  --> prediccion:  46665
dia:  100.0  --> prediccion:  47377
dia:  101.0  --> prediccion:  48091
dia:  102.0  --> prediccion:  48808
dia:  103.0  --> prediccion:  49527
dia:  104.0  --> prediccion:  50248
dia:  105.0  --> prediccion:  50971
dia:  106.0  --> prediccion:  51697
dia:  107.0  --> prediccion:  52425
dia:  108.0  --> prediccion:  53156
dia:  109.0  --> prediccion:  53888
dia:  110.0  --> prediccion:  54623
dia:  111.0  --> prediccion:  55359
dia:  112.0  --> prediccion:  56098
dia:  113.0  --> prediccion:  56839
dia:  114.0  --> prediccion:  57582
dia:  115.0  --> prediccion:  58326
dia:  116.0  --> prediccion:  59073
dia:  117.0  --> prediccion:  59822
dia:  118.0  --> prediccion:  60573
dia:  119.0  --> prediccion:  61325
dia:  120.0  --> prediccion:  62080
dia:  121.0  --> prediccion:  62836
dia:  122.0  --> prediccion:  63594
dia:  123.0  --> prediccion:  64354
dia:  124.0  --> prediccion:  65116
dia:  125.0  --> prediccion:  65879
dia:  126.0  --> prediccion:  66645
dia:  127.0  --> prediccion:  67412
dia:  128.0  --> prediccion:  68181
dia:  129.0  --> prediccion:  68951
dia:  130.0  --> prediccion:  69723
dia:  131.0  --> prediccion:  70497
dia:  132.0  --> prediccion:  71273
dia:  133.0  --> prediccion:  72050
dia:  134.0  --> prediccion:  72829
dia:  135.0  --> prediccion:  73609
dia:  136.0  --> prediccion:  74391
dia:  137.0  --> prediccion:  75175
dia:  138.0  --> prediccion:  75960
dia:  139.0  --> prediccion:  76747
dia:  140.0  --> prediccion:  77536
dia:  141.0  --> prediccion:  78326
dia:  142.0  --> prediccion:  79117
dia:  143.0  --> prediccion:  79911
dia:  144.0  --> prediccion:  80705
dia:  145.0  --> prediccion:  81502
dia:  146.0  --> prediccion:  82299
dia:  147.0  --> prediccion:  83099
dia:  148.0  --> prediccion:  83900
dia:  149.0  --> prediccion:  84702
dia:  150.0  --> prediccion:  85506
dia:  151.0  --> prediccion:  86311
dia:  152.0  --> prediccion:  87118
dia:  153.0  --> prediccion:  87927
dia:  154.0  --> prediccion:  88737
dia:  155.0  --> prediccion:  89548
dia:  156.0  --> prediccion:  90362
dia:  157.0  --> prediccion:  91176
dia:  158.0  --> prediccion:  91992
```

```
dia:  159.0  --> prediccion:  92810
dia:  160.0  --> prediccion:  93629
dia:  161.0  --> prediccion:  94450
dia:  162.0  --> prediccion:  95272
dia:  163.0  --> prediccion:  96096
dia:  164.0  --> prediccion:  96922
dia:  165.0  --> prediccion:  97749
dia:  166.0  --> prediccion:  98577
dia:  167.0  --> prediccion:  99408
dia:  168.0  --> prediccion:  100240
dia:  169.0  --> prediccion:  101073
dia:  170.0  --> prediccion:  101908
dia:  171.0  --> prediccion:  102745
dia:  172.0  --> prediccion:  103583
dia:  173.0  --> prediccion:  104423
dia:  174.0  --> prediccion:  105265
dia:  175.0  --> prediccion:  106108
dia:  176.0  --> prediccion:  106953
dia:  177.0  --> prediccion:  107800
dia:  178.0  --> prediccion:  108649
dia:  179.0  --> prediccion:  109499
dia:  180.0  --> prediccion:  110351
dia:  181.0  --> prediccion:  111205
dia:  182.0  --> prediccion:  112061
dia:  183.0  --> prediccion:  112919
dia:  184.0  --> prediccion:  113778
dia:  185.0  --> prediccion:  114639
dia:  186.0  --> prediccion:  115503
dia:  187.0  --> prediccion:  116368
dia:  188.0  --> prediccion:  117235
dia:  189.0  --> prediccion:  118104
dia:  190.0  --> prediccion:  118975
dia:  191.0  --> prediccion:  119848
dia:  192.0  --> prediccion:  120723
dia:  193.0  --> prediccion:  121600
dia:  194.0  --> prediccion:  122480
dia:  195.0  --> prediccion:  123361
dia:  196.0  --> prediccion:  124245
dia:  197.0  --> prediccion:  125130
dia:  198.0  --> prediccion:  126018
dia:  199.0  --> prediccion:  126909
dia:  200.0  --> prediccion:  127801
dia:  201.0  --> prediccion:  128696
dia:  202.0  --> prediccion:  129593
dia:  203.0  --> prediccion:  130493
dia:  204.0  --> prediccion:  131395
dia:  205.0  --> prediccion:  132299
dia:  206.0  --> prediccion:  133206
dia:  207.0  --> prediccion:  134116
dia:  208.0  --> prediccion:  135028
dia:  209.0  --> prediccion:  135942
dia:  210.0  --> prediccion:  136860
dia:  211.0  --> prediccion:  137779
dia:  212.0  --> prediccion:  138702
dia:  213.0  --> prediccion:  139627
dia:  214.0  --> prediccion:  140556
dia:  215.0  --> prediccion:  141487
dia:  216.0  --> prediccion:  142421
dia:  217.0  --> prediccion:  143357
dia:  218.0  --> prediccion:  144297
dia:  219.0  --> prediccion:  145240
```

```
dia:  220.0  --> prediccion:  146186
dia:  221.0  --> prediccion:  147135
dia:  222.0  --> prediccion:  148087
dia:  223.0  --> prediccion:  149042
dia:  224.0  --> prediccion:  150001
dia:  225.0  --> prediccion:  150963
dia:  226.0  --> prediccion:  151928
dia:  227.0  --> prediccion:  152896
dia:  228.0  --> prediccion:  153868
dia:  229.0  --> prediccion:  154844
dia:  230.0  --> prediccion:  155823
dia:  231.0  --> prediccion:  156805
dia:  232.0  --> prediccion:  157792
dia:  233.0  --> prediccion:  158782
dia:  234.0  --> prediccion:  159775
dia:  235.0  --> prediccion:  160773
dia:  236.0  --> prediccion:  161774
dia:  237.0  --> prediccion:  162780
dia:  238.0  --> prediccion:  163789
dia:  239.0  --> prediccion:  164802
dia:  240.0  --> prediccion:  165820
dia:  241.0  --> prediccion:  166841
dia:  242.0  --> prediccion:  167867
dia:  243.0  --> prediccion:  168897
dia:  244.0  --> prediccion:  169932
```



UTILIZACION DE MODELO PROBABILISTICO

In [34]:

```
media = ndf1.values[:,1].mean()
mediana = np.median(ndf1.values[:,1])
print("MEDIA: ",media)
print("MEDIANA: ",mediana)
```

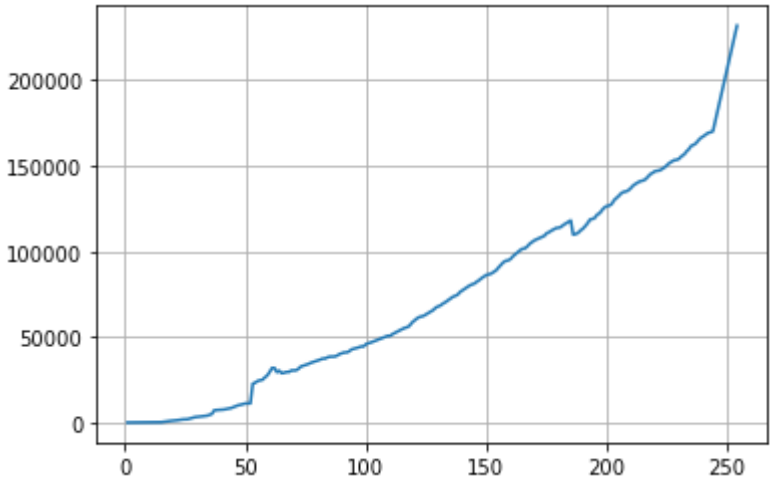
```
MEDIA:  69707.6844262295
MEDIANA:  61746.5
```

In [39]:

```
for it in range(int(x[-1]), int(x[-1]+10)):
    x_pro=np.append(x,(it+1))
    y_pro=np.append(y,y[-1] + mediana)
plt.plot(x_pro,y_pro)
plt.grid(True)
plt.show()


tablePrueba = []
print(tabulate(tablePrueba))
for i,a in zip(x_pro,y_pro):
    tablePrueba.append([i,a])
print(tabulate(tablePrueba, headers=["Dia","Probabilistico"]))

```



Dia	Probabilistico
1	1
2	6
3	7
4	10
5	13
6	14
7	15
8	17
9	23
10	28
11	37
12	58
13	111
14	168
15	199
16	426
17	532
18	789
19	981
20	1082
21	1211
22	1403
23	1627
24	1835
25	1890

26	1966
27	2302
28	2758
29	3163
30	3368
31	3465
32	3646
33	3747
34	3995
35	4450
36	4965
37	7161
38	7257
39	7466
40	7529
41	7603
42	7858
43	8225
44	8450
45	9022
46	9468
47	10128
48	10398
49	10850
50	11183
51	11183
52	11183
53	22719
54	23240
55	24258
56	24675
57	24934
58	26336
59	27464
60	29538
61	31881
62	31881
63	29420
64	30298
65	28818
66	29071
67	29559
68	29509
69	30419
70	30486
71	30502
72	31467
73	32763
74	33182
75	33582
76	34151
77	34854
78	35306
79	35828
80	36258
81	36756
82	37355
83	37355
84	38103
85	38471
86	38571

87	38571
88	39098
89	39994
90	40414
91	40966
92	40966
93	41575
94	42728
95	43120
96	43378
97	43917
98	44440
99	44440
100	45778
101	46356
102	46751
103	47322
104	47943
105	48490
106	49097
107	49731
108	50183
109	50640
110	50640
111	51643
112	52334
113	53156
114	53856
115	54574
116	55255
117	55665
118	56432
119	58257
120	59468
121	60657
122	61535
123	61958
124	62380
125	63245
126	64221
127	65018
128	65801
129	67209
130	67870
131	68459
132	69570
133	70329
134	71365
135	72444
136	73382
137	74013
138	74620
139	76217
140	77257
141	78148
142	79049
143	80036
144	80694
145	81161
146	82279
147	83193

148	84370
149	85355
150	86232
151	86524
152	87041
153	87963
154	88866
155	90537
156	91969
157	93572
158	94459
159	94701
160	95563
161	97110
162	98343
163	99409
164	100688
165	101542
166	101751
167	102941
168	104475
169	105508
170	106481
171	107089
172	107769
173	108289
174	109030
175	110549
176	111219
177	112141
178	112906
179	113648
180	113767
181	114309
182	115457
183	116360
184	117175
185	118045
186	109784
187	110092
188	110757
189	112166
190	113206
191	114732
192	116451
193	118594
194	118911
195	119553
196	121525
197	122257
198	124129
199	125620
200	126419
201	126711
202	127643
203	129892
204	131146
205	132475
206	133981
207	134747
208	134965

209	135749
210	137047
211	138584
212	139534
213	140351
214	141034
215	141339
216	142056
217	143531
218	145045
219	145848
220	146828
221	147033
222	147315
223	148171
224	149083
225	150360
226	151659
227	152422
228	153289
229	153423
230	154115
231	155625
232	156451
233	158270
234	159614
235	161635
236	162178
237	163192
238	164908
239	166302
240	167147
241	168192
242	169194
243	169562
244	170110
254	231856

In [40]:

```
#!pip install tabulate
y_pred=np.array([])
y_pred=funcion(x)
from tabulate import tabulate
table = []
print(tabulate(table))
for i,a,b in zip(x_pro,y_pro,y_pred):
    table.append([i,a,round(b,2)])
print(tabulate(table, headers=["Dia","Probabilistico", "Polinomial"]))

```

```
  Dia    Probabilistico    Polinomial
-----  ----------------  ------------
    1                 1      -3033.07
    2                 6      -2834.32
    3                 7      -2626.52
    4                10      -2409.76
    5                13      -2184.14
    6                14      -1949.76
    7                15      -1706.72
    8                17      -1455.11
    9                23      -1195.04
   10                28       -926.59
   11                37       -649.87
   12                58       -364.98
   13               111        -71.99
   14               168        228.99
   15               199        537.86
   16               426        854.54
   17               532       1178.94
   18               789       1510.96
   19               981       1850.52
   20              1082       2197.52
   21              1211       2551.87
   22              1403       2913.49
   23              1627       3282.29
   24              1835       3658.19
   25              1890       4041.08
   26              1966       4430.9
   27              2302       4827.56
   28              2758       5230.97
   29              3163       5641.04
   30              3368       6057.7
   31              3465       6480.86
   32              3646       6910.45
   33              3747       7346.37
   34              3995       7788.56
   35              4450       8236.92
   36              4965       8691.39
   37              7161       9151.88
   38              7257       9618.32
   39              7466      10090.6
   40              7529      10568.7
   41              7603      11052.5
   42              7858      11542
   43              8225      12037
   44              8450      12537.5
```

45	9022	13043.5
46	9468	13554.8
47	10128	14071.4
48	10398	14593.1
49	10850	15120.1
50	11183	15652.1
51	11183	16189.1
52	11183	16731
53	22719	17277.8
54	23240	17829.3
55	24258	18385.7
56	24675	18946.7
57	24934	19512.2
58	26336	20082.4
59	27464	20657
60	29538	21236
61	31881	21819.3
62	31881	22407
63	29420	22998.8
64	30298	23594.9
65	28818	24195
66	29071	24799.2
67	29559	25407.4
68	29509	26019.5
69	30419	26635.5
70	30486	27255.4
71	30502	27879
72	31467	28506.3
73	32763	29137.2
74	33182	29771.8
75	33582	30409.9
76	34151	31051.5
77	34854	31696.6
78	35306	32345.1
79	35828	32996.9
80	36258	33652
81	36756	34310.4
82	37355	34971.9
83	37355	35636.7
84	38103	36304.5
85	38471	36975.4
86	38571	37649.3
87	38571	38326.2
88	39098	39006
89	39994	39688.7
90	40414	40374.2
91	40966	41062.6
92	40966	41753.7
93	41575	42447.5
94	42728	43144.1
95	43120	43843.2
96	43378	44545
97	43917	45249.3
98	44440	45956.2
99	44440	46665.5
100	45778	47377.3
101	46356	48091.5
102	46751	48808.1
103	47322	49527.1
104	47943	50248.4
105	48490	50971.9

106	49097	51697.8
107	49731	52425.8
108	50183	53156.1
109	50640	53888.5
110	50640	54623
111	51643	55359.7
112	52334	56098.4
113	53156	56839.2
114	53856	57582
115	54574	58326.8
116	55255	59073.6
117	55665	59822.4
118	56432	60573.1
119	58257	61325.6
120	59468	62080.1
121	60657	62836.4
122	61535	63594.6
123	61958	64354.6
124	62380	65116.3
125	63245	65879.9
126	64221	66645.2
127	65018	67412.3
128	65801	68181.1
129	67209	68951.6
130	67870	69723.8
131	68459	70497.7
132	69570	71273.3
133	70329	72050.5
134	71365	72829.4
135	72444	73609.8
136	73382	74391.9
137	74013	75175.7
138	74620	75961
139	76217	76747.9
140	77257	77536.3
141	78148	78326.4
142	79049	79118
143	80036	79911.2
144	80694	80705.9
145	81161	81502.1
146	82279	82299.9
147	83193	83099.3
148	84370	83900.1
149	85355	84702.5
150	86232	85506.5
151	86524	86311.9
152	87041	87118.9
153	87963	87927.4
154	88866	88737.4
155	90537	89549
156	91969	90362.1
157	93572	91176.7
158	94459	91992.8
159	94701	92810.5
160	95563	93629.8
161	97110	94450.6
162	98343	95272.9
163	99409	96096.8
164	100688	96922.3
165	101542	97749.3
166	101751	98577.9

```
167             102941      99408.2
168             104475     100240
169             105508     101073
170             106481     101909
171             107089     102745
172             107769     103584
173             108289     104424
174             109030     105265
175             110549     106109
176             111219     106954
177             112141     107801
178             112906     108649
179             113648     109500
180             113767     110352
181             114309     111206
182             115457     112061
183             116360     112919
184             117175     113778
185             118045     114640
186             109784     115503
187             110092     116368
188             110757     117235
189             112166     118104
190             113206     118975
191             114732     119848
192             116451     120724
193             118594     121601
194             118911     122480
195             119553     123362
196             121525     124245
197             122257     125131
198             124129     126019
199             125620     126909
200             126419     127802
201             126711     128697
202             127643     129594
203             129892     130493
204             131146     131395
205             132475     132300
206             133981     133207
207             134747     134116
208             134965     135028
209             135749     135943
210             137047     136860
211             138584     137780
212             139534     138703
213             140351     139628
214             141034     140556
215             141339     141487
216             142056     142421
217             143531     143358
218             145045     144298
219             145848     145240
220             146828     146186
221             147033     147135
222             147315     148087
223             148171     149043
224             149083     150001
225             150360     150963
226             151659     151928
227             152422     152897
```

```
228          153289     153869
229          153423     154844
230          154115     155823
231          155625     156806
232          156451     157792
233          158270     158782
234          159614     159776
235          161635     160773
236          162178     161775
237          163192     162780
238          164908     163790
239          166302     164803
240          167147     165820
241          168192     166842
242          169194     167868
243          169562     168898
244          170110     169932
```

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

In [41]:

```
x_ant=np.arange(1,(len(ndf1)+1)-7,1, dtype='float')
y_ant=np.array(ndf1.values[:len(ndf1)-7,1], dtype='float')
table2 = []
print(tabulate(table2))
for i,c,d in zip(x_ant,y_ant,y_pred):
    if i == len(x_ant):
        table2.append([i,c,c,round(d,2)])
print(tabulate(table2, headers=["Dia","Reales","Probabilistico", "Polinomial"]))
```

```
  Dia    Reales    Probabilistico    Polinomial
-----  --------  ----------------  ------------
  237    163192            163192        162780
```

Analisis

Hemos combrobado que el modelo probabilistico no es muy adecuado para problemas de esta magnitud debido a que los valores de la vida real dan un salto considerable, en cambio el modelo matematico que hemos utilizado en este caso el cual es el polinomial tiene saltos tan considerables como el modelo probabilistico.

Conclusiones

- Se ha realizado la practica de manera correcta utilizando un modelo matematico y probabiistico y se ha podido visualizar las diferencias entre ellos.
- Algunos modelos de regresion no son recomendables de utilizar en ciertos problemas, ya que cada modelo trabaja de una manera que puede afectar al poblema que deseamos implementar dicho modelo.

Criterio personal (politico, economico y social de la situacion)

POLITICO

POLITICO

- La situacion hoy en dia esta afectando a la gente de manera considerable, por lo que las autoridades politicas deberian de aplicar leyes seberas o que se cumplan de manera diaria las impuestas previamente para que podamos convivir de una manera adecuada dia a dia sin tener ningun temor.

ECONOMICO

- La economia hoy en dia esta siendo afectada por esta situacion por lo que el pais pasa por una crisis economica global ya que las empresas han perdido una cantidad considerable de capital lo que ha llebado a cabo disminucion de empleo, y otros factores que perjudican a la sociedad.

SOCIAL

- Las personas hoy en dia tienen problemas economicos debido a la dismucion de empleo o tambien al miedo de poner en peligro su salud, lo que conlleva a que la gente no pueda dia a dia poder trabajar para poder pagar servicios que necesita a diario.

Referencias

- https://www.researchgate.net/publication/340092755_Infeccion_del_Covid-19_en_Colombia_Una_comparacion_de_modelos_logisticos_y_exponenciales_aplicados_a_la_infeccion_pc (https://www.researchgate.net/publication/340092755_Infeccion_del_Covid-19_en_Colombia_Una_comparacion_de_modelos_logisticos_y_exponenciales_aplicados_a_la_infeccion_pc
- https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/ (https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/)