

# MODELO EXPONENCIAL

AUTOR: CARLOS MOROCHO

In [1]:

```
#Importación de las librerías necesarias
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime
from scipy.optimize import curve_fit

#Obtención de los datos de COVID-19 de Ecuador
url_datos = 'DatosCOVID2020.csv'

datos = pd.read_csv(url_datos, sep = ',')
```

Out[1]:

	muestras	muestras_pcr	muestras_pcr_nuevas	pruebas_rezagadas	muerter_confirmadas	muerter_probables	muerter	muerter
0	129	129	0	106	1	0	1	
1	206	206	77	178	2	0	2	
2	273	273	67	236	2	0	2	
3	354	354	81	296	2	0	2	
4	762	762	408	651	2	0	2	
5	762	762	0	282	3	0	3	
6	1153	1153	391	481	4	0	4	
7	1670	1670	517	711	7	0	7	
8	2051	2051	381	870	7	0	7	
9	2360	2360	309	699	14	0	14	
10	2780	2780	420	708	18	0	18	
11	3618	3618	838	1311	27	0	27	
12	4290	4290	672	1692	29	0	29	
13	5090	5090	800	1965	34	0	34	
14	5915	5915	825	2347	41	0	41	
15	6615	6615	700	2680	48	0	48	
16	6992	6992	377	2869	58	0	58	
17	7451	7451	459	3232	62	0	62	
18	8251	8251	800	3423	79	0	79	
19	9019	9019	768	3428	98	76	174	
20	9604	9604	585	3302	120	78	198	
21	10317	10317	713	3661	145	101	246	
22	11309	11309	992	4475	172	146	318	
23	12386	12386	1077	5137	180	159	339	
24	13039	13039	653	5449	191	173	364	
25	14406	14406	1367	6455	220	182	402	
26	15526	15526	1120	6868	242	240	482	
27	19102	19102	3576	9463	272	284	556	
28	21568	21568	2466	7911	297	311	608	
29	22649	22649	1081	8448	315	338	653	

...	muestras	muestras_pcr	muestras_pcr_nuevas	pruebas_rezagadas	muerdes_confirmadas	muerdes_probables	muerdes	muerdes
217	507559	486987	5798	69369	8072	4285	12357	
218	512457	491885	4898	71367	8087	4288	12375	
219	515705	495133	3248	70258	8099	4288	12387	
220	516424	495852	719	70411	8106	4289	12395	
221	520243	499671	3819	72099	8115	4289	12404	
222	525803	505231	5560	71511	8160	4293	12453	
223	530086	509514	4283	71121	8195	4305	12500	
224	535429	514857	5343	67043	8221	4307	12528	
225	537556	516984	2127	61807	8235	4307	12542	
226	542463	521891	4907	56590	8248	4305	12553	
227	543529	522957	1066	56235	8266	4307	12573	
228	555016	534444	11487	63165	8280	4308	12588	
229	557421	536849	2405	59355	8297	4311	12608	
230	558365	537793	944	53342	8312	4310	12622	
231	562074	541502	3709	53605	8321	4311	12632	
232	567080	546508	5006	53176	8357	4313	12670	
233	569362	548790	2282	51707	8371	4313	12684	
234	569798	549226	436	50815	8380	4312	12692	
235	570515	549943	717	49712	8386	4312	12698	
236	577335	556763	6820	51224	8394	4310	12704	
237	579156	558584	1821	50370	8420	4310	12730	
238	583731	563159	4575	50582	8449	4312	12761	
239	587193	566621	3462	49940	8492	4323	12815	
240	593170	572598	5977	49175	8507	4323	12830	
241	596540	575968	3370	50989	8516	4323	12839	
242	597099	576527	559	49598	8525	4324	12849	
243	600741	580169	3642	49067	8592	4328	12920	
244	605331	584759	4590	50256	8614	4332	12946	
245	610265	589693	4934	50604	8642	4335	12977	
246	614531	593959	4266	50485	8658	4339	12997	

247 rows × 32 columns



In [2]:

```
#Se filtran los datos para obtener solo los casos positivos
datos_infectados = datos.loc[:, ['positivas', 'created_at']]

#Expresar los datos de fecha en número de días desde el inicio de año
formato = '%d/%m/%Y'
fecha = datos_infectados['created_at']
datos_infectados['created_at'] = fecha.map(lambda x:
    (datetime.strptime(x, formato) - datetime.strptime('01/01/2020', formato)).days
)

datos_infectados
```

Out[2]:

	positivas	created_at
0	23	72
1	28	73
2	37	74

3	positivas <sup>58</sup>	created_at <sup>75</sup>
4	111	76
5	168	77
6	260	78
7	426	79
8	532	80
9	789	81
10	981	82
11	1082	83
12	1211	84
13	1403	85
14	1627	86
15	1835	87
16	1924	88
17	1966	89
18	2302	90
19	2758	91
20	3163	92
21	3368	93
22	3465	94
23	3646	95
24	3747	96
25	3995	97
26	4450	98
27	4965	99
28	7161	100
29	7257	101
...	...	...
217	160615	289
218	161378	290
219	162245	291
220	162379	292
221	163071	293
222	164581	294
223	165407	295
224	167226	296
225	168570	297
226	170591	298
227	171134	299
228	172148	300
229	173864	301
230	175258	302
231	176103	303
232	177148	304
233	178150	305
234	178518	306
235	179066	307
236	180389	308
237	180739	309

	positivas	created_at
238	181464	310
239	182442	311
240	183863	312
241	184225	313
242	184667	314
243	185586	315
244	186469	316
245	187630	317
246	188583	318

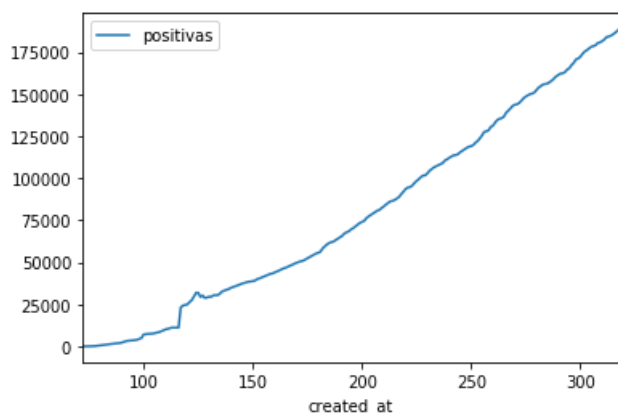
247 rows × 2 columns

In [3]:

```
datos_infectados.plot(x = 'created_at', y = 'positivas')
```

Out[3]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x2204b07d940>



Como se puede observar, el número de casos inicia el día 72 (13/03/2020) con 23 casos confirmados. Hasta el día 318 (14/11/2020) se registran 188583 casos confirmados.

In [4]:

```
#Obtención de las variables para el entrenamiento
x = list(datos_infectados.iloc[:, 1]) #Fecha (Número de día)
y = list(datos_infectados.iloc[:, 0]) #Numero de infectados

#Definición del modelo exponencial
modelo_exponencial = lambda x, a, b: a + b * np.exp(x)

#Realizamos el ajuste de curva para los datos
ajuste = curve_fit(modelo_exponencial, x, y)

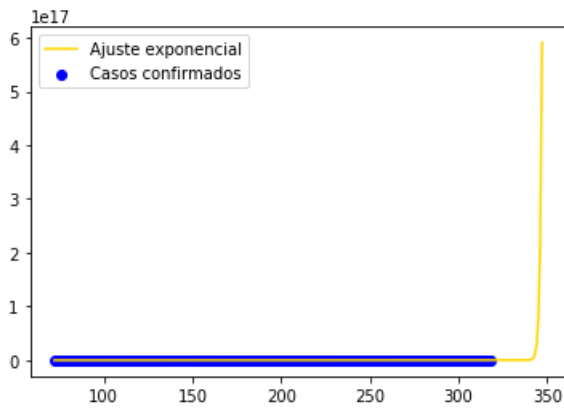
dias = 30
x_real = list(range(min(x), max(x) + dias)) #Realizamos la predicción para un mes

prediccion_total = [modelo_exponencial(i, ajuste[0][0], ajuste[0][1]) for i in x_real]

#Comprobación de la predicción para 30 días despues del 14/11/2020
prediccion = prediccion_total[-1]

#Gráfica de los datos y la predicción
print('Predicción de casos positivos para el día 347: ', int(prediccion))
plt.scatter(x, y, color = 'blue')
plt.plot(x_real, prediccion_total, color = 'gold')
plt.legend(('Ajuste exponencial', 'Casos confirmados'))
plt.show()
```

Predicción de casos positivos para el día 347: 590482843768204416



## Análisis

El modelo exponencial que se muestra en este ejemplo reacciona ante las muestras de tal manera que la cantidad de casos confirmados se vuelve demasiado grande. El crecimiento de infecciones es tanto que las muestras dan la impresión de ser aplanadas cuando se realiza la gráfica demostrativa.

## Conclusiones

Mediante el presente trabajo se determina que cuando se usa el modelo exponencial, el tamaño de los datos aumenta de manera desbordada a medida que se aumentan más el valor de entrada.

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

Debido a todo el caos que provocó el virus en todo el mundo, estos modelos de predicción que si bien no son del todo exactos pueden ayudarnos a solventar algunas respuestas que toda la sociedad está buscando, existen millones de empresas que de seguro están desarrollando estos sistemas debido a la contingencia pero creo que ninguno beneficia a la sociedad como la información como tal.

In [ ]: