

Prueba

Enunciado:

- Diseñe y desarrolle un modelo y/o script que permita simular el siguiente caso real:
- Investigar los datos de los países contiguos por COVID-19, especialmente de latinoamérica (menos Ecuador), deberán escoger uno y que no se repita, para ello se va a seleccionar el orden en el que publique dentro del foro "Tema prueba 1", con estos datos obtener los siguientes modelos:
- Generar un modelo matemático de predicción para regresión lineal, exponencial, polinómico y logarítmico, de los nuevos contactos en la próxima semana (7 días después).
- Generar un modelo probabilístico con los datos.
- Finalmente, contrarrestar los modelos matemáticos y generar las siguientes conclusiones
 - Cual tiene una mejor predicción
 - Ventajas y desventajas de los modelos.
 - Cual es el principal problema del modelo probabilístico
 - El proceso de simulación desarrollado deberá considerar los siguientes aspectos:
 - Se debe establecer un modelo basado en modelos matemáticos y probabilísticos.
 - El programa deberá generar gráficas que indiquen la ecuación matemática y probabilística de tendencias.
 - Deben calcularse las siguientes métricas:
 - Total de infectados dentro de 7 días (matemático y probabilístico).

Finalmente, desarrollar dentro de un cuaderno de Jupyter Notebook, generar un PDF y subir al repositorio.

DESARROLLO

```
In [160]: 1 import pandas as pd
          2 import numpy as np
          3 from datetime import datetime, timedelta
          4 from sklearn.metrics import mean_squared_error
          5 from scipy.optimize import fsolve
          6 from sklearn import linear_model
          7 import matplotlib.pyplot as plt
          8 from scipy.optimize import curve_fit
          9 from sklearn.linear_model import LogisticRegression
         10 %matplotlib inline
```

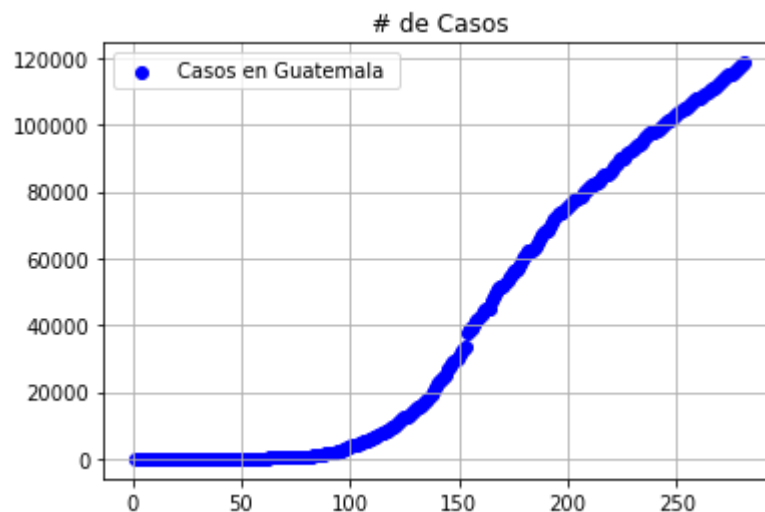
```

In [143]: 1 from datetime import datetime, timedelta
2
3 url = 'covidDatos.csv'
4 df = pd.read_csv(url).fillna(0)
5 df = df[df['location'].isin(['Guatemala'])]
6 df = df.loc[:, ['date', 'total_cases']]
7 FMT = '%Y-%m-%d'
8 date = df['date']
9 df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.st
10
11 x=np.arange(1,len(df)+1,1)
12 y=np.array(df.values[:,1], dtype='float')
13
14 print("CASOS ULTIMO DIA")
15 print(y[len(y)-1])
16 plt.scatter(x,y,label='Casos en Guatemala ', color='blue')
17 plt.grid(True)
18 plt.legend()
19 plt.title('# de Casos');

```

CASOS ULTIMO DIA

118629.0



Modelo Lineal

In [144]:

```

1 x = list(df.iloc[:, 0])
2 y = list(df.iloc[:, 1])
3
4 regr = linear_model.LinearRegression()
5 regr.fit(np.array(x).reshape(-1, 1), y)
6 y_prediccion = regr.predict([[len(x)+7]])
7
8 print("ECUACION DE LA RECTA F(X) = MX + B")
9 print('M = ' + str(regr.coef_) + ', b = ' + str(regr.intercept_))
10 print("PREDICCION +7")
11 print(int(y_prediccion))
12 plt.scatter(x, y)
13 x_real = np.array(range(min(x), max(x)))
14 plt.plot(x_real, regr.predict(x_real.reshape(-1, 1)), color='green')
15 plt.show()

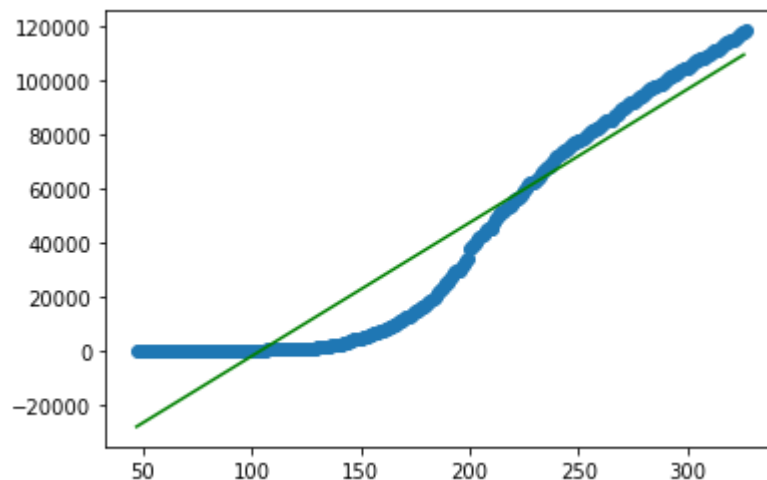
```

ECUACION DE LA RECTA $F(X) = MX + B$

M = [494.81019102], b = -51621.719244664615

PREDICCION

90883

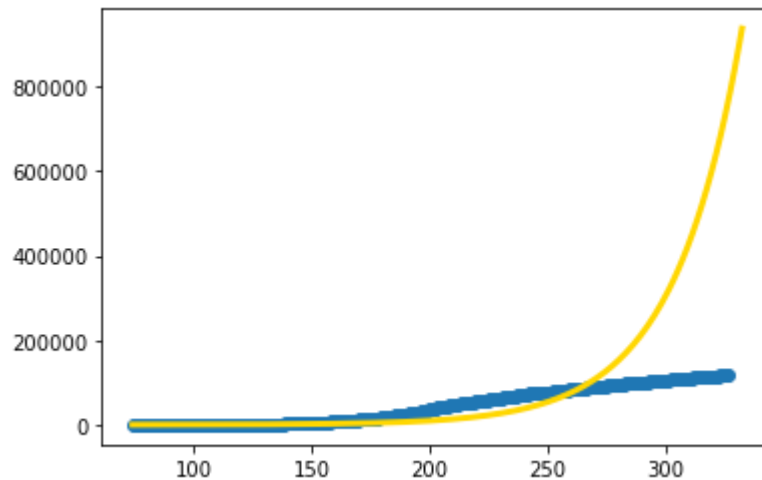


Modelo Exponencial

```
In [145]: 1 from scipy.optimize import curve_fit
2
3 x = x[27:len(x)-1]
4 y = y[27:len(y)-1]
5 curve_fit=np.polyfit(x,np.log(y),deg=1)
6 print(curve_fit)
7
8 pred_x=np.array(list(range(min(x),max(x)+7)))
9 yx=np.exp(curve_fit[1])*np.exp(curve_fit[0]*pred_x)
10 plt.plot(x,y,"o")
11 plt.plot(pred_x,yx,color='gold',linewidth=3.0)
12 print(yx[len(yx)-1])
```

```
[0.03475559 2.21179784]
```

```
937201.5917971993
```

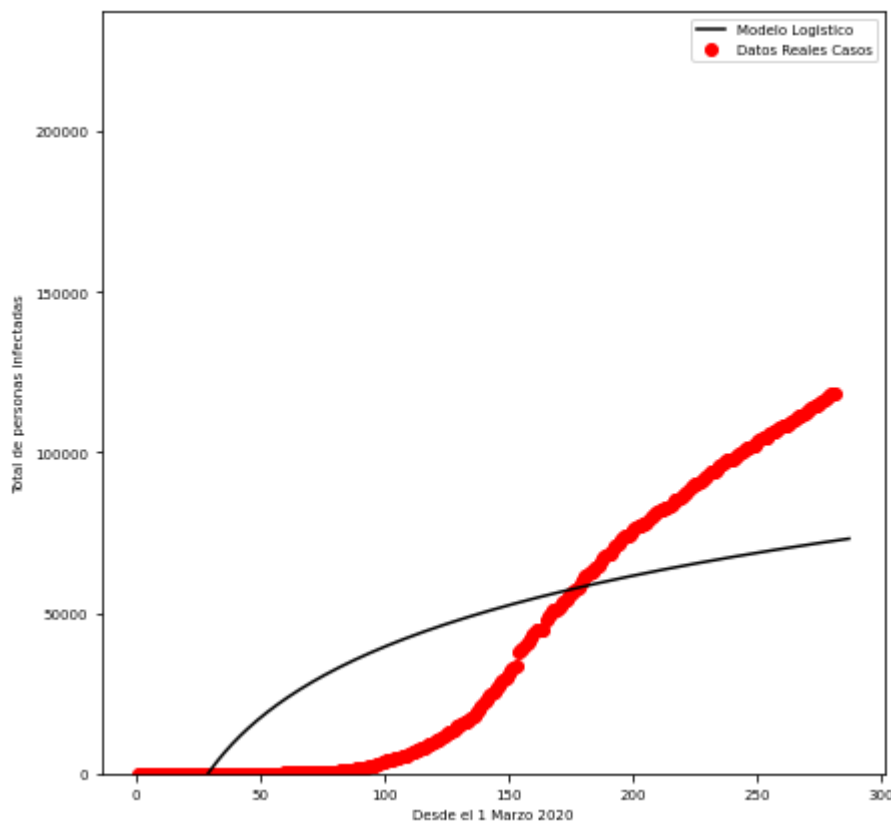


Modelo Logístico

```

In [159]: 1 from scipy.optimize import curve_fit
2 from sklearn.linear_model import LogisticRegression
3
4 def modelo_logistico(x,a,b):
5     return a+b*np.log(x)
6
7 x=np.arange(1,len(df)+1,1)
8 y=np.array(df.values[:,1])
9
10 exp_fit = curve_fit(modelo_logistico,x,y)
11
12 pred_x = list(range(min(x),max(x)+7))
13 plt.rcParams['figure.figsize'] = [7, 7]
14 plt.rc('font', size=7)
15
16 plt.scatter(x,y,label="Datos Reales Casos",color="red")
17
18 val = [modelo_logistico(i,exp_fit[0][0],exp_fit[0][1]) for i in pred_x]
19 plt.plot(pred_x, [modelo_logistico(i,exp_fit[0][0],exp_fit[0][1]) for i in
20 plt.legend()
21 plt.xlabel("Desde el 1 Marzo 2020")
22 plt.ylabel("Total de personas infectadas")
23 plt.ylim(0,max(y)*2)
24 plt.show()
25 print("PREFICCIÓN EN UNA SEMANA")
26 print(val[len(pred_x)-1])

```

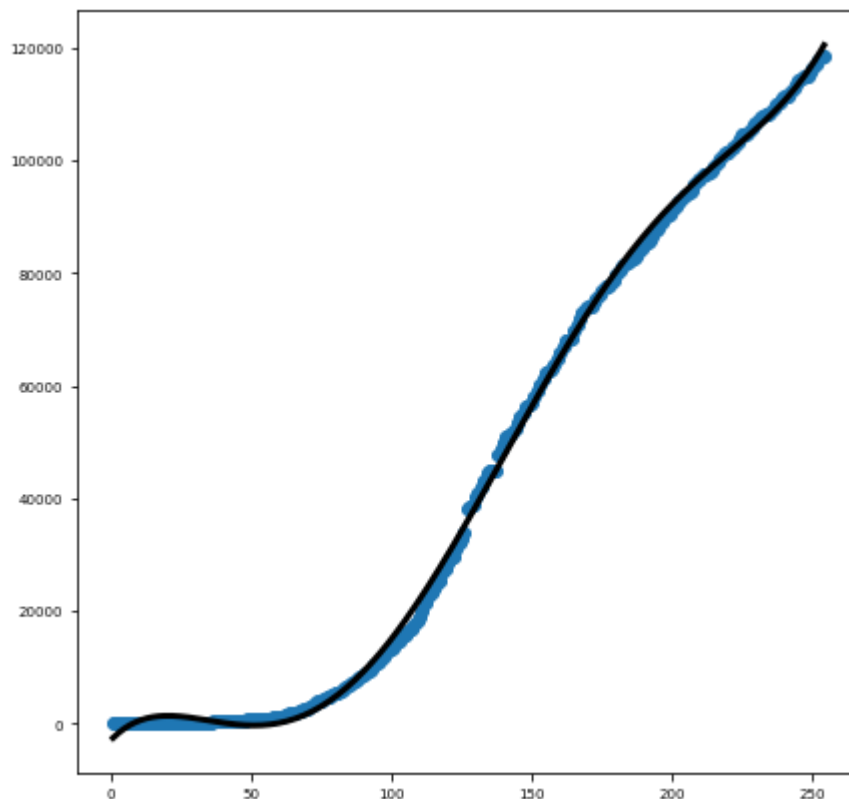


PREFICCIÓN EN UNA SEMANA
73135.14078157538

Modelo Polinomial

```
In [148]: 1 df = pd.read_csv('covidDatos.csv').fillna(0)
2 ndf= df.loc[(df['location'] == 'Guatemala') & (df['total_cases'] != 0)]
3 ndf1=ndf[['date','total_cases']]
4 x=np.arange(1,len(ndf1)+1,1, dtype='float')
5 y=np.array(ndf1.values[:,1], dtype='float')
6 fun_pol = np.poly1d(np.polyfit(x, y, 5))
7 print(fun_pol)
8
9 y_pred=fun_pol(x)
10
11 plt.scatter(x, y)
12 plt.plot(x, y_pred, c='black',lw=3)
13 plt.show()
14
15 print("PREDICCIÓN: " , round(fun_pol(len(x)+7),5))
```

5 4 3 2
1.889e-06 x - 0.001244 x + 0.2731 x - 20.22 x + 526.7 x - 3071



PREDICCIÓN: 127477.95194

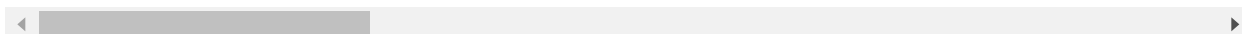
Modelo Probabilistico

```
In [149]: 1 url = 'covidDatos.csv'
          2 df = pd.read_csv(url).fillna(0)
          3 df
```

Out[149]:

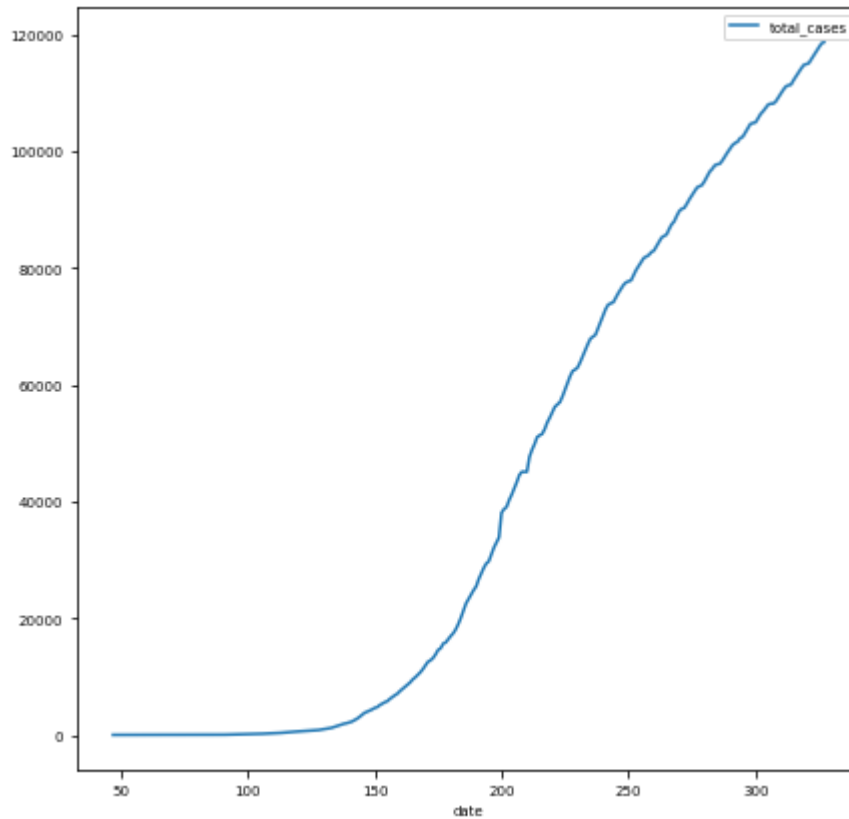
	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths
0	ABW	North America	Aruba	2020-03-13	2.0	2.0	0.000	0.000
1	ABW	North America	Aruba	2020-03-19	0.0	0.0	0.286	0.000
2	ABW	North America	Aruba	2020-03-20	4.0	2.0	0.286	0.000
3	ABW	North America	Aruba	2020-03-21	0.0	0.0	0.286	0.000
4	ABW	North America	Aruba	2020-03-22	0.0	0.0	0.286	0.000
...
58693	0	0	International	2020-11-19	696.0	0.0	0.000	0.000
58694	0	0	International	2020-11-20	696.0	0.0	0.000	0.000
58695	0	0	International	2020-11-21	696.0	0.0	0.000	0.000
58696	0	0	International	2020-11-22	696.0	0.0	0.000	0.000
58697	0	0	International	2020-11-23	696.0	0.0	0.000	0.000

58698 rows × 9 columns



```
In [150]: 1 df = df[df['location'].isin(['Guatemala'])]
2 df = df.loc[:,['date','total_cases']]
3 date = df['date']
4 df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.st
5
6 df
7 df.plot(x='date', y='total_cases')
```

Out[150]: <matplotlib.axes._subplots.AxesSubplot at 0x1f3be5fe730>



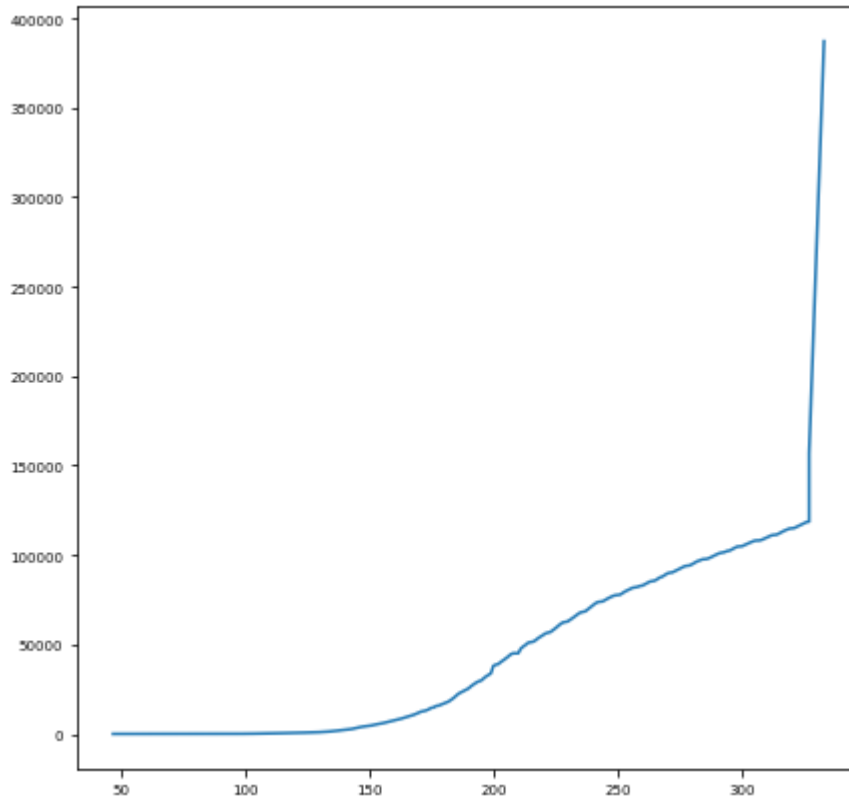

```
In [156]: 1 filtro = df["total_cases"][27:]
2 media = filtro.mean()
3 mediana = filtro.median()
4 print(mediana)
5 print(media)
6
7 url = 'covidDatos.csv'
8 df = pd.read_csv(url).fillna(0)
9 df = df[df['location'].isin(['Guatemala'])]
10 df = df.loc[:, ['date', 'total_cases']]
11 FMT = '%Y-%m-%d'
12 date = df['date']
13 df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.st
14 y = list(df.iloc[:, 1])
15 x = list(df.iloc[:, 0])
16
17
18 #Realizamos un ejemplo de prediccion
19 prediccion_siguiente = int(y[-1] + mediana)
20 print(prediccion_siguiente)
```

38359.5

45256.25196850394

156988

```
In [158]: 1 for i in range(x[-1], x[-1]+7):
2         x.append(i)
3         y.append(int(y[-1] + mediana))
4 prediccion_siguiete = int(y[-1] + mediana)
5 plt.plot(x, y)
6 plt.show()
7 print(prediccion_siguiete)
```



425501

Cual tiene una mejor prediccion

El modelo que presenta o tiene una mejor prediccion con respecto a los datos que se tiene, es el Modelo Polinomial, el cual para este caso lo utilizamos con funcion polinomica de grado 5. El modelo probabilistico como se puede ver en la grafica del desarrollo se ajusta con una aproximacion muy grande, y el valor de la prediccion hace referencia a un valor aproximado que se puede tener ya que los datos de contagios tienen una tendencia creciente lo que permite que el modelo polinomial pueja ajustarse y predecir de manera correcta.

Ventajas y desventajas de los modelos.

- Ventajas
 - La principal ventaja del modelo lineal es que presenta una ecuacion lineal que permite su facil interpretacion. El modelo lineal es implementado mayormente con datos que no tienen mucha informacion para una aporximacion asertada.
 - La ventaja del modelo exponencial es que permite aporximaciones a valores grandes y permite modelar con datos que presenta un crecimineto a medida que auta o crece su relacion.

- El modelo logístico emplea una función logarítmica que permite determinar valores que presentan un crecimiento a inicios y mantienen con un crecimiento estable. Este modelo es fácil de emplear para datos de tendencia y que después presenta una estabilidad creciente.
- El modelo polinomial permite incorporar una función lineal como inicio y ajustarla a un modelo polinómico con un grado n a partir de la ecuación lineal. Este modelo es empleado en datos que presentan un crecimiento lineal y modificarse en el tiempo
- Desventajas
 - El modelo lineal solo puede ser utilizado en datos que presenten una tendencia lineal, para los datos que presentan relaciones no se pueden emplear sin una transformación de la entrada.
 - La desventaja principal del modelo exponencial es que presenta un crecimiento que aumenta de forma exponencial es decir los datos son representados para un crecimiento positivo o imparable para los datos.
 - El modelo logístico puede ser empleado siempre y cuando los datos se ajusten a un determinado crecimiento y se mantengan estables de forma creciente.
 - El modelo polinomial puede ser empleado con datos grandes pero el grado recomendado para las ecuaciones es de 3 o 4.

Cual es el principal problema del modelo probabilístico

El principal problema del modelo probabilístico es que en base a la media y mediana de los datos que se tiene se pueden presentar un crecimiento descontrolado. Este modelo puede ser indeterminado con datos muy grandes y descontrolados los cuales no presentan una normalización para poder realizar el modelo que determine y pueda predecir de forma adecuada.