



Modelo exponencial

Materia:

Simulación

Docente:

Ing. Diego Quisi

Estudiante:

Ricardo Vinicio Jara Jara

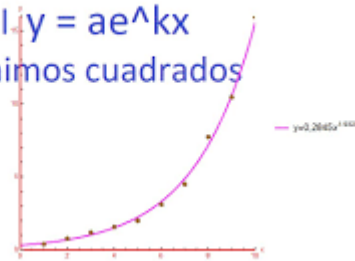
Modelo exponencial

Mientras que el modelo logístico describe un crecimiento de infección que se detendrá en el futuro, el modelo exponencial describe un crecimiento de infección imparabile . Por ejemplo, si un paciente infecta a 2 pacientes por día, después de 1 día tendremos 2 infecciones, 4 después de 2 días, 8 después de 3 y así sucesivamente.

$$f(x, a, b, c) = \frac{c}{1 + e^{-(x-b)/a}}$$

A continuacion se tiene un ejemplo de regresion exponencial

Curva de ajuste para una función tipo
exponencial $y = ae^{kx}$
usando mínimos cuadrados



Implementación

```
In [17]: # 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

from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures

#Mis Datos COVID EC

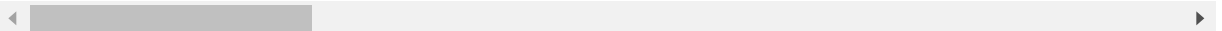
url = 'https://raw.githubusercontent.com/andrab/ecuacovid/master/datos_crudos/
ecuacovid.csv'

df = pd.read_csv(url)
df
```

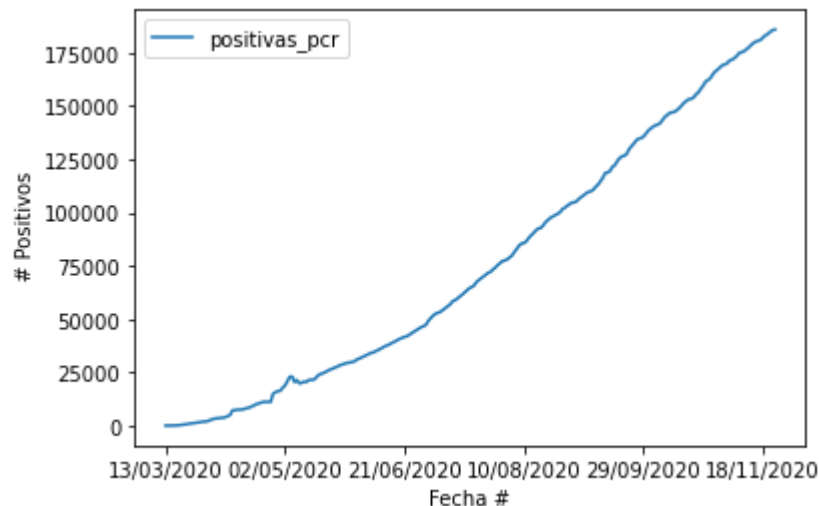
Out[17]:

	muestras	muestras_pcr	muestras_pcr_nuevas	pruebas_rezagadas	muertes_confirmadas	n
0	129	129	0	106	1	
1	206	206	77	178	2	
2	273	273	67	236	2	
3	354	354	81	296	2	
4	762	762	408	651	2	
...
251	629093	608521	5063	48617	8729	
252	632453	611881	3360	48275	8750	
253	636428	615856	3975	47265	8787	
254	641542	620970	5114	45942	8804	
255	643405	622833	1863	45756	8825	

256 rows × 32 columns



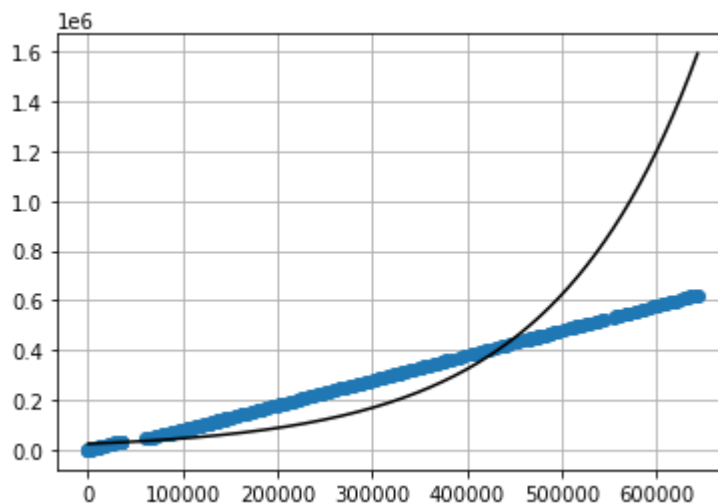
```
In [18]: ax = df.plot(x='created_at', y='positivas_pcr')
ax.set_xlabel("Fecha #")
ax.set_ylabel("# Positivos")
FMT = '%d/%m/%Y'
date = df['created_at']
df['created_at'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.s
trptime("01/01/2020", FMT)).days)
x = list(df.iloc[:, 0]) # Fecha
y = list(df.iloc[:, 1]) # Numero de casos
```



```
In [19]: curve_fit = np.polyfit((x),np.log(y), deg=1)
print(curve_fit)
```

```
[6.54535133e-06 1.00686897e+01]
```

```
In [20]: pred_x = np.array(list(range(min(x),max(x)+5)))
yx = np.exp(curve_fit[1]) * np.exp(curve_fit[0]*pred_x)
plt.plot(x,y, 'o')
plt.plot(pred_x,yx,color="black")
plt.grid(True)
```



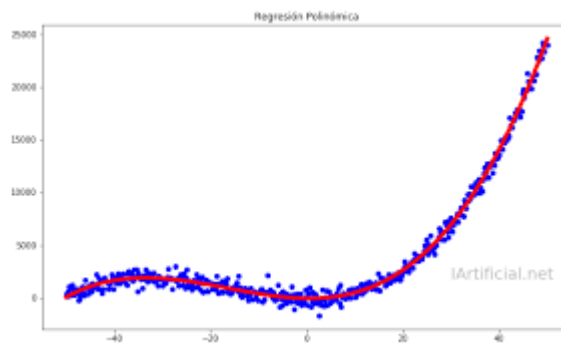
Modelo polinomial

Predicción de una variable de respuesta cuantitativa a partir de una variable predictora cuantitativa, donde la relación se modela como una función polinomial de orden n (esto significa que pueden tener de diferentes exponenciales o grados y se debe ir probando)

Se puede tener una ecuación con diferentes grados

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_nx^n + \varepsilon$$

Ejemplo de una regresión polinómica de grado 4.



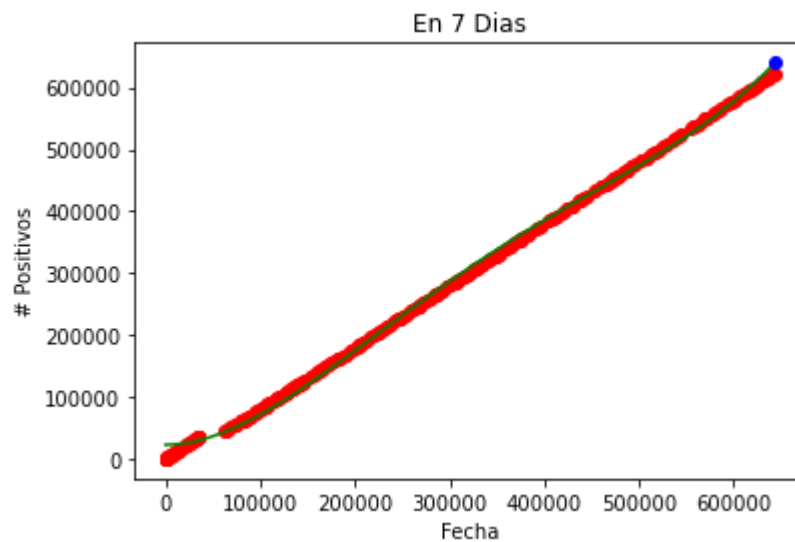
```

In [65]: pf = PolynomialFeatures(degree = 4)
X = pf.fit_transform(np.array(x).reshape(-1, 1))
regresion_lineal = LinearRegression()
regresion_lineal.fit(X, y)
pred_x = list(range(0,max(x)+50))

puntos = pf.fit_transform(np.array(pred_x).reshape(-1, 1))
prediccion_entrenamiento = regresion_lineal.predict(puntos)
respuesta = round((prediccion_entrenamiento[-1]+7))

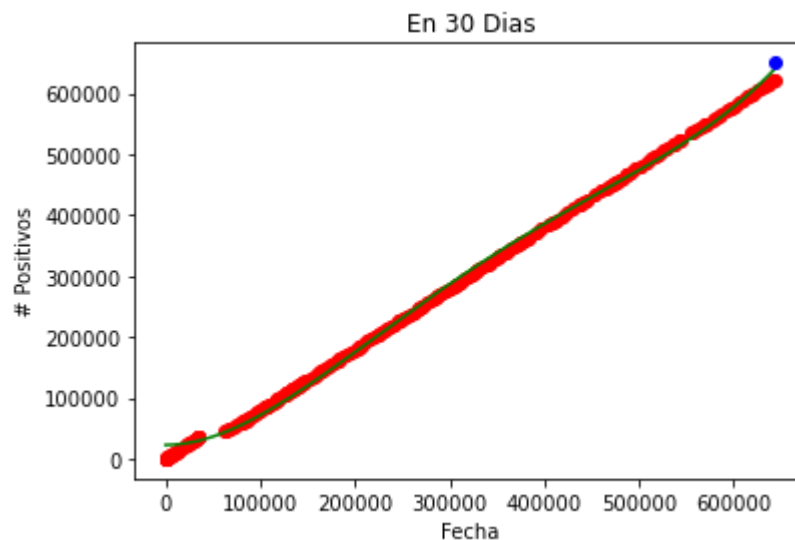
plt.title("En 7 Dias")
plt.plot(pred_x, prediccion_entrenamiento, color='green')
plt.scatter(x,y,label="Datos Reales",color="red")
plt.xlabel("Fecha")
plt.ylabel("# Positivos")
plt.plot(x[-1]+7, respuesta, 'o',color="blue")
plt.show()

```



```
In [68]: puntos = pf.fit_transform(np.array(pred_x).reshape(-1, 1))
prediccion_entrenamiento = regresion_lineal.predict(puntos)
respuesta = round((prediccion_entrenamiento[-1]+30))

plt.title("En 30 Dias")
plt.plot(pred_x, prediccion_entrenamiento, color='green')
plt.scatter(x,y,label="Datos Reales",color="red")
plt.xlabel("Fecha")
plt.ylabel("# Positivos")
plt.plot(x[-1]+30, respuesta, 'o',color="blue")
plt.show()
```



Conclusiones: Con estos modelos matemáticos tanto el **Modelo lineal**, **Modelo logístico**, **Modelo exponencial** y el **Modelo polinomial** y con una base de datos completa y sobre todo que tenga datos reales podríamos trabajar en simulaciones para tratar de predecir que pasará en el futuro es decir tratar de ver cómo se mostrarán los datos en un futuro cercano o lejano de acuerdo a la cantidad de datos que tengamos en nuestro bd