

**Nombre: Jorge Arévalo**

**Docente: Ing. Diego Quisi**

**Materia: Simulación**

## **Covid-19 infección en Ecuador. Modelos probabilísticos**

Implementación de un modelo probabilístico de infección por el virus Covid-19

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

Se crean 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 [37]:

```
1 # Importar las librerías para el análisis
2 import pandas as pd
3 import numpy as np
4 from datetime import datetime, timedelta
5 from sklearn.metrics import mean_squared_error
6 from scipy.optimize import curve_fit
7 from scipy.optimize import fsolve
8 from sklearn import linear_model
9 import matplotlib.pyplot as plt
10 %matplotlib inline
11
```

In [38]:

```

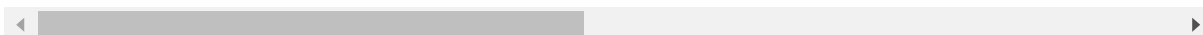
1 # Actualizar los datos (URL)
2 url = 'https://covid.ourworldindata.org/data/ecdc/new_cases.csv'
3 df = pd.read_csv(url)
4 df

```

Out[38]:

	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	1
1	2020-01-01	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	1
2	2020-01-02	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	1
3	2020-01-03	17	0.0	NaN	0.0	NaN	NaN	NaN	NaN	1
4	2020-01-04	0	0.0	NaN	0.0	NaN	NaN	NaN	NaN	1
...	...	...	...	...	...	...	...	...	...	...
319	2020-11-14	634423	66.0	490.0	867.0	109.0	104.0	0.0	2.0	1185
320	2020-11-15	597736	205.0	532.0	844.0	0.0	171.0	0.0	1.0	846
321	2020-11-16	504453	163.0	597.0	860.0	147.0	146.0	0.0	0.0	564
322	2020-11-17	554913	65.0	602.0	910.0	42.0	77.0	0.0	0.0	789
323	2020-11-18	560504	383.0	694.0	1002.0	37.0	164.0	0.0	0.0	1063

324 rows × 216 columns



Imprimos los resultados y agregamos el numero del dia

In [39]:

```

1 df = df.loc[:,['date','Ecuador']] #Selecciono Las columnas de analisis
2 # Expresar Las fechas en numero de dias desde el 01 Enero
3 FMT = '%Y-%m-%d'
4 date = df['date']
5 df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-0
6 df

```

Out[39]:

	date	Ecuador
0	-1	0.0
1	0	0.0
2	1	0.0
3	2	0.0
4	3	0.0
...	...	...
319	318	1161.0
320	319	953.0
321	320	668.0
322	321	381.0
323	322	428.0

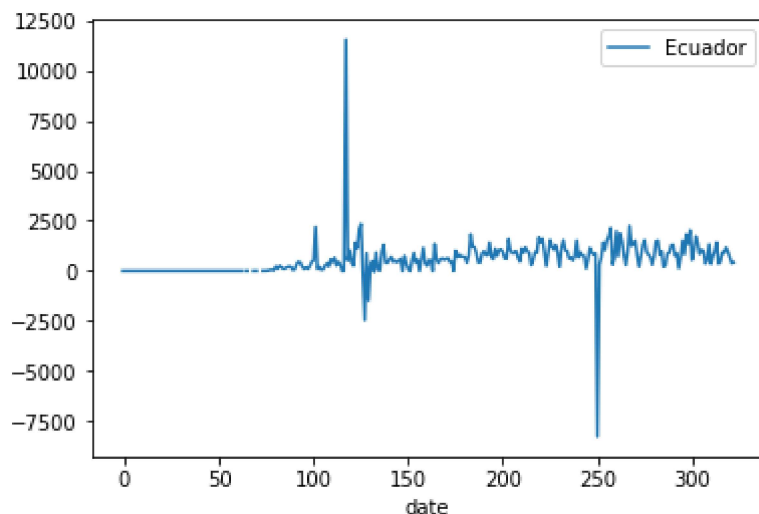
324 rows × 2 columns

In [40]:

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

Out[40]:

&lt;matplotlib.axes.\_subplots.AxesSubplot at 0x25f5fe13188&gt;



```

1 Ahora podemos analizar un modelo probabilistico para el examen.
2

```

```
3 ## El modelo basado en probabilidad
4
5 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 tasa de crecimiento de los nuevos casos.
6
7
```

In [41]:

```
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
```

663.0

701.953488372093

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

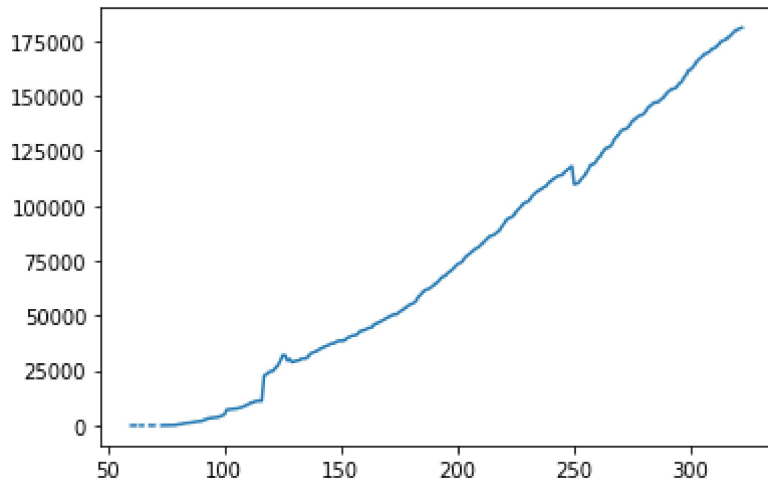
In [42]:

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

181767

In [43]:

```
1 # Quiero predecir cuántos "Casos" voy a obtener de aqui a 10 dias.
2 for i in range(x[-1], x[-1]-7):
3     x.append(i)
4     y.append(int(y[-1] + mediana))
5 probabilistico=y[319]
6 plt.plot(x[61:], y[61:])
7 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 cargar 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.

## Implementar

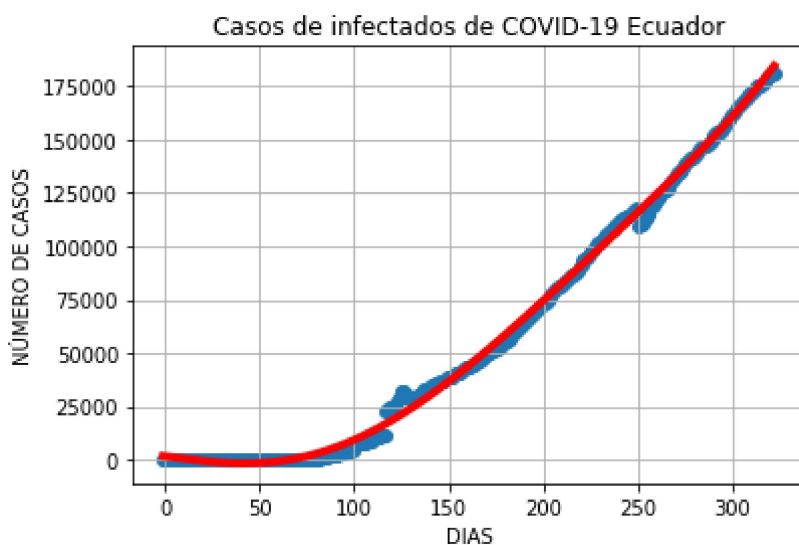
1. Comparar el modelo de predicción matemático vs probabilidad.

In [44]:

```

1  from sklearn.preprocessing import PolynomialFeatures
2
3  x = list(df_t.iloc[:, 0])
4  y = list(df_t.iloc[:, 1])
5
6  y= np.nan_to_num(y)
7  x=np.array(x).reshape(-1,1)
8  y=np.array(y).reshape(-1,1)
9  poly = PolynomialFeatures(degree=5, include_bias=False)
10 x_poly = poly.fit_transform(x)
11
12 model = linear_model.LinearRegression()
13
14 model.fit(x_poly, y)
15 y_pred = model.predict(x_poly)
16 y_pred = y_pred-7
17 plt.title('Casos de infectados de COVID-19 Ecuador')
18 plt.scatter(x, y)
19 plt.plot(x, y_pred, color='r',lw=4)
20 plt.grid(True)
21 plt.xlabel('DIAS')
22 plt.ylabel('NÚMERO DE CASOS')
23 plt.show()

```



Al comparar el modelo matemático con el probabilístico, el modelo que mejor se ajusta es el Polinomial ya que es muy similar a la Regresión lineal pero con una ligera desviación en la forma en que tratamos nuestro espacio.

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

In [45]:

```

1 #Implementar
2 #--- El modelo basado en probabilidad ---
3 probabilistico=probabilistico-7
4 print('Casos hace 7 días: ',np.round(valorProbabilistico,1))

```

Número de casos hace 7 días: 178667.0

In [46]:

```

1 # Modelo Logistico
2 logistico=y_pred[319-7]
3 print('Casos hace 7 días:',np.round(logistico,0))

```

Casos hace 7 días: [172142.]

**> Puntos extras: Investigas sobre la correlacion de variables y aplicar el calculo en base a los datos del Ecuador.**

In [52]:

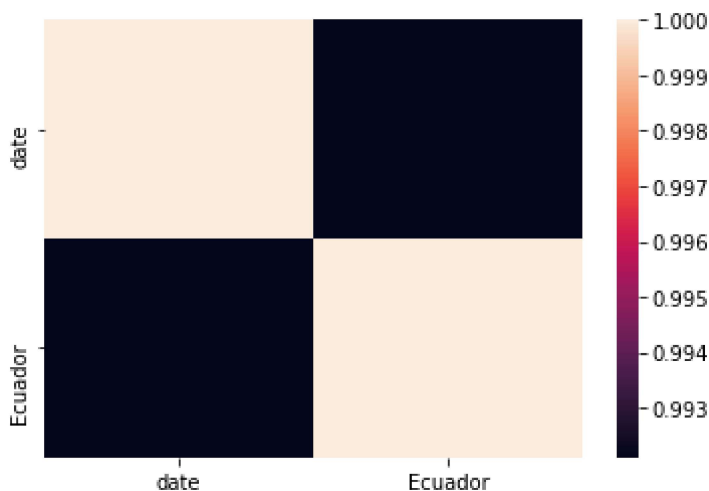
```

1 import seaborn as sns
2 corr = df_t.corr()
3 sns.heatmap(corr,
4             xticklabels=corr.columns.values,
5             yticklabels=corr.columns.values)

```

Out[52]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x25f61363088>



## Analisis

La regresión polinomial es una forma de regresión lineal en la que la relación entre la variable independiente  $x$  y la variable dependiente  $y$  es modelada como un polinomio de grado  $n$  en  $x$ .

## Conclusiones

Se realizó la comparación en donde se puede ver que la regresión polinomial es la mejor opción para demostrar los casos del COVID – 19

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

Se debe tener presente las medidas de prevención contra el virus ya que así podremos evitar que se aumente los contagios en el país.

Este virus ha afectado a varias empresas ya que por evitar el contagio deben cerrarse la cual, tambien en algunas empresas despedían a empleados por lo que estaba en stop el negocio y no generaban ingresos. El país presente grandes desequilibrios en dinero ya que por las restricciones se tiene una diferente manera de trabajar

Mediante este análisis podemos ver en qué estado se encuentra el país por lo que debemos seguir con las medidas de prevención contra el COVID-19 para así evitar contagios ya que en las gráficas se puede observar altos índices de casos nuevos y de numerosas muertes en el Ecuador.

## Referencias

- [https://www.researchgate.net/publication/340092755\\_Infeccion\\_del\\_Covid-19\\_en\\_Colombia\\_Una\\_comparacion\\_de\\_modelos\\_logisticos\\_y\\_exponenciales\\_aplicados\\_a\\_la\\_infeccion\\_por\\_SARS-CoV-2](https://www.researchgate.net/publication/340092755_Infeccion_del_Covid-19_en_Colombia_Una_comparacion_de_modelos_logisticos_y_exponenciales_aplicados_a_la_infeccion_por_SARS-CoV-2)
- <https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/>