

Wilson Rodas

Modelo Probabilístico

```
In [1]: #Importación de las librerías necesarias
import pandas as pd
import numpy as np
from datetime import datetime
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [2]: #Obtención de los datos de COVID-19 de Ecuador
url_datos = 'DatosCOVID2020.csv'

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

```
Out[2]:
```

	muestras	muestras_pcr	muestras_pcr_nuevas	pruebas_rezagadas	muerteres_confirmadas	muerteres_probables	muerteres	muerteres_nuevas	positivas
0	129	129	0	106	1	0	1	0	23
1	206	206	77	178	2	0	2	1	28
2	273	273	67	236	2	0	2	0	37
3	354	354	81	296	2	0	2	0	58
4	762	762	408	651	2	0	2	0	111
...
242	597099	576527	559	49598	8525	4324	12849	10	184667
243	600741	580169	3642	49067	8592	4328	12920	71	185586
244	605331	584759	4590	50256	8614	4332	12946	26	186469
245	610265	589693	4934	50604	8642	4335	12977	31	187630
246	614531	593959	4266	50485	8658	4339	12997	20	188583

247 rows × 32 columns

```

In [3]: #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[3]:

```

	positivas	created_at
0	23	72
1	28	73
2	37	74
3	58	75
4	111	76
...
242	184667	314
243	185586	315
244	186469	316
245	187630	317
246	188583	318

247 rows × 2 columns

```

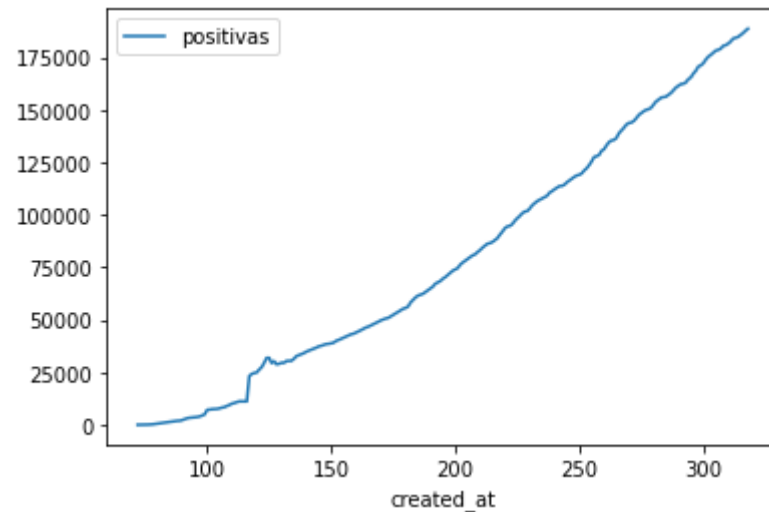
In [4]: datos_infectados.plot(x = 'created_at', y = 'positivas')

```

```

Out[4]: <AxesSubplot:xlabel='created_at'>

```



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 [17]: #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

mediana = datos_infectados['positivas'].median()

#Creación del modelo
modelo_probabilistico = lambda y: int(y[-1] + mediana)

#Comprobación de la predicción para 30 días despues del 14/11/2020
dia = x[-1] + 30 #Se obtiene el dia a predecir sumando la ultima fecha registrada mas un mes

prediccion = y

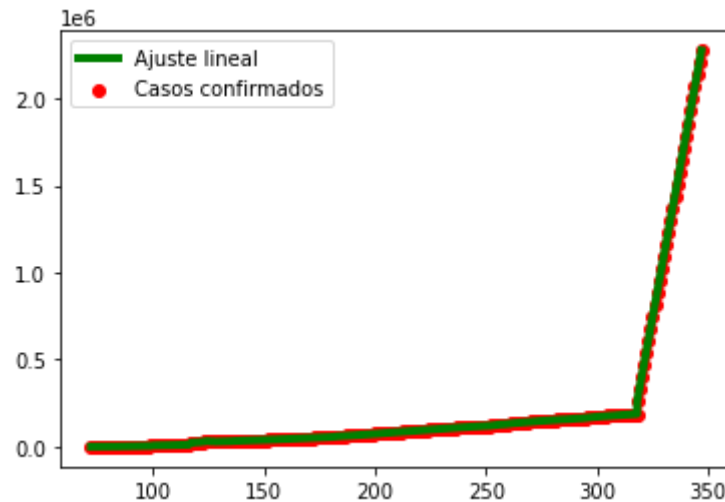
for i in range(x[-1], dia):
    x.append(i)
    prediccion.append(modelo_probabilistico(prediccion))

prediccion_semana = modelo_probabilistico(y)
print('Predicción de casos positivos para el 13/12/2020: ', int(prediccion_semana))
```

Predicción de casos positivos para el 13/12/2020: 2345253

```
In [13]: #Gráfica de los datos y la predicción
```

```
plt.scatter(x, y, color = 'red')
plt.plot(x, prediccion, color = 'green', linewidth = 4)
plt.legend(('Ajuste lineal', 'Casos confirmados'))
plt.show()
```



De acuerdo al modelo entrenado, el número de casos que habrá el 13/12/2020 será 2345253 confirmados.

1) Comparación con el modelo matemático

```
In [80]: #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 logístico
definicion = PolynomialFeatures(degree = 4)
modelo_polinomial = LinearRegression()

#Ajuste de la entrada a la forma polinomial
X = definicion.fit_transform(np.array(x).reshape(-1, 1))

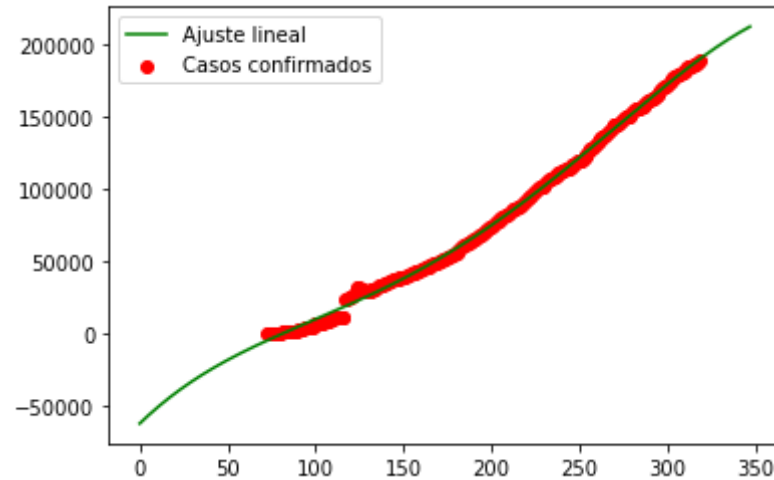
#Realizamos el ajuste de curva para los datos
modelo_polinomial.fit(X, y)

dias = 30
prediccion = modelo_polinomial.predict([X[-1] + dias])
print('Predicción de casos positivos para el 13/12/2020: ', int(prediccion))
```

Predicción de casos positivos para el 13/12/2020: 224133

```
In [85]: #Gráfica de los datos y predicción
x_real = np.array(range(0, 348))
x_real = definicion.fit_transform(x_real.reshape(-1, 1))

plt.scatter(x, y, color = 'red')
plt.plot(modelo_polinomial.predict(x_real), color = 'green')
plt.legend(('Ajuste lineal', 'Casos confirmados'))
plt.show()
```



Como se puede observar, el modelo probabilístico da predicciones con saltos demasiados grandes, tal que el resultado de usarlo fue que el 13/12/2020 se pronostican 258153 casos confirmados, mientras que el modelo matemático (Polinomial de 4to grado en este caso) predice 224133 casos confirmados, lo cual se apega más a la realidad.

2) Retroceder 7 días y observar si el modelo se ajusta

```
In [98]: #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

mediana = datos_infectados['positivas'].median()

#Creación del modelo
modelo_probabilistico = lambda y: int(y[-1] + mediana)
```

```
#Comprobación de la predicción para 30 días después del 14/11/2020
dia = x[-1] - 7 #Se obtiene el día a predecir sumando la última fecha registrada más un mes

y_real = y

for i in range(dia, x[-1]):
    x.append(i)
    y_real.append(modelo_probabilistico(y_real))

prediccion = modelo_probabilistico(y)
print('Predicción de casos positivos para el 07/11/2020: ', int(prediccion))
print('Casos reales confirmados el 07/11/2020: ', (y_real[-7]))
```

Predicción de casos positivos para el 07/11/2020: 745143
Casos reales confirmados el 07/11/2020: 258153

De acuerdo a los datos obtenidos, el modelo probabilístico no funciona para este conjunto de datos. Los saltos de predicción se encuentran demasiado alejados de los reales.

3) Análisis

Los datos fueron obtenidos del siguiente enlace: <https://github.com/andrab/ecuacovid>, el cual es un proyecto que reúne los datos de los informes nacionales. Con respecto al modelo lineal entrenado en este trabajo, se ha utilizado un total de 246 registros, de los cuales los datos principales a tratar fueron: la fecha ('created_at'), los casos confirmados ('positivas').

El modelo probabilístico toma como referencia el valor de la mediana para realizar las predicciones, sin embargo, en el conjunto de datos presentados el error es muy grande como para usarlo como método definitivo.

4) Conclusiones

Mediante este trabajo se determina que el modelo probabilístico genera saltos demasiado grandes y que en comparación, los modelos matemáticos funcionan mucho mejor para este conjunto de datos.

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

La emergencia sanitaria por COVID-19 en el Ecuador y en el resto del mundo es una situación realmente dura, siendo así que la gran mayoría de sistemas de salud se han visto saturados con el incremento y llegada de nuevos pacientes. La prioridad de las naciones debería ser apoyar a la investigación de la cura contra el virus, o caso contrario, apoyar a la compra de insumos médicos para respaldar a las casas de salud.