```
Universidad Politecnica Salesiana
           Nombre: Erika Morocho
           Asignatura: Simulación
           Covid-19 infección en Ecuador. Modelos matemáticos y predicciones
           Implementacion de un modelo probabilistico de infección por el virus Covid-19
           Se realiza un análisis probabilistico simple del crecimiento de la infección en Python y el modelos para comprender mejor la
           evolución de la infección.
           Se crea 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
  In [1]: # Importar las librerias para el analasis
           import pandas as pd
           import numpy as np
           from scipy.integrate import solve_ivp
           from scipy.optimize import minimize
           from scipy.integrate import odeint
           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
  In [2]: # Actualizar los datos (URL)
           url = 'https://covid.ourworldindata.org/data/ecdc/new_cases.csv'
           df = pd.read_csv(url)
           df = df.loc[:,['date','Ecuador']] #Selecciono las columnas de analasis
           # Expresar las fechas en numero de dias desde el 01 Enero
           df = df[df['date'] >= '2020-03-01']
           FMT = '\%Y - \%m - \%d'
           date = df['date']
           df['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-01-01"
           , FMT)).days)
  Out[2]:
                date Ecuador
             61 60
                         1.0
             62
                 61
                         5.0
             63
                 62
                         1.0
             64
                 63
                        NaN
             65
                 64
                         3.0
                         3.0
             67
                 66
                        NaN
                 67
             68
                        NaN
             69
                 68
                         1.0
             70
                 69
                         1.0
             71 70
                         2.0
             72 71
                        NaN
             73 72
                        NaN
             74 73
             75
                74
                         5.0
             76
                 75
                         9.0
             77
                 76
                        21.0
             78
                 77
                        53.0
             79
                78
                        57.0
             80
                 79
                        31.0
             81
                 80
                       227.0
             82
                 81
                       106.0
             83
                       257.0
                 82
             84
                 83
                       192.0
             85
                 84
                       101.0
             86
                 85
                       129.0
             87
                 86
                       192.0
             88
                 87
                       224.0
             89
                 88
                       208.0
             90
                 89
                        55.0
             •••
            305 304
                       845.0
                      1045.0
            306
                305
            307 306
                       1002.0
            308 307
                       368.0
            309
                308
                       548.0
                       1323.0
            310 309
            311 310
                       350.0
                       725.0
            312 311
            313 312
                       978.0
                      1421.0
            314 313
            315 314
                       362.0
            316 315
                       442.0
            317 316
                       919.0
            318 317
                       883.0
            319 318
                       1161.0
                       953.0
            320 319
            321 320
                       668.0
            322 321
                       381.0
            323 322
                       428.0
            324 323
                      1146.0
            325
                324
                       996.0
            326 325
                       594.0
            327 326
                       1036.0
            328
                327
                       767.0
            329 328
                       301.0
            330 329
                       492.0
            331 330
                       794.0
            332 331
                       908.0
            333 332
                      1396.0
            334 333 1375.0
           274 rows × 2 columns
           Imprimos los resultados y agregamos el numero del dia
In [220]: df.plot(x ='date', y='Ecuador')
Out[220]: <matplotlib.axes._subplots.AxesSubplot at 0x1fc7ae44888>
                     Ecuador
            2000
            1500
            1000
             500
                60
                        70
                                                 100
                                                          110
                                     date
           Ahora podemos analizar un modelo probabilisto para el examen.
           El modelo basado en probabilidad
           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 taza
  In [4]: filtro = df["Ecuador"][0:] # Filtro los datos que se empezo a tener casos
           #Obtenemos la mediana
           media = filtro.mean()
           mediana = filtro.median()
           print(mediana)
           print(media)
           670.0
           709.6988847583643
           De la ecuación de la recta y = mX + b nuestra pendiente «m» es el coeficiente y el término independiente «b»
In [245]: #Vamos a comprobar:
           # según la media y la mediana podemos obtener la taza de crecieminto y predicir su comportam
           # Cargamos los datos de total de casos
           url = 'https://covid.ourworldindata.org/data/ecdc/total_cases.csv'
           df_t = pd.read_csv(url)
           FMT = '\%Y - \%m - \%d'
           date = df_t['date']
           df_t['date'] = date.map(lambda x : (datetime.strptime(x, FMT) - datetime.strptime("2020-01-0"))
           1", FMT)).days)
           df_t = df_t.loc[:,['date','Ecuador']] #Selecciono las columnas de analasis
           df_t = df_t[df_t['date'] > 59]
           df_t['Ecuador'] = df['Ecuador'].fillna(0)
           x = list(df_t.iloc [:, 0]) # Dias
           y = list(df_t.iloc [:, 1]) # Total casos
           #Realizamos un ejemplo de prediccion
           prediccion\_siguiente = int(y[-1] + mediana)
           766
In [246]: # Quiero predecir cuántos "Casos" voy a obtener de aqui a 10 dias.
           print(aux)
           for i in range(x[-1]+1, x[-1]+11):
               x.append(i)
               y.append(int(y[-1] + mediana))
           fig = plt.figure(facecolor='w')
           fig = plt.figure(figsize=(15,5))
           ax = fig.add_subplot(111, axisbelow=True)
           ax.plot(x[aux:], y[aux:],'b', alpha=1, lw = 2, label = 'Modelo probabilístico')
           ax.grid()
           ax.legend()
           ax.set_title("Modelo probabilístico")
           52
Out[246]: Text(0.5, 1.0, 'Modelo probabilístico')
           <Figure size 432x288 with 0 Axes>
                                                         Modelo probabilístico
                    Modelo probabilístico
            1600
            1400
            1200
            1000
                                                                              118
           Practica
             1. Comparar el modelo de predicion matematico vs probabilidad.
             2. Retroceder un semana y comparar el modelo matematico vs probabilidad vs reales. Solo cargan los datos para generar
               los modelos menos 7 dias.
           Puntos extras: Investigas sobre la correlacion de variables y aplicar el calculo en base a los datos del Ecuador.
In [247]: # Implementar
           # 1. Comparar el modelo de predicción matemático vs probabilístico:
           # Modelo probabilístico
           fig = plt.figure(facecolor='w')
           fig = plt.figure(figsize=(15,5))
           ax = fig.add_subplot(111, axisbelow=True)
           ax.plot(x[:len(x)-9], y[:len(x)-9], 'b', alpha=1, lw = 2, label = 'Datos reales')
           ax.plot(x[len(x)-10:], y[len(x)-10:], 'r', alpha=1, lw = 2, label = 'Modelo probabilístico (p
           redicción)')
           ax.set_xlabel('Tiempo en dias')
           ax.set_ylabel('Numero de Personas')
           ax.set_title("Modelo probabilístico")
           ax.legend()
           ax.grid()
           x_{matematico} = x[:]
           y_matematico = y[:]
           # Modelo polinomial
           grado = 6
           modeloPol = pd.DataFrame(columns=('Días', 'Modelo Polinomial'))
           y_polinomial = np.poly1d(np.polyfit(x_matematico,y_matematico,grado))
           # Asignación de valores a la variable dependiente según la ecuación anterior
           for pred in x_matematico:
                modeloPol.loc[len(modeloPol)] = [pred, y_polinomial(pred)]
           # Gráfico del modelo polinomial de grado n
           print("Pacientes fallecidos: ",datos_dia.loc[0, 'Pacientes']-datos_dia.loc[0, 'Hospitalizado
           s'])
print("Ecuación polinomial de grado: ",grado," ")
           Ecuación polinomial de grado: 6
           -1.467e-06 \times + 0.0008151 \times - 0.1855 \times + 22.16 \times - 1465 \times + 5.085e+04 \times - 7.242e+05
           <Figure size 432x288 with 0 Axes>
                                                          Modelo probabilístico
                                                                                         Datos reales
                                                                                            Modelo probabilístico (predicción)
              2000
            Personas
1500
            Numero de
              500
                     60
                                                                             100
                                                                                           110
                                                                                                          120
                                                             Tiempo en dias
           <Figure size 432x288 with 0 Axes>
                                                           Modelo polinomial
                                                                                           Datos reales
                                                                                              Modelo polinomial
              2000
                                                                                              Modelo polinomial (Predicción)
            Dersonas
1500
            Numero de
1000
              500
                                                                                           110
                                                                                                          120
                                                               90
                                                                             100
                                                             Tiempo en dias
 In [13]: # METODO PROBABILISTICO
           media = ndf1.values[:,1].mean()
           mediana = np.median(ndf1.values[:,1])
           print("MEDIA: ", media)
           print("MEDIANA: ", mediana)
            175000
            150000
            125000
            100000
             75000
             50000 -
             25000
                                  100
                                         150
                                                         250
                           50
                                                 200
           MEDIA: 77559.49049429658
           MEDIANA: 69570.0
In [249]: # 2. Retroceder un semana y comparar el modelo matematico vs probabilidad vs reales.
           # Solo cargan los datos para generar los modelos menos 7 dias.
           # Datos de hace una semana (Abril 14 a Abril 21)
           x_{\text{Reales\_Historicos}} = x[len(x)-17:len(x)-10]
           y_{Reales_{Historicos}} = y[len(y)-17:len(y)-10]
           fig = plt.figure(facecolor='w')
           fig = plt.figure(figsize=(15,5))
           ax = fig.add_subplot(111, axisbelow=True)
           ax.plot(x_Reales_Historicos, y_Reales_Historicos, 'r', lw=2, label='Datos Reales')
           ax.set_xlabel('Tiempo en dias')
           ax.set_ylabel('Numero de Personas')
           ax.legend()
           ax.grid()
           # Gráfica del modelo matemático (polinomial)
           fig = plt.figure(facecolor='w')
           fig = plt.figure(figsize=(15,5))
           ax = fig.add_subplot(111, axisbelow=True)
           ax.plot(modeloPol.iloc[len(modeloPol)-17:len(modeloPol)-10,0],
                    modeloPol.iloc[len(modeloPol)-17:len(modeloPol)-10,1], 'g', lw=2, label='Datos del mod
           elo matemático (polinomial)')
           ax.set_xlabel('Tiempo en dias')
           ax.set_ylabel('Numero de Personas')
           ax.legend()
           ax.grid()
           # Gráfica del modelo probabilístico
           # Obtención de datos desde hace una semana a la fecha mediante la adición de la media
           aux = y[len(x)-17]
           for i in range(len(x)-17, len(x)-10):
               y[i] = (int(aux + mediana))
               aux = y[i]
           fig = plt.figure(facecolor='w')
           fig = plt.figure(figsize=(15,5))
           ax = fig.add_subplot(111, axisbelow=True)
           ax.plot(x[len(x)-17:len(x)-10],y[len(y)-17:len(y)-10],'b',lw=2, label='Modelo Probabilístic
           ax.set_xlabel('Tiempo en dias')
           ax.set_ylabel('Numero de Personas')
           ax.legend()
           <Figure size 432x288 with 0 Axes>
                    Datos Reales
              600
             500
            필 400
            등 300
             200
             100
                                                               108
                                                            Tiempo en dias
           <Figure size 432x288 with 0 Axes>
                    Datos del modelo matemático (polinomial)
              650
              600
              550
            е
              500
              450
              400
                                                                                             110
                                                                                                           111
```

Puntos extras: Investiga sobre la correlacion de variables y aplicar el calculo en base a los datos del Ecuador. Coeficiente de Correlación

106

107

negativa, con -1.00. No existe relación entre las variables cuando el coeficiente es de 0.00.

<Figure size 432x288 with 0 Axes>

Modelo Probabilístico

800

700

600

를 500 요

400

300

200

xy=crx*cry

105

 $r = \frac{N\sum XY - (\sum X)(\sum Y)}{\sqrt{N\sum X^2 - (\sum X)^2} * \sqrt{N\sum Y^2 - (\sum Y)^2}}$ In [14]: xx=crx**2 yy=cry**2

Es un valor cuantitativo de la relación entre dos o más variables. La coeficiente de correlación puede variar desde -1.00 hasta 1.00. La correlación de proporcionalidad directa o positiva se establece con los valores +1.00 y de proporcionalidad inversa o

Tiempo en dias

108 Tiempo en dias 109

110

111

Analisis Una vez analizados los datos con el modelo probabilistico,en el presente caso en el que tenemos distintas predicciones pero ninguna de ellas se aleja demasiado del valor real.

*sum(yy)-(sum(cry)**2)))

Conclusiones Podemos decir que con el modelo probabilistico la prediccion se centra en la obtencion de una coincidencia o aproximación entre los resultados de dos herramientas distintas, se puede predecir con mayor precisión el evento que ocurrirá a

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 el virus en • https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/