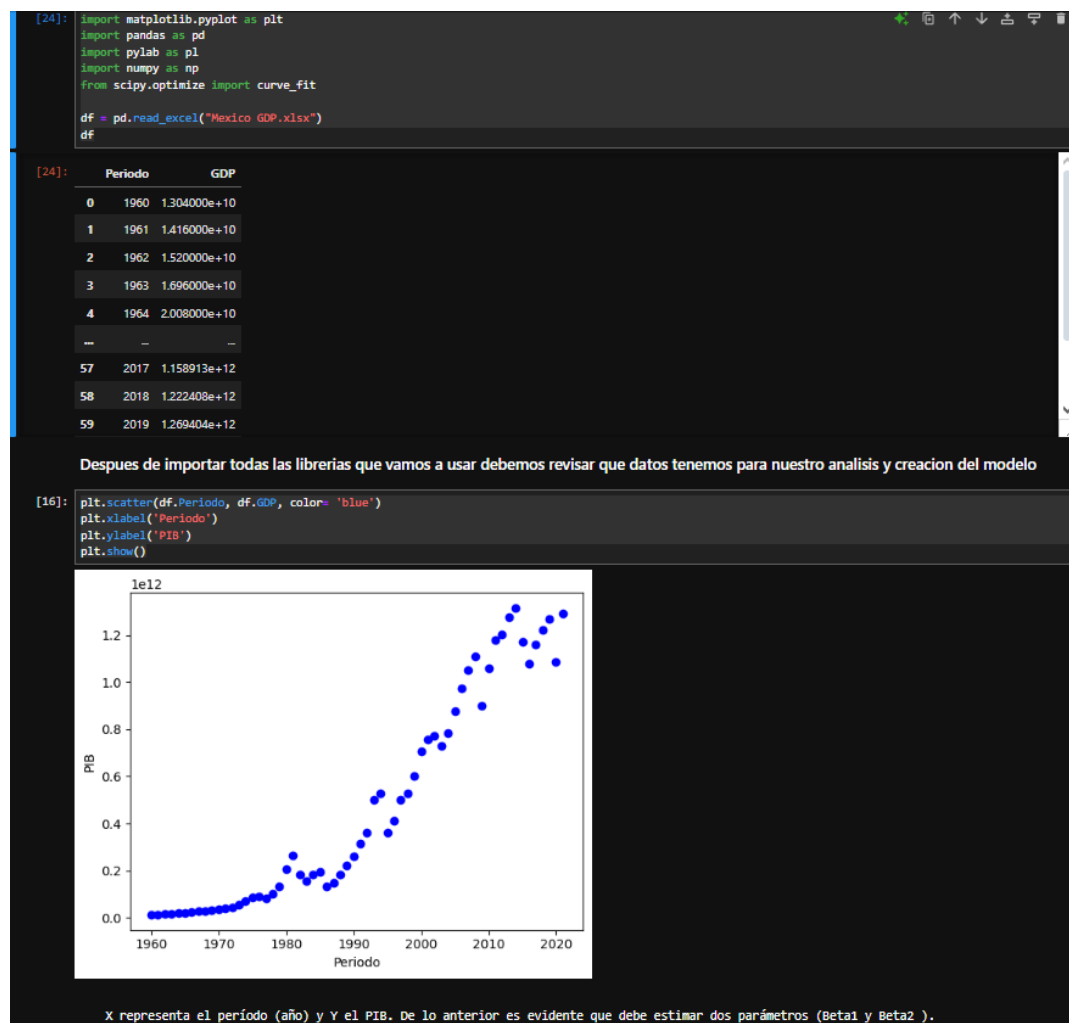


Creación de Modelo logístico para el PIB futuro de México

Mediante una regresión de mínimos cuadrados ajuste los datos del archivo del PIB de México a través del tiempo al modelo logístico siguiente:

$$\hat{Y} = \frac{1}{1 + e^{\beta_1(X - \beta_2)}}$$

Empecé por importar las librerías correspondientes y analizar si la regresión logística era la mejor opción



Sabiendo que los datos son correspondientes para una función de ese tipo hacemos las máscaras para el entrenamiento de nuestro modelo y la función logística con la formula dada hace un momento

Ahora que notamos que nuestros datos si tienen una señal de que cada vez que pasan los años aumenta el PIB pero que tienen un límite estamos hablando de una función logística

Que debemos hacer para empezar a desarrollar nuestro modelo?

Separamos por "mascaras para entrenar nuestro modelo"

```
[100]: # Dividimos en entrenamiento y prueba
msk = np.random.rand(len(df)) < 0.8
train = df[msk]
test = df[~msk]
```

Usar función logística

```
[126]: def logistic(x, beta1, beta2):
        return 1 / (1 + np.exp(beta1 * (x - beta2)))
```

Aquí surgió un problema, pero lo contestamos en las preguntas del final, normalizamos los valores de GDP al ser tan grandes les ponemos una especie de límite con esto. Creamos nuestra pendiente del modelo, aquí vamos a ajustar un inicio suave y un límite en los valores que se pueden alcanzar.

Solo faltaba probar el R2 de nuestro modelo como podemos ver es del 0.98, siendo un modelo bastante acertado

```
[106]: # Normalizamos los datos
pib_train_norm = train.GDP / max(df.GDP)
pib_test_norm = test.GDP / max(df.GDP)
```

El inicio de la pendiente suave

Esto para darle al optimizador una mejor pista inicial, y que inicie suave junto con la media del periodo

```
[108]: # Ajustamos el modelo con valores iniciales
p0 = [0.1, np.median(df.Periodo)]
popt, pcov = curve_fit(logistic, train.Periodo, pib_train_norm, p0=p0, maxfev=10000)
beta1, beta2 = pop
```

Creando predicciones

Aplica el modelo logístico ajustado a los años del conjunto de prueba para obtener las predicciones de PIB.

Donde R2 es la precisión del modelo siendo 1.0 el 100%

```
[136]: y_pred_test = logistic(test.Periodo, beta1, beta2)
r2_test = r2_score(pib_test_norm, y_pred_test)
print(f"R2 Score en datos de prueba: {r2_test:.2f}")
```

R2 Score en datos de prueba: 0.98

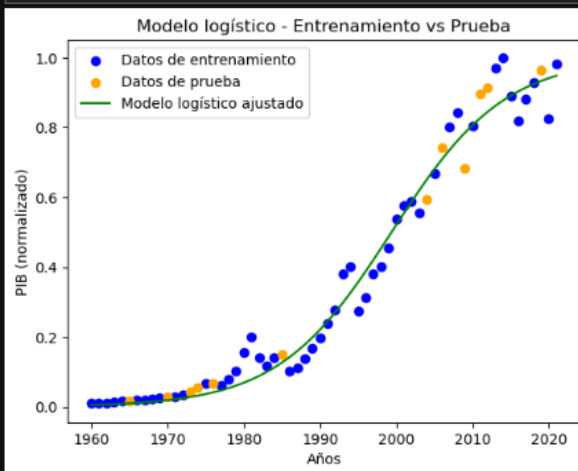
Ahora solo falta ver la gráfica de nuestro modelo, para eso creamos un scatter para ver la relacion de crecimiento del PIB con respecto a los años, donde vamos a ver también los datos de entrenamiento y de prueba en conjunto con nuestro modelo

Creando el grafico que prueba datos de test y train

Creamos un gráfico para representar los valores de entrenamiento, prueba y el modelo para ver que tan acertado fue

```
[118]: # Visualizacion del resultado

plt.scatter(train.Periodo, pib_train_norm, color='blue', label='Datos de entrenamiento')
plt.scatter(test.Periodo, pib_test_norm, color='orange', label='Datos de prueba')
x_range = np.linspace(min(df.Periodo), max(df.Periodo), 100)
plt.plot(x_range, logistic(x_range, beta1, beta2), color='green', label='Modelo logístico ajustado')
plt.legend()
plt.xlabel('Años')
plt.ylabel('PIB (normalizado)')
plt.title('Modelo logístico - Entrenamiento vs Prueba')
plt.show()
```



Y por último solo nos falta seguir con nuestro modelo, agregando unos cuantos años en nuestro modelo para que prediga el crecimiento del pib

Crear la predicción del pib del 2022 al 2027

```
[149]: #Crea una nueva lista de años que incluya los futuros
periodos_futuros = np.arange(max(df.Periodo) + 1, max(df.Periodo) + 7)

#Usa el modelo ajustado para predecir
prediccion_futura = logistic(periodos_futuros, beta1, beta2) * max(df.GDP)

#Muestra las predicciones
for año, pib in zip(periodos_futuros, prediccion_futura):
    print(f'Predicción para {año}: {pib:.2f}')

Predicción para 2022: 1255334744883.18
Predicción para 2023: 1262588718210.20
Predicción para 2024: 1268984175123.10
Predicción para 2025: 1274635659116.90
Predicción para 2026: 1279617703986.17
Predicción para 2027: 1284005126170.38
```

Preguntas relacionadas con el modelo predictivo

¿Tuvo algún problema en el proceso de estimación de parámetros?

- Si, Al usar datos tan grandes Python no puede manejar estos números así, entonces tenemos que normalizarlos y limitarlos

¿Obtuve resultados satisfactorios en esta ocasión?

- Si, al normalizarlos los resultados y los datos pueden manejarse mucho más fácil en este ejercicio, además de acertados

¿Cuál sería el pronóstico del PIB (GDP) para México en el 2022?

- Predicción para 2022: 1255334744803.18

¿Qué mecanismo de transformación tendría que realizar en dicho caso?

- Tuve que hacer una lista con los años futuros tomando en cuenta el último año que está en nuestro Df y con esta lista hecha solo tomamos en cuenta los valores de beta 1 y 2 para usar nuestro modelo, así obtuvimos no solo la predicción del 2022 si no de 5 años mas