

## DOMINGUEZ AGUILAR SARAH ITZEL

Considero que el método esta correctamente ya que como vimos en clase primero identificamos X y Y, en este caso Y(promedio salarial) depende de X(tiempo), luego procede a realizar el diagrama de dispersión y encontrar los mínimos cuadrados.

```
In [ ]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [7]: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
In [8]: datos=pd.read_csv("C:\SalarioMinimo.csv")
datos
```

Out[8]:

	Tiempo	Promedio_Salarial
0	2000	101.57
1	2001	88.62
2	2002	89.26
3	2003	89.21
4	2004	88.83
5	2005	89.26
6	2006	89.59
7	2007	89.52
8	2008	88.59
9	2009	88.00
10	2010	88.59
11	2011	89.19
12	2012	89.30
13	2013	89.72
14	2014	89.75
15	2015	91.57
16	2016	94.49
17	2017	98.48
18	2018	102.78
19	2019	122.00
20	2020	139.37
21	2021	152.23
22	2022	174.54

```
In [10]: datos.shape
```

Out[10]: (23, 2)

```
In [11]: #Asigna Los valores de Tiempo en x
x=datos.iloc[:,1].values
x
```

```
Out[11]: array([[2000],
 [2001],
 [2002],
 [2003],
 [2004],
 [2005],
 [2006],
 [2007],
 [2008],
 [2009],
 [2010],
 [2011],
 [2012],
 [2013],
 [2014],
 [2015],
 [2016],
 [2017],
 [2018],
 [2019],
 [2020],
 [2021],
 [2022]], dtype=int64)
```

```
In [9]: #Asigna Los valores de Promedio_Salarial en y
y=datos.iloc[:, 1].values
y
```

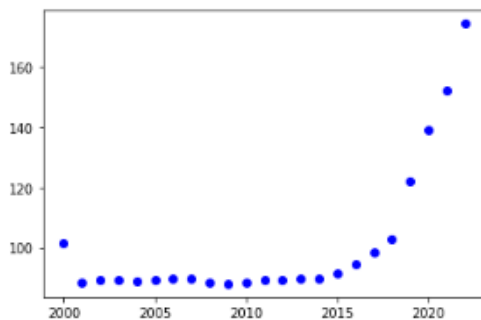
```
Out[9]: array([101.57, 88.62, 89.26, 89.21, 88.83, 89.26, 89.59, 89.52,
 88.59, 88. , 88.59, 89.19, 89.3 , 89.72, 89.75, 91.57,
 94.49, 98.48, 102.78, 122. , 139.37, 152.23, 174.54])
```

```
In [12]: #Asignamos a La variable el metodo de Mínimos cuadrados ordinarios Regresión Lineal.
regressor=LinearRegression()
```

```
In [13]: #Ajuste al modelo Lineal.
regressor.fit(x,y)
```

```
Out[13]: LinearRegression()
LinearRegression()
```

```
In [14]: #Diagrama de dispersion con X y Y asignandole color azul
plt.scatter(x,y, color='blue');
```



```
In [17]: #Graficacion de diagrama de dirpercion dandole mejor formato y añadiendo línea con referencia en x
plt.scatter(x,y, color='blue')
plt.plot(x, regressor.predict(x), color='red')
plt.title('Salarios')
plt.xlabel('años')
plt.ylabel('Pesos');
```

