**Pasos para implementar un modelo**

**Primero:** Importar las librerías necesarias

* import **pandas as** pd
* import **matplotlib.pyplot** as plt
* from **sklearn.model\_selection** import **train\_test\_split** #Para separar nuestro dataset
* from **sklearn.linear\_model** import **LinearRegression**

**Segundo:** Cargamos el dataset con el que vamos a trabajar.

dataset = **pd.read.csv**(‘salarios.csv’)

**Notita:** “.iloc” es para localizar cuáles son los datos que debemos utilizar para entrenar

**x** = **dataset.iloc[: , : -1].**values #Con values estamos accediendo a los valores

**y** = **dataset.iloc[: , 1 ].**values

**Tercero:** Pasamos a separar los datos para entrenamiento y verificación

**X\_train**, X\_test, **Y\_train**, Y\_test = **train\_test\_split(x , y , test\_size =0,2, random\_state =0)**

**Cuarto:** Pasamos a preparar nuestro modelo linear, logística, etc.

regressor = **LinearRegression( )**

**Quinto:** Pasamos a entrenar nuestro modelocon **“.fit”**

regressor**.fit**(**X\_train, Y\_train**)

**Sexto:** Procedemos a mostrar el resultado del entrenamiento

viz\_train = plt #Instanciamos una gráfica de pyplot

viz\_train.scatter(**X\_train, Y\_train**, color = ‘blue’) #Mostramos nuestros elementos

viz\_train.plot(**X\_train, regressor.predict(X\_train),** color = ‘black’)

Procedemos a indicar las etiquetas de los ejes

viz\_train.title(‘Salario Vs Experiencia’)

viz\_train.xlabel(‘Experiencia’)

viz\_train.YLABEL(‘Salario’)

viz\_train.show( )

**Sexto:** Procedemos a mostrar el resultado del entrenamiento

viz\_train = plt #Instanciamos una gráfica de pyplot

viz.train.scatter(**X\_train, Y\_train**, color = ‘blue’) #Mostramos nuestros elementos (puntos)

viz\_train.plot(**X\_train, regressor.predict(X\_train),** color = ‘black’)

Procedemos a indicar las etiquetas de los ejes

viz\_train.title(‘Salario Vs Experiencia’)

viz\_train.xlabel(‘Experiencia’)

viz\_train.YLABEL(‘Salario’)

viz\_train.show( )

**Séptimo:** Procedemos a mostrar el resultado del entrenamiento, pero con nuestros datos de prueba

viz\_train = plt #Instanciamos una gráfica de pyplot

viz.train.scatter(**X\_test, Y\_test**, color = ‘red’) #Mostramos nuestros elementos (puntos)

viz\_train.plot(**X\_train, regressor.predict(X\_train),** color = ‘black’) #La misma reg. lineal

Procedemos a indicar las etiquetas de los ejes

viz\_train.title(‘Salario Vs Experiencia’)

viz\_train.xlabel(‘Experiencia’)

viz\_train.YLABEL(‘Salario’)

viz\_train.show( )

**random.randint(Start, Stop):** escoge aleatoriamente un núemro entre Start y Stop

**np.random.randn(10) :** Crea una lista de 10 números aleatorios entre [0, 1]

**Dato:** Si queremos agregar una fila adicional a nuestro data o archivo .csv

datos = pd.read\_csv(‘energy.csv’ , header = ‘0’)

Una opción es