**Machine Learning**

**Ex: 04 Linear regression**

**Name:**Athithraja. R

**Reg.no:** 2022503702

1) Linear regression(Hard Code)

**Code:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

df = pd.read\_csv('Salary\_Data.csv')

X = df['YearsExperience']

y = df['Salary']

n = len(X)

sum\_X = np.sum(X)

sum\_Y = np.sum(y)

sum\_XY = np.sum(X \* y)

sum\_X\_squared = np.sum(X \*\* 2)

m = (n \* sum\_XY - sum\_X \* sum\_Y) / (n \* sum\_X\_squared - sum\_X \*\* 2)

b = (sum\_Y - m \* sum\_X) / n

print(f"Slope (m): {m}")

print(f"Intercept (b): {b}")

y\_pred = m \* X + b

plt.scatter(X, y, color='blue', label='Actual Data')

plt.plot(X, y\_pred, color='red', label='Regression Line')

plt.title('Years of Experience vs Salary')

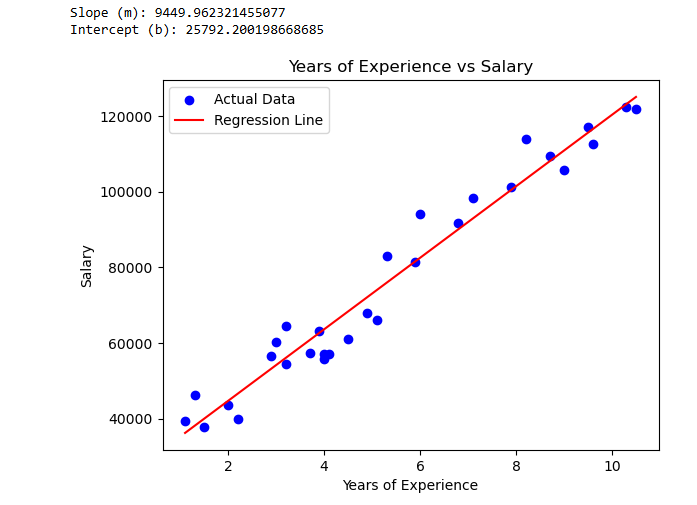
plt.xlabel('Years of Experience')

plt.ylabel('Salary')

plt.legend()

plt.show()

**Output:**



2) Linear Regression (sklearn):

**Code:**

import pandas as pd

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

df = pd.read\_csv('Salary\_Data.csv')

X = df[['YearsExperience']]

y = df['Salary']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

model = LinearRegression()

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

import matplotlib.pyplot as plt

plt.scatter(X, y, color='blue')

plt.plot(X, model.predict(X), color='red')

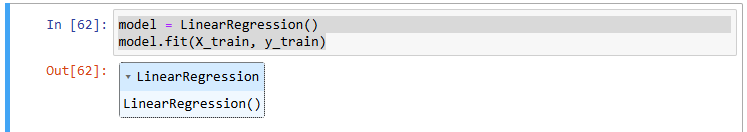
plt.title('Years of Experience vs Salary')

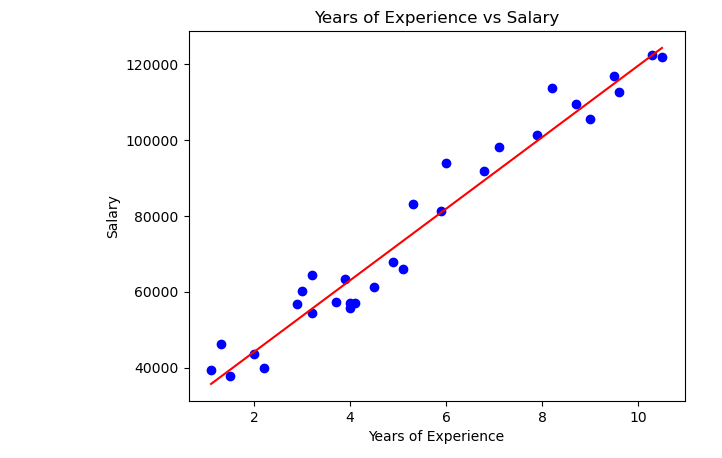
plt.xlabel('Years of Experience')

plt.ylabel('Salary')

plt.show()

**Output:**





**Actual Excersice:**

1. Simple linear regression predict price of a car based on its milage:

**Code:**

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

import matplotlib.pyplot as plt

df = pd.read\_excel('car\_sales.xlsx')

X = df[['Milage']]

y = df['price']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

model = LinearRegression()

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

plt.scatter(X, y, color='blue')

plt.plot(X, model.predict(X), color='red')

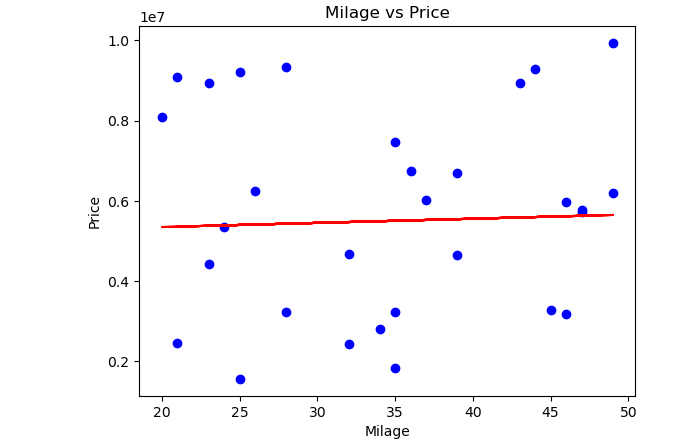
plt.title('Milage vs Price')

plt.xlabel('Milage')

plt.ylabel('Price')

plt.show()

**Output:**



2. To Predict sales of car in us based on milage & price

**Code:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

df = pd.read\_excel('car\_sales 1.xlsx')

X = df[['Milage', 'price']]

y = df['sales']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

model = LinearRegression()

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

fig = plt.figure(figsize=(10, 7))

ax = fig.add\_subplot(111, projection='3d')

ax.scatter(df['Milage'], df['price'], df['sales'], color='blue', label="Actual Sales")

ax.scatter(X\_test['Milage'], X\_test['price'], y\_pred, color='red', label="Predicted Sales", marker='^')

ax.set\_xlabel('Milage')

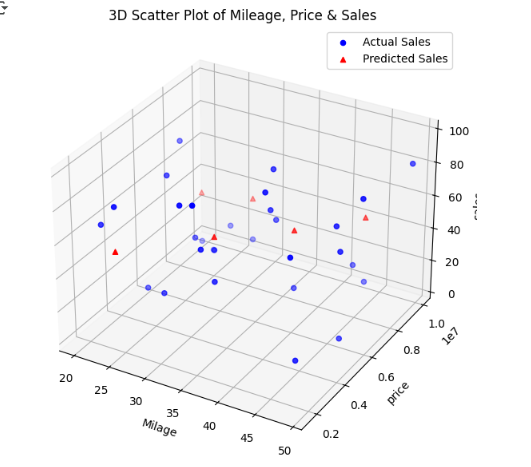
ax.set\_ylabel('price')

ax.set\_zlabel('sales')

ax.set\_title('3D Scatter Plot of Mileage, Price & Sales')

ax.legend()

plt.show()



**Output:**