Linear_Regression_User_defined

December 7, 2025

User defined Regression Analysis

Equation : $y = \beta_0 + \beta_1 x$

- This is the simple linear regression equation where β_0 is the constant and β_1 is the slope and describes the relationship between x (independent variable) and y(dependent variable)
- $y_i = \beta_0 + \beta_1 x_i$, β_0 ($y-$intercept) and β_1 (slope) are the coefficients whose values represent the accuracy of predicted values with the actual values.

[1]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
```

[2]:
```
# Get dataset
df_sal = pd.read_csv('Salary_Data.csv')
df_sal.head()
```

[2]:

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0

[3]:
```
# Describe data
df_sal.describe()
```

[3]:

	YearsExperience	Salary
count	30.000000	30.000000
mean	5.313333	76003.000000
std	2.837888	27414.429785
min	1.100000	37731.000000
25%	3.200000	56720.750000
50%	4.700000	65237.000000
75%	7.700000	100544.750000
max	10.500000	122391.000000

[4]:
```
x_actual=df_sal["YearsExperience"]
y_actual=df_sal["Salary"]
```

[5]:
```
x_bar=np.mean(x_actual)
y_bar=np.mean(y_actual)
n=len(x_actual)
print('n=',n, 'x_bar=', x_bar, 'y_bar=', y_bar)
```

```
n= 30 x_bar= 5.3133333333333335 y_bar= 76003.0
```

[6]:
```
#Slope and Constant
N=0
D=0
for i in range(n):
    N=N+(x_actual[i]-x_bar)*(y_actual[i]-y_bar)
    D=D+(x_actual[i]-x_bar)**2
beta1=N/D
beta0=y_bar-beta1*x_bar
print("Slope beta1=",beta1)
print("Intercept beta0=",beta0)
```

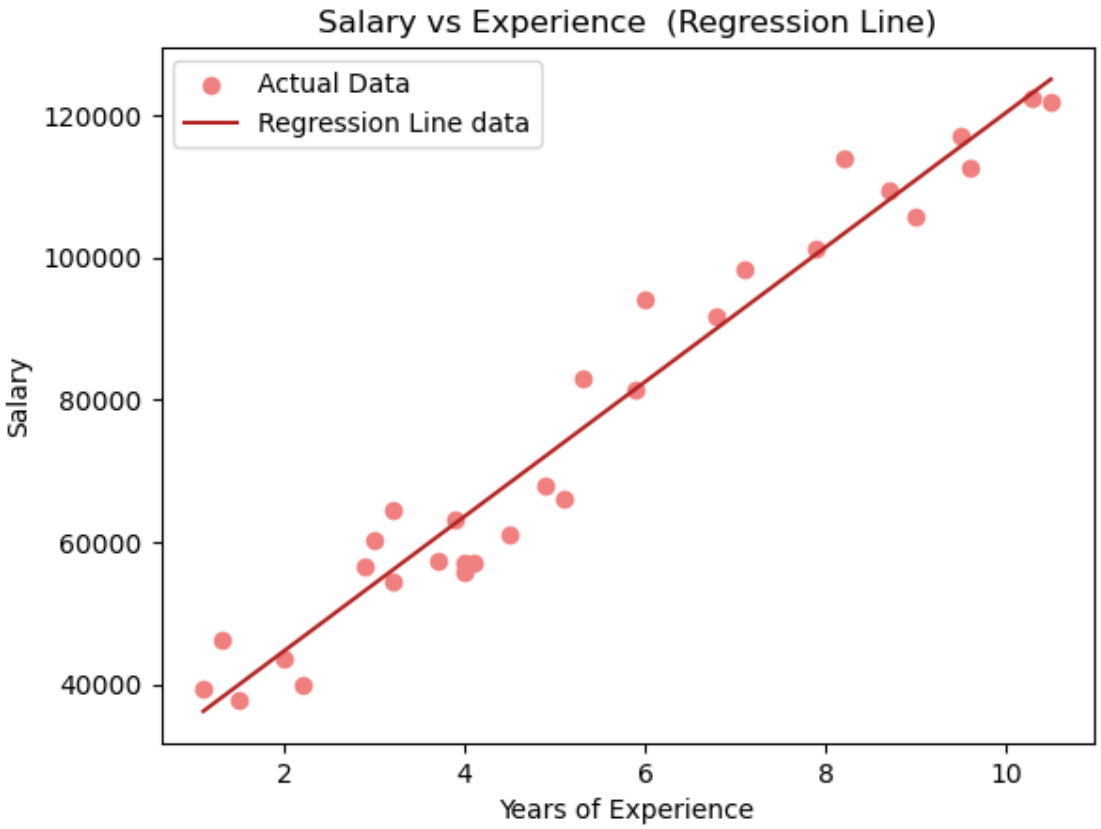
```
Slope beta1= 9449.962321455077
Intercept beta0= 25792.20019866869
```

[7]:
```
print("Line of regression is y=",round(beta0,2),"+",round(beta1,2),"x")
```

```
Line of regression is y= 25792.2 + 9449.96 x
```

[8]:
```
# Predict
y_pred=np.zeros(n)
for i in range(n):
    y_pred[i]=beta0+beta1*x_actual[i]
```

[9]:
```
# Prediction
x_val=df_sal['YearsExperience']
y_val=y_pred
plt.scatter(x_actual, y_actual, color = 'lightcoral')
plt.plot(x_val, y_val, color = 'firebrick')
plt.title('Salary vs Experience  (Regression Line)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.legend(['Actual Data', 'Regression Line data', ], loc='best',␣
 ↪facecolor='white')
plt.box(True)
plt.show()
```


Salary vs Experience (Regression Line)
Actual Data
Regression Line data
120000
100000
80000
60000
40000
Salary
2
4
6
8
10
Years of Experience

[ ]:

[ ]: