WEEK-8

Linear Regression

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
salary=pd.read_csv("Salary_dataset.csv")
salary.head()
```

```
   Unnamed: 0  YearsExperience   Salary
0           0              1.2  39344.0
1           1              1.4  46206.0
2           2              1.6  37732.0
3           3              2.1  43526.0
4           4              2.3  39892.0
```

```
salary.describe()
```

```
       Unnamed: 0  YearsExperience         Salary
count   30.000000        30.000000      30.000000
mean    14.500000         5.413333   76004.000000
std      8.803408         2.837888   27414.429785
min      0.000000         1.200000   37732.000000
25%      7.250000         3.300000   56721.750000
50%     14.500000         4.800000   65238.000000
75%     21.750000         7.800000  100545.750000
max     29.000000        10.600000  122392.000000
```

```
salary.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30 entries, 0 to 29
Data columns (total 3 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Unnamed: 0       30 non-null     int64  
 1   YearsExperience  30 non-null     float64
 2   Salary           30 non-null     float64
dtypes: float64(2), int64(1)
memory usage: 848.0 bytes
```

```
salary.rename({
    "Unnamed: 0":"Id"
},inplace=True,axis=1)
```

```
salary.columns
```

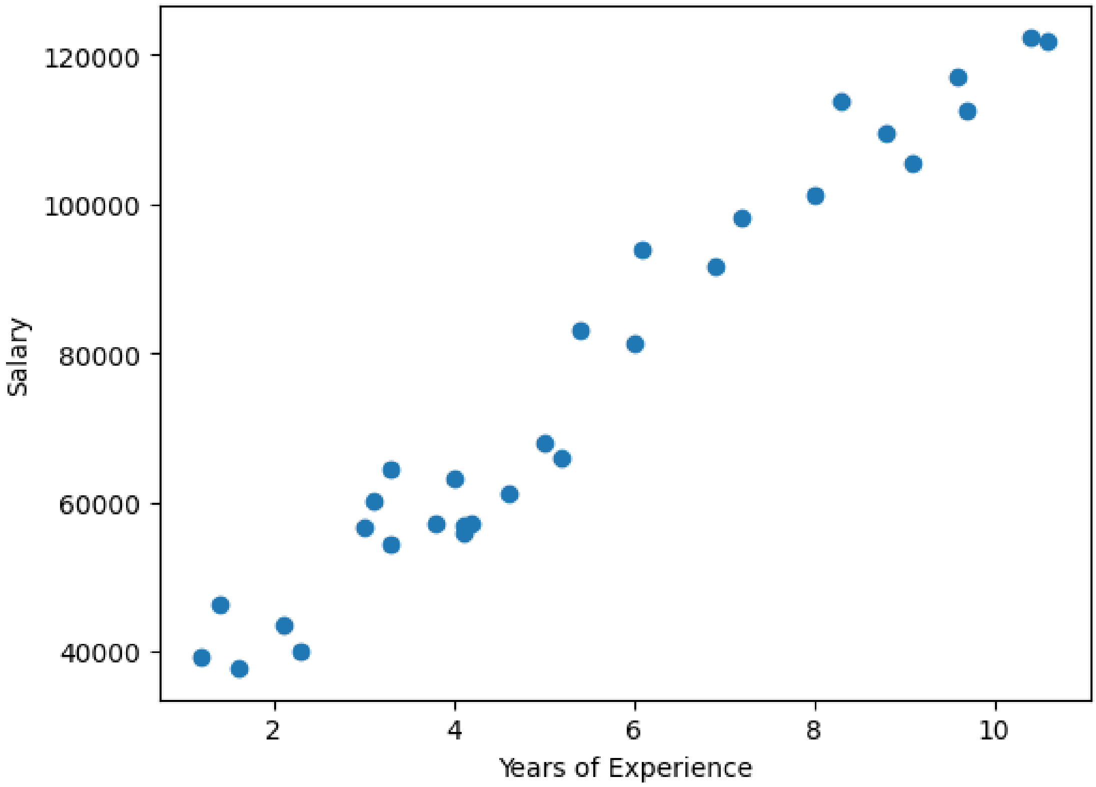
```
Index(['Id', 'YearsExperience', 'Salary'], dtype='object')
```

```
salary.isnull().sum()
```

```
Id                0
YearsExperience    0
Salary            0
dtype: int64
```

```
plt.xlabel("Years of Experience")
plt.ylabel("Salary")
plt.scatter(x=salary["YearsExperience"],y=salary["Salary"])
```

```
<matplotlib.collections.PathCollection at 0x1c382dbb940>
```



```
X=salary.drop("Salary",axis=1)
X.head()
```

```
  Id  YearsExperience
0   0             1.2
1   1             1.4
2   2             1.6
3   3             2.1
4   4             2.3
```

```
Y=salary["Salary"]
Y.head()
```

```
0    39344.0
1    46206.0
2    37732.0
```

```
3    43526.0
4    39892.0
Name: Salary, dtype: float64
```

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,random_state=10,test_size=0.1)
```

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(X_train,Y_train)
model.score(X_test,Y_test)
```

```
0.9824411734400897
```

```
y_predict=model.predict(X_test)
```

```
y_array=np.array(Y_test)
y_array[0]
```

```
91739.0
```

```
X_array=np.array(X_test)
X_array[0]
```

```
array([20. ,  6.9])
```

```
coeff=model.coef_
```

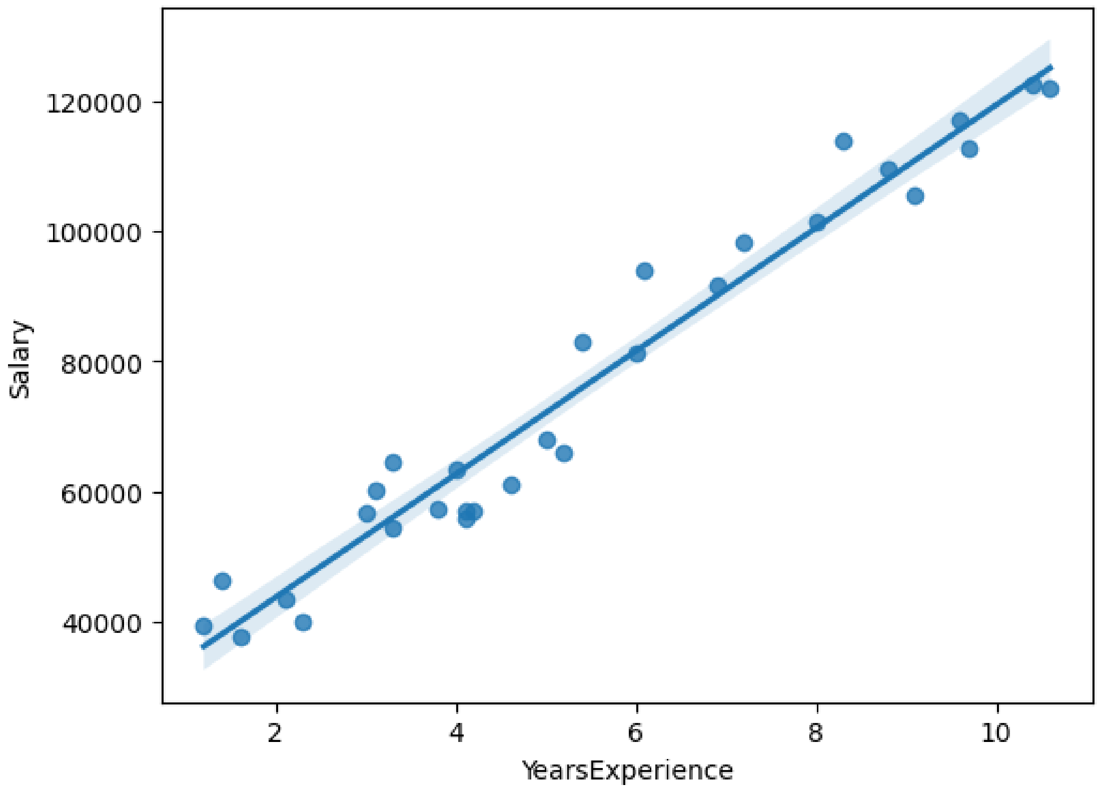
```
intercept=model.intercept_
```

```
coeff*6.9+intercept
```

```
array([20637.76340882, 98767.29358298])
```

```
sns.regplot(x=salary["YearsExperience"],y=salary["Salary"])
```

```
<Axes: xlabel='YearsExperience', ylabel='Salary'>
```



```
X_= np.array([[1, 1], [1, 2], [2, 2], [2, 3]])
y_ = np.dot(X, np.array([1, 2])) + 3
```

```
model.score(X_, y_)
```

```
C:\Users\ASUS\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.py:420:
UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

```
-636283438.4136908
```

```
model.coef_
```

```
array([ -462.79629147, 10860.32402362])
```

```
model.intercept_
```

```
23831.05781999221
```

```
model.predict(np.array([[3, 5]]))
```

```
C:\Users\ASUS\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.py:420:
UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

```
array([76744.28906368])
```