Simple Linear Regression

Importing the libraries

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

Importing the dataset

In [2]:
```
dataset = pd.read_csv(r"C:\Users\91628\Desktop\CSV\Salary_Data.csv")
dataset
```

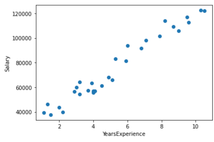
Out[2]:

	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
5	2.9	56642.0
6	3.0	60150.0
7	3.2	54445.0
8	3.2	64445.0
9	3.7	57189.0
10	3.9	63218.0
11	4.0	55794.0
12	4.0	56957.0
13	4.1	57081.0
14	4.5	61111.0
15	4.9	67938.0
16	5.1	66029.0
17	5.3	83088.0
18	5.9	81363.0
19	6.0	93940.0
20	6.8	91738.0
21	7.1	98273.0
22	7.9	101302.0
23	8.2	113812.0
24	8.7	109431.0
25	9.0	105582.0
26	9.5	116969.0
27	9.6	112635.0
28	10.3	122391.0
29	10.5	121872.0

In [ ]:
```

```

In [3]:
```
plt.scatter(dataset['YearsExperience'],dataset['Salary'])
plt.xlabel("YearsExperience")
plt.ylabel("Salary")
```

Out[3]:
```
Text(0, 0.5, 'Salary')
```



In [ ]:
```

```

In [ ]:
```

```

In [4]:
```
X = dataset.iloc[:,:-1].values
```

In [5]:
```
X
```

Out[5]:
```
array([[ 1.1],
       [ 1.3],
       [ 1.5],
       [ 2. ],
       [ 2.2],
       [ 2.9],
       [ 3. ],
       [ 3.2],
       [ 3.2],
       [ 3.7],
       [ 3.9],
       [ 4. ],
       [ 4. ],
       [ 4.1],
       [ 4.5],
       [ 4.9],
       [ 5.1],
       [ 5.3],
       [ 5.9],
       [ 6. ],
       [ 6.8],
       [ 7.1],
       [ 7.9],
       [ 8.2],
       [ 8.7],
       [ 9. ],
       [ 9.5],
       [ 9.6],
       [10.3],
       [10.5]])
```

In [6]:
```
y = dataset.iloc[:, -1].values
```

In [7]:
```
y
```

Out[7]:
```
array([ 39343.,  46205.,  37731.,  43525.,  39891.,  56642.,  60150.,
        54445.,  64445.,  57189.,  63218.,  55794.,  56957.,  57081.,
        61111.,  67938.,  66029.,  83088.,  81363.,  93940.,  91738.,
        98273., 101302., 113812., 109431., 105582., 116969., 112635.,
       122391., 121872.])
```

Splitting the dataset into the Training set and Test set

In [8]:
```
from sklearn.model_selection import train_test_split


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state =0)

print("X_train is " ,X_train.shape)
print("X_test is " ,X_test.shape)
print("Y_train is " ,y_train.shape)
print("Y_test is " ,y_test.shape)
```

```
X_train is  (24, 1)
X_test is  (6, 1)
Y_train is  (24,)
Y_test is  (6,)
```

In [9]:
```
X_test
```

Out[9]:
```
array([[ 1.5],
       [10.3],
       [ 4.1],
       [ 3.9],
       [ 9.5],
       [ 8.7]])
```

In [10]:
```
y_test
```

Out[10]:
```
array([ 37731., 122391.,  57081.,  63218., 116969., 109431.])
```

In [ ]:
```

```

In [ ]:
```

```

Training the Simple Linear Regression model on the Training set

In [11]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

Out[11]:
```
LinearRegression()
```

In [ ]:
```

```

Predicting the Test set results

In [12]:
```
y_pred = regressor.predict(X_test)
y_pred
```

Out[12]:
```
array([  40748.96184072, 122699.62295594,  64961.65717022,  63099.14214487,
        115249.56285456, 107799.50275317])
```

In [13]:
```
y_test
```

Out[13]:
```
array([ 37731., 122391.,  57081.,  63218., 116969., 109431.])
```

In [ ]:
```

```

In [14]:
```
y_pred = regressor.predict([[11]])
y_pred
```

Out[14]:
```
array([129218.42554465])
```

In [ ]:
```

```

In [15]:
```
m = regressor.coef_
```

In [16]:
```
m
```

Out[16]:
```
array([9312.57512673])
```

In [17]:
```
c = regressor.intercept_
```

In [18]:
```
y = m*(11)+c
```

In [19]:
```
print(y)
```

```
[129218.42554465]
```

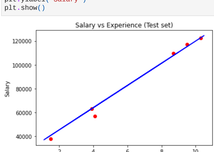
In [ ]:
```

```

Visualising the Training set results

In [20]:
```
plt.scatter(X_train, y_train, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Training set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
```

Visualising the Test set results

In [21]:
```
plt.scatter(X_test, y_test, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Test set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
```



In [ ]:
```

```