Superwise learning lab -2

NAME:- Ayub Alam

Rollno:-TNU2021053100031L

Assigment Date:-

Submission date:-

Gradient desent

In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

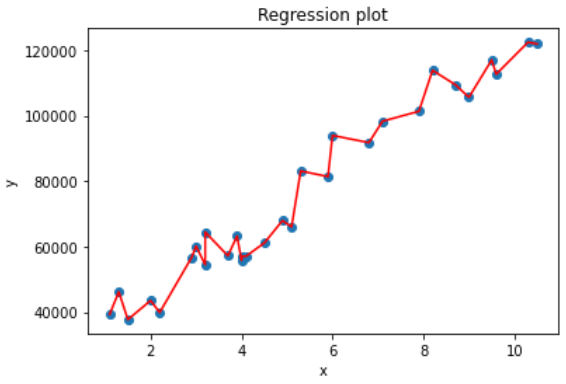
In [3]:
```
# here upload the linear_regression data set
data=pd.read_csv(r'linear_regression.csv')
data
```

Out[3]:

	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 [4]:
```
# here we define dependent and dependent variable
x=data.iloc[:,0]
y=data.iloc[:,1]
```

In [5]:
```
#here drow the graph between independent and dependent variable
plt.scatter(x, y)
plt.plot(x,y,color='red')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Regression plot')
plt.show()
```



In [6]:
```
#here we define slope,constant and l= learning rate , e is epouch
# here epouch means no of iteretion
m=0
c=0
L=0.0001
e=1000
n=float(len(x))
for i in range (e):
    y_pred=m*x+c
    d_m=(-2/n)*sum(x*(y-y_pred))
    d_c=(-2/n)*sum(y-y_pred)
    m=m - L*d_m
    c=c - L*d_c
print(m, c)
```

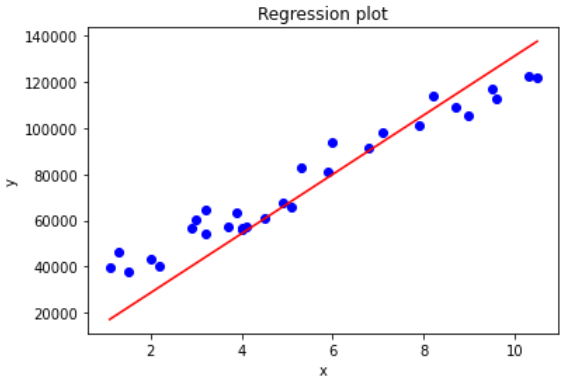
```
12836.600965885045 2915.2044856014018
```

In [7]:
```
# here we find the coeficient
y_prd=m*x+c
y_prd
```

Out[7]:
```
0      17035.465548
1      19602.785741
2      22170.105934
3      28588.406417
4      31155.726611
5      40141.347287
6      41425.007383
7      43992.327576
8      43992.327576
9      50410.628059
10     52977.948253
11     54261.608349
12     54261.608349
13     55545.268446
14     60679.908832
15     65814.549218
16     68381.869412
17     70949.189605
18     78651.150184
19     79934.810281
20     90204.091054
21     94055.071343
22    104324.352116
23    108175.332406
24    114593.632889
25    118444.613179
26    124862.913662
27    126146.573758
28    135132.194434
29    137699.514627
Name: YearsExperience, dtype: float64
```

In [9]:
```
#here draw the scatter plot between x,y
plt.scatter(x,y,color='Blue')
plt.plot(x,y_prd,color='red')
plt.xlabel('x')
plt.ylabel('y')
```

```
plt.title('Regression plot')
plt.show()
```



In [11]:
```
#use the sklearn model for spliting the testing or training data
from sklearn.model_selection import train_test_split
```

In [13]:
```
#here we spliting the hole data in 1/3(there is X_trian,y_train (75) for training and 25 testing)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=1/3,random_state=0)
x_train
```

Out[13]:
```
5      2.9
16     5.1
8      3.2
14     4.5
23     8.2
20     6.8
1      1.3
29    10.5
6      3.0
4      2.2
18     5.9
19     6.0
9      3.7
7      3.2
25     9.0
3      2.0
0      1.1
21     7.1
15     4.9
12     4.0
Name: YearsExperience, dtype: float64
```

In [14]:
```
x_train
```

Out[14]:
```
5      2.9
16     5.1
8      3.2
14     4.5
23     8.2
20     6.8
1      1.3
29    10.5
6      3.0
4      2.2
18     5.9
19     6.0
9      3.7
7      3.2
25     9.0
3      2.0
0      1.1
21     7.1
15     4.9
12     4.0
Name: YearsExperience, dtype: float64
```

In [15]:
```
y_train
```

Out[15]:
```
5      56642.0
16     66029.0
8      64445.0
14     61111.0
23    113812.0
20     91738.0
1      46205.0
29    121872.0
6      60150.0
4      39891.0
```

```
18     81363.0
19     93940.0
9      57189.0
7      54445.0
25    105582.0
3      43525.0
0      39343.0
21     98273.0
15     67938.0
12     56957.0
Name: Salary, dtype: float64
```

In [27]:
```
l=0.0001
m=0
c=0
epochs=1000
n=float(len(x_train))
n
```

Out[27]:
```
20.0
```

In [28]:
```
#here we write the graident desent formula for finding the coeficient and intercept
for i in range(epochs):
    y_pred=m*x_train+c
    d_m=(-2/n)*sum(x_train*(y_train-y_pred))
    d_c=(-2/n)*sum(y_train-y_pred)
    m=m - l*d_m
    c=c - l*d_c
print(m,c)
```

```
13210.62109711793 3259.361202893137
```

In [29]:
```
y_pred=m*x_train+c
y_pred
```

Out[29]:
```
5      41570.162385
16     70633.528798
8      45533.348714
14     62707.156140
23    111586.454199
20     93091.584663
1      20433.168629
29    141970.882723
6      42891.224494
4      32322.727617
18     81202.025676
19     82523.087786
9      52138.659262
7      45533.348714
25    122154.951077
3      29680.603397
0      17791.044410
21     97054.770992
15     67991.404579
12     56101.845591
Name: YearsExperience, dtype: float64
```

In [31]:
```
#here we draw the scatter plot  between x,y and draw line between x_train,y_pred
plt.scatter(x_train,y_train,color='Blue')
plt.plot(x_train,y_pred,color='red')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Regression plot')
plt.show()
```