DATA SCIENCE AND BUSINESS ANALYTICS INTERNSHIP

THE SPARKS FOUNDATION, GRADUATE ROTATIONAL INTERNSHIP PROGRAM (GRIPJULY21)

Task-1: Prediction using supervised machine learning

problem statement: What will be predicted score if a student studies for 9.25hrs/day ?

Author: Rakesh kumar mandal

importing all important librarires required for this task

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
```

Reading data from the given link

In [2]:
```
url = "http://bit.ly/w-data"
dataset = pd.read_csv("http://bit.ly/w-data")
print("Data imported successfully")
dataset
```

```
Data imported successfully
```

Out[2]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30
5	1.5	20
6	9.2	88
7	5.5	60
8	8.3	81
9	2.7	25
10	7.7	85
11	5.9	62
12	4.5	41
13	3.3	42
14	1.1	17
15	8.9	95
16	2.5	30
17	1.9	24
18	6.1	67
19	7.4	69
20	2.7	30

	Hours	Scores
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

Description of datasets

In [3]:
```
dataset.describe()
```

Out[3]:

	Hours	Scores
count	25.000000	25.000000
mean	5.012000	51.480000
std	2.525094	25.286887
min	1.100000	17.000000
25%	2.700000	30.000000
50%	4.800000	47.000000
75%	7.400000	75.000000
max	9.200000	95.000000

In [4]:
```
dataset.shape
```

Out[4]:
```
(25, 2)
```

In [5]:
```
X = dataset.iloc[:, :-1].values
#print(X)
X
```

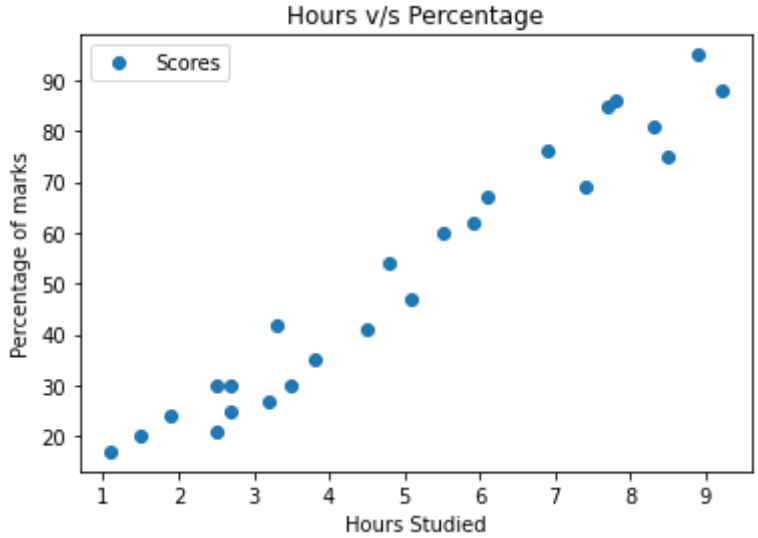
Out[5]:
```
array([[2.5],
       [5.1],
       [3.2],
       [8.5],
       [3.5],
       [1.5],
       [9.2],
       [5.5],
       [8.3],
       [2.7],
       [7.7],
       [5.9],
       [4.5],
       [3.3],
       [1.1],
       [8.9],
       [2.5],
       [1.9],
       [6.1],
       [7.4],
       [2.7],
       [4.8],
       [3.8],
       [6.9],
       [7.8]])
```

In [6]:
```
Y = dataset.iloc[:,1].values
#print(Y)
Y
```

Out[6]:
```
array([21, 47, 27, 75, 30, 20, 88, 60, 81, 25, 85, 62, 41, 42, 17, 95, 30,
       24, 67, 69, 30, 54, 35, 76, 86])
```

Plotting the distribution of scores

In [7]:
```
dataset.plot(x='Hours', y='Scores', style='o')
plt.title('Hours v/s Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage of marks')
plt.show()
```



From the above figure its clearly visible that the percentage of scores and hours of study are positively related to each other

Splitting the dataset into training and test sets

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)
```

Training the algorithm

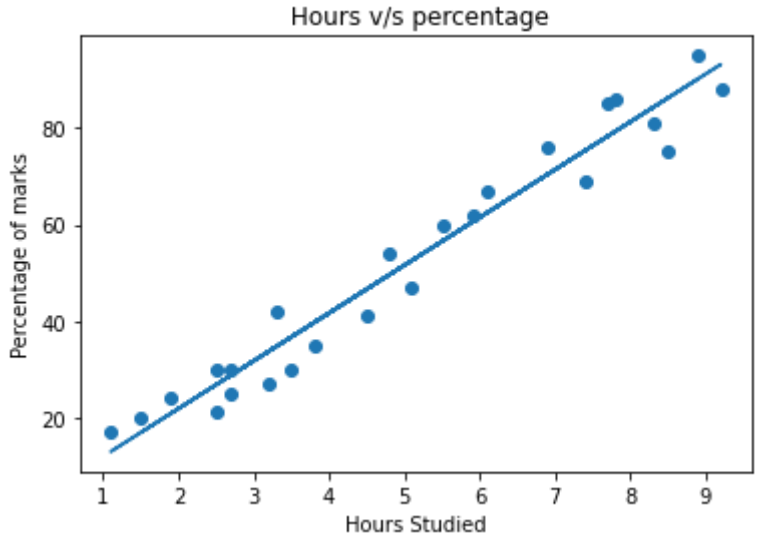
In [9]:
```
regressor = LinearRegression()
regressor.fit(X_train, Y_train)

print("Training complete.")
```

```
Training complete.
```

Plotting the regression line

In [10]:
```
line = regressor.coef_*X+regressor.intercept_
```

In [11]:
```
plt.scatter(X, Y)
plt.plot(X, line)
plt.xlabel('Hours Studied')
plt.ylabel('Percentage of marks')
plt.title(' Hours v/s percentage')
plt.show()
```



Making predictions about the datasets

In [12]:
```
print(X_test) # Testing data - In Hours
Y_pred = regressor.predict(X_test) # Predicting the scores
```

```
[[1.5]
 [3.2]
 [7.4]
 [2.5]
 [5.9]]
```

In [13]:
```
print(Y_test)
```

```
[20 27 69 30 62]
```

In [14]:
```
# Comparing Actual vs Predicted
df = pd.DataFrame({'Actual': Y_test, 'Predicted': Y_pred})
df
```

Out[14]:

	Actual	Predicted
0	20	16.884145
1	27	33.732261
2	69	75.357018
3	30	26.794801
4	62	60.491033

In [15]:
```
X_testN=np.append (X_test, [9.25])
print(X_testN)
```

```
[1.5  3.2  7.4  2.5  5.9  9.25]
```

Calculating predicted score of student

In [16]:
```
hours = [9.25]
answer = regressor.predict([hours])
print("No. of hours = {}".format(hours))
print("Predicted score = {}".format(round(answer[0],3)))
```

```
No. of hours = [9.25]
Predicted score = 93.692
```

Evaluating the performance of the algorithm

In [17]:
```
from sklearn import metrics
print('Mean Absolute Error = ',
      metrics.mean_absolute_error(Y_test, Y_pred))
print('r2 score= ', metrics.r2_score(Y_test, Y_pred))
```

```
Mean Absolute Error =  4.183859899002982
r2 score=  0.9454906892105354
```

The value of R2 represents the goodness of fit of the model such that the percentage of varition in dependent variable explained by independent varible.In our case,

Dependent varible- percentage of marks,
Independent variable- hours of study,
R2 vaue- 94.549% (approx),
This high value indicates that the data fits into the model very well.