Author: Prateek Soni

GRIP @ The Sparks Foundation

In this regression task I have predicted the percentage of marks that a student is expected to score based upon the number of hours they studied. This is a simple linear regression problem as it has just one predictor.

Step:1 Importing Useful Python Library

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

Step:2 Importing Dataset

In [2]:
```
data=pd.read_csv("data.csv")
print('Importing Data Successfully')
```

```
Importing Data Successfully
```

In [3]:
```
print('First ten data')
data.head(10)
```

```
First ten data
```

Out[3]:

	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

Preparing Data for Machine learning

In [25]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25 entries, 0 to 24
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Hours   25 non-null    float64
 1   Scores  25 non-null    int64  
dtypes: float64(1), int64(1)
memory usage: 528.0 bytes
```

In [4]:
```
#Data cleaning
data.isnull().sum()
```

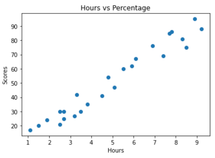
Out[4]:
```
Hours     0
Scores    0
dtype: int64
```

Step:3 Data Visualisation

In [5]:
```
x=np.array(data[['Hours']])
```

In [6]:
```
y=np.array(data[["Scores"]])
```

In [7]:
```
plt.scatter(x,y)
plt.title("Hours vs Percentage")
plt.xlabel('Hours')
plt.ylabel('Scores')
plt.show()
```



In [8]:
```
print('We can see that Scores increses as the no. of hours studied is increase')
print('hence we can conclude that there exist a positive linear relation between the number of hours studied and percentage of score.')
```

```
We can see that Scores increses as the no. of hours studied is increase
hence we can conclude that there exist a positive linear relation between the number of hours studied and percentage of score.
```

Step:4 Train-Test-Split

In [9]:
```
x=data[['Hours']]
y=data[['Scores']]


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)
```

Step:5 Training Algorithm

In [10]:
```
from sklearn.linear_model import LinearRegression
regressor=LinearRegression()

regressor.fit(x_train, y_train)
print('Training Complete')
```

```
Training Complete
```

Step:6 Plotting the Line of Regression

In [11]:
```
regressor.coef_
```

Out[11]:
```
array([[9.53375469]])
```

In [12]:
```
regressor.intercept_
```

Out[12]:
```
array([2.70791426])
```

In [13]:
```
x_test
```

Out[13]:

	Hours
14	1.1
10	7.7
20	2.7
15	8.9
11	5.9

In [21]:
```
y_test
```

Out[21]:

	Scores
14	17
10	85
20	30
15	95
11	62

In [27]:
```
y_pred=regressor.predict(x_test)
y_pred
```

Out[27]:
```
array([[13.19504443],
       [76.11782541],
       [28.44905194],
       [87.55833104],
       [58.95706696]])
```

In [28]:
```
pd.DataFrame(np.c_[x_test,y_test,y_pred], columns=['Study Hours', 'Original Student Marks','Predicted Student Marks'])
```

Out[28]:

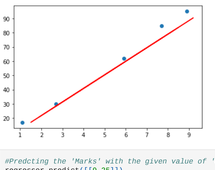
	Study Hours	Original Student Marks	Predicted Student Marks
0	1.1	17.0	13.195044
1	7.7	85.0	76.117825
2	2.7	30.0	28.449052
3	8.9	95.0	87.558331
4	5.9	62.0	58.957067

In [29]:
```
regressor.score(x_test,y_test)
```

Out[29]:
```
0.9649659232530428
```

In [30]:
```
plt.scatter(x_test,y_test)
plt.plot(x_train, regressor.predict(x_train), color='red')
```

Out[30]:
```
[<matplotlib.lines.Line2D at 0x2335737fbe0>]
```



In [34]:
```
#Predicting the 'Marks' with the given value of 'Hours'
regressor.predict([[9.25]])
```

Out[34]:
```
array([[90.89514519]])
```

Step:7 Evaluating the model

In [48]:
```
from sklearn import metrics
print('Mean Absolute Error', metrics.mean_absolute_error(y_test,y_pred))
```

```
Mean Absolute Error 4.944536044700928
```

Conclusion

I have carried out the prediction using Supervised Machine Leaning and evaluated performance of the model. From above analysis, I reached to the conclusion that if a student study for 9.25 then he/she will score 90.89 percentage/marks.

In [ ]:
```

```

In [ ]:
```

```