The spark foundation ¶

Data Science & Business Analytics intern (July-2022)

Author : Priti kolkante

Task 1 :Prediction using supervised ML

In [20]:

```
# Importing all required libraries

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error
```

In [21]:

```
# Reading the data

data='http://bit.ly/w-data'
df=pd.read_csv(data)
df.head()
```

Out[21]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30

In [22]:

```
# Checking for null values
df.isnull().sum()
```

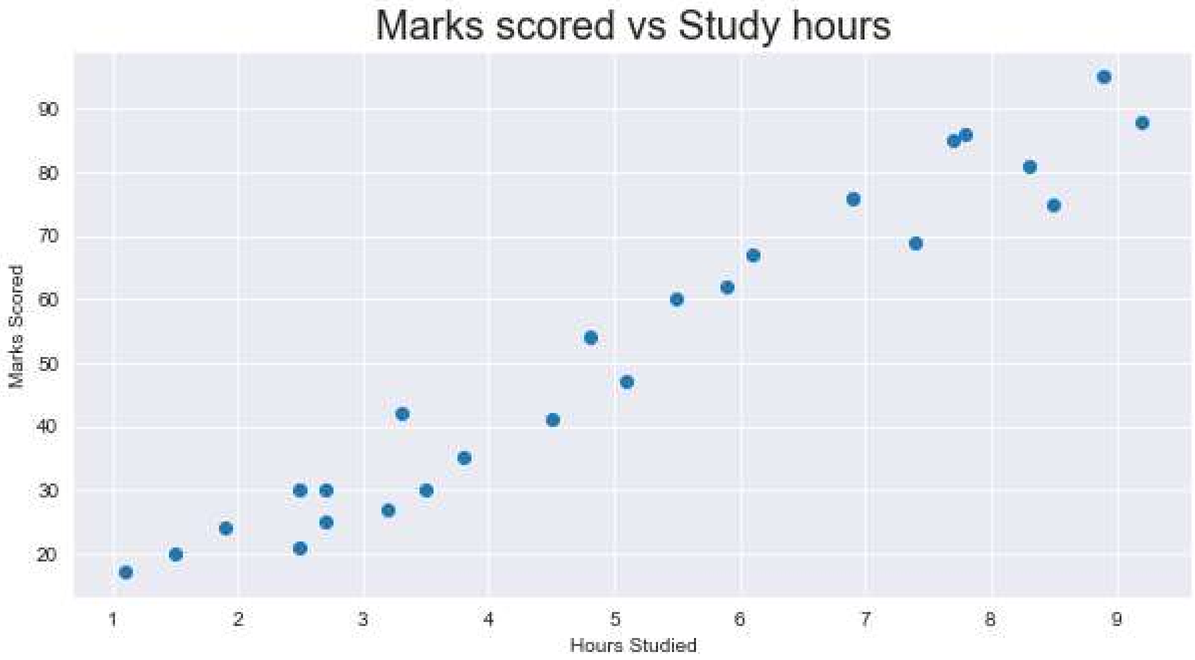
Out[22]:

```
Hours     0
Scores    0
dtype: int64
```

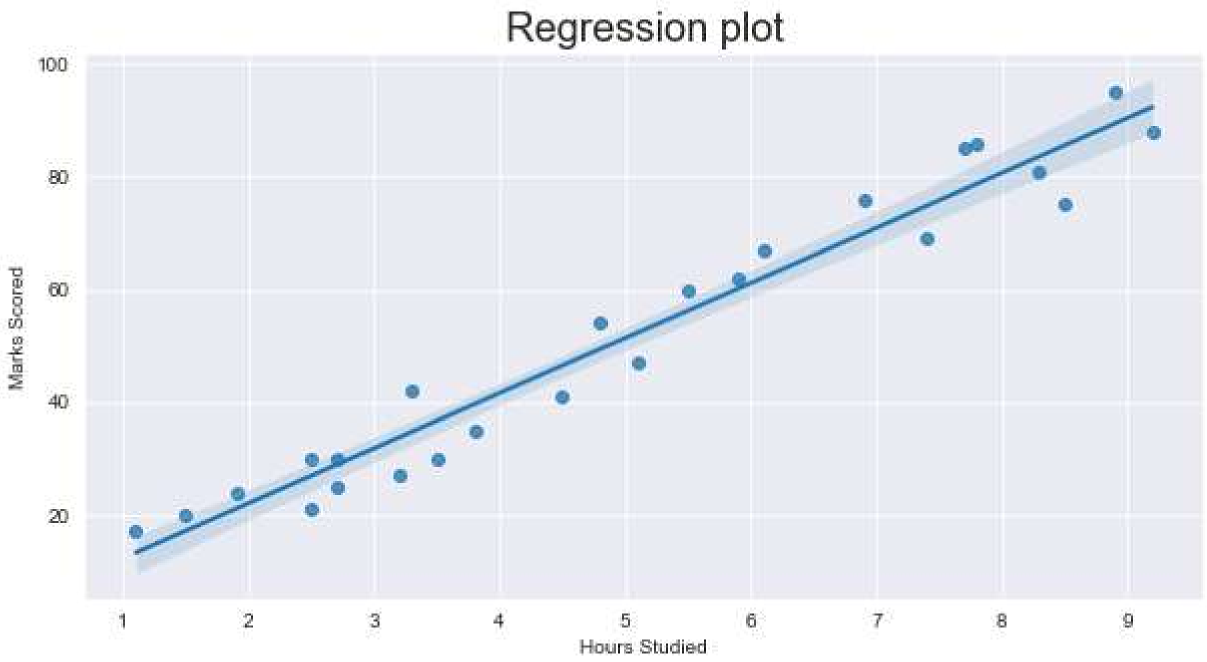
In [23]:

```
plt.figure(figsize=(10,5))
sns.set_style('darkgrid')
plt.scatter(x=df['Hours'],y=df['Scores'])
plt.title('Marks scored vs Study hours',size=20)
plt.xlabel('Hours Studied',size=10)
plt.ylabel('Marks Scored',size=10)

plt.show()
```



In [24]:
```
fig=plt.figure(figsize=(10,5))
sns.regplot(x='Hours',y='Scores',data=df)
plt.title('Regression plot',size=20)
plt.xlabel('Hours Studied',size=10)
plt.ylabel('Marks Scored',size=10)
plt.show()
```



In [25]:
```
df.corr()
```

Out[25]:
```
          Hours     Scores
Hours   1.000000  0.976191
Scores  0.976191  1.000000
```

It's confirmed that the variables are positively correlated.

Training Model

In [26]:
```
# Splitting the data

x=df.drop(columns=['Scores'])
y=df['Scores']
```

In [27]:
```
# spliting it into train and test sets
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.25)
```

In [28]:
```
# Training the model
linear=LinearRegression()
linear.fit(x_train,y_train)
predicitions=linear.predict(x_test)
print('Training completed')
```

```
Training completed
```

In [32]:
```
# Testing Module
predict=linear.predict(x_test)
prediction=pd.DataFrame({'Hours':x_test['Hours'], 'Predictied Marks':predict})
prediction
```

Out[32]:

	Hours	Predictied Marks
2	3.2	33.378209
20	2.7	28.442596
4	3.5	36.339576
13	3.3	34.365331
11	5.9	60.030516
16	2.5	26.468351
5	1.5	16.597126

In [33]:

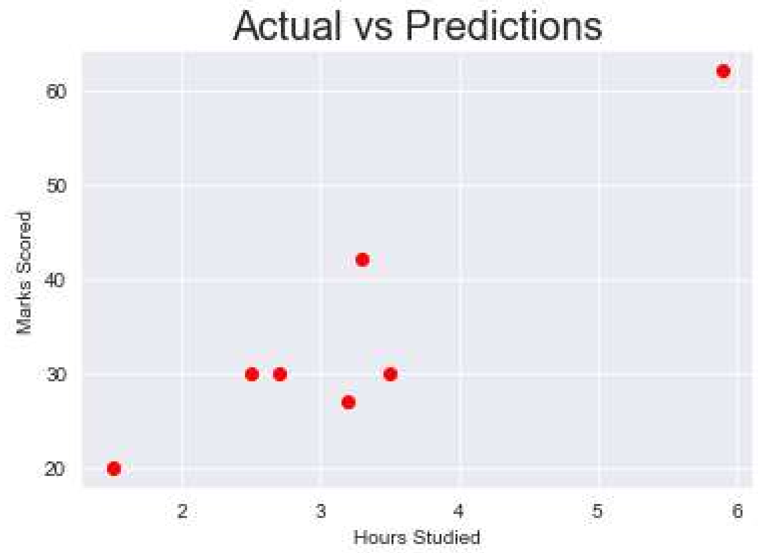
```
# Comparing it with actual marks
compare=pd.DataFrame({'Hours':x_test['Hours'], 'Actual Marks':y_test ,'Predictied Marks':predict})
compare
```

Out[33]:

```
    Hours  Actual Marks  Predictied Marks
 2    3.2            27         33.378209
20    2.7            30         28.442596
 4    3.5            30         36.339576
13    3.3            42         34.365331
11    5.9            62         60.030516
16    2.5            30         26.468351
 5    1.5            20         16.597126
```

In [36]:

```
plt.scatter (x=x_test,y=y_test,color='red')
plt.plot(X=x_test,Y=predict,color='Black')
plt.title('Actual vs Predictions',size=20)
plt.xlabel('Hours Studied',size=10)
plt.ylabel('Marks Scored',size=10)
plt.show()
```



Evaluating Model

In [38]:

```
print('Mean error is:',mean_absolute_error(y_test,predict))
print('Small value of mean absolute error state that the chances of error or incorrect forcasting through the model is ve
```

```
Mean error is: 4.401980529060308
Small value of mean absolute error state that the chances of error or incorrect forcasting through the model is very le
ss
```

Finding predicted score of student who have studied for 9.25 hours/day

In [40]:

```
hou=[9.25]
ans=linear.predict([hou])
print('Score is',ans[0])
print('According to regression model if student have studies for 9.25 hours/day then student likely to score 93.66 marks
```

```
Score is 93.09911932826901
According to regression model if student have studies for 9.25 hours/day then student likely to score 93.66 marks

C:\Users\priya\anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but L
inearRegression was fitted with feature names
  warnings.warn(
```

In [ ]: