TASK 1-PREDICTION USING SUPERVISED ML

Predict the percentage of an student based on the no. of study hours.

Simple Linear Regression

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

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

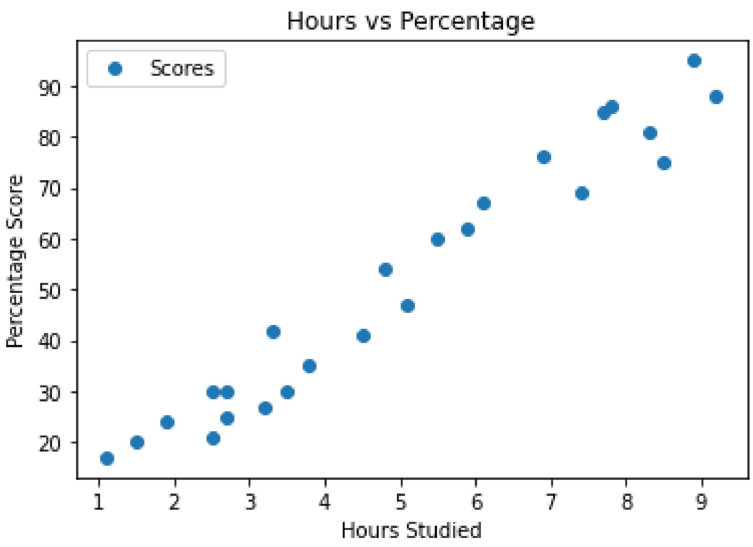
```
url = "http://bit.ly/w-data"
s_data = pd.read_csv(url)
print("Data imported successfully")

s_data.head(10)
```

```
Data imported successfully
```

	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

Visualization

```
#Hours Vs Percentage of Scores
s_data.plot(x='Hours', y='Scores', style='o')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Score')
plt.show()
```



Train-Test Split

```
#X will take all the values except for the last column which is our dependent variable (targe
X = s_data.iloc[:, :-1].values
y = s_data.iloc[:, 1].values
```

```
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)
```

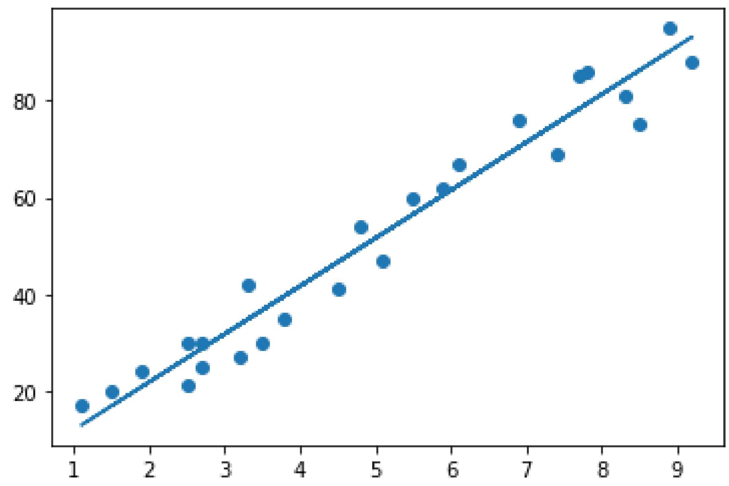
Training the Simple Linear Regression model on the Training set

```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)

print("Training complete.")
```

```
     Training complete.
```

```
# Plotting the regression line
line = regressor.coef_*X+regressor.intercept_
```

```
# Plotting for the test data
plt.scatter(X, y)
plt.plot(X, line);
plt.show()
```



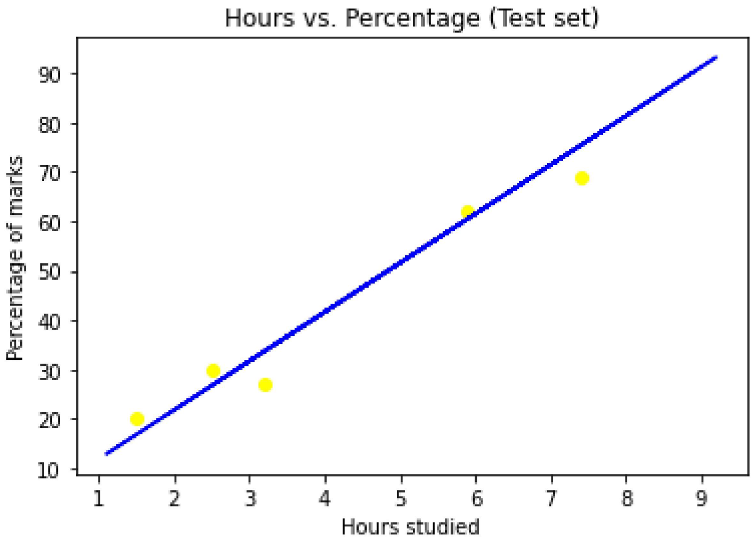
```
#Predicting the Test set results
print(X_test) # Testing data - In Hours
y_pred = regressor.predict(X_test) # Predicting the scores
```

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

```
#Visualising the Training set results
plt.scatter(X_train, y_train, color = 'yellow')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Hours vs. Percentage (Training set)')
plt.xlabel('Hours studied')
plt.ylabel('Percentage of marks')
plt.show()
```

Hours vs. Percentage (Training set)

```
#Visualising the Test set results
plt.scatter(X_test, y_test, color = 'yellow')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Hours vs. Percentage (Test set)')
plt.xlabel('Hours studied')
plt.ylabel('Percentage of marks')
plt.show()
```



```
#Comparing the actual values with the predicted ones.
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
df
```

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

```
#predicting the score
dataset = np.array(9.25)
dataset = dataset.reshape(-1, 1)
pred = regressor.predict(dataset)
print("If the student studies for 9.25 hours/day, the score is {}.".format(pred))
```

```
    If the student studies for 9.25 hours/day, the score is [93.69173249].
```

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

```
Mean Absolute Error: 4.183859899002982
```

```
from sklearn.metrics import r2_score
print("The R-Square of the model is: ",r2_score(y_test,y_pred))
```

```
The R-Square of the model is:  0.9454906892105354
```

Conclusion: We used a Linear Regression Model to predict the score of a student if he/she studies for 9.25 hours/day and the Predicted Score came out to be 92.91.

✓ 0s completed at 8:25 PM ● ×