Prediction using Supervised ML
(Level - Beginner)

AIM : To predict the percentage of an student based on the no. of study hours. This is a simple linear regression task as it involves just 2 variables.What will be predicted score if a student studies for 9.25 hrs/ day? Data can be found at http://bit.ly/w-data

First, let's import the necessary libraries and read the data from the provided link:

In [1]:

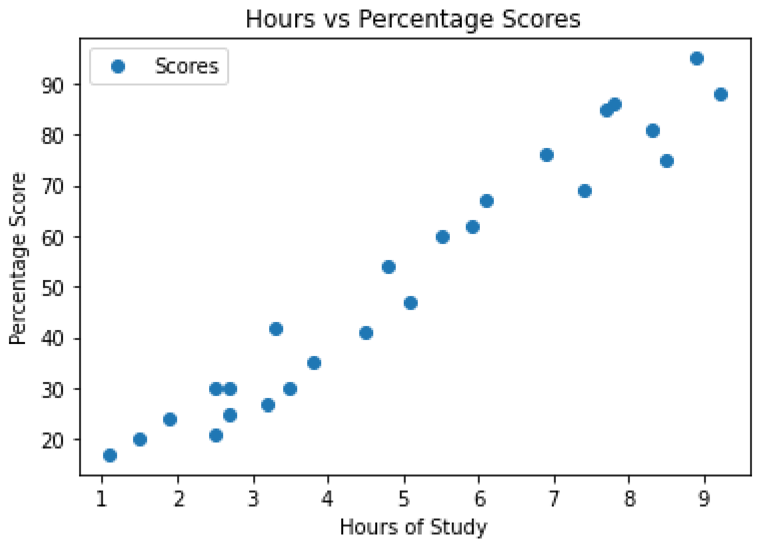
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

url = "http://bit.ly/w-data"
data = pd.read_csv(url)
```

Next, we can visualize the relationship between the number of study hours and the percentage scores using a scatter plot:

In [2]:

```
data.plot(x='Hours', y='Scores', style='o')
plt.title('Hours vs Percentage Scores')
plt.xlabel('Hours of Study')
plt.ylabel('Percentage Score')
plt.show()
```



The resulting scatter plot shows a clear positive linear relationship between the number of study hours and percentage scores.

Next, we can split the data into training and testing sets using the train_test_split() function from the 'sklearn' library:

In [3]:

```
from sklearn.model_selection import train_test_split

X = data.iloc[:, :-1].values
y = data.iloc[:, 1].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)
```

We can then fit a linear regression model to the training data using the LinearRegression() function from sklearn:

In [4]:

```
from sklearn.linear_model import LinearRegression

regressor = LinearRegression()
regressor.fit(X_train, y_train)

print("Training complete.")
```

```
Training complete.
```

Once the model is trained, we can use it to make predictions on the testing data:

In [6]:

```
y_pred = regressor.predict(X_test)
```

Finally, we can compare the actual percentage scores with the predicted scores using a DataFrame:

In [7]:

```
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
print(df)
```

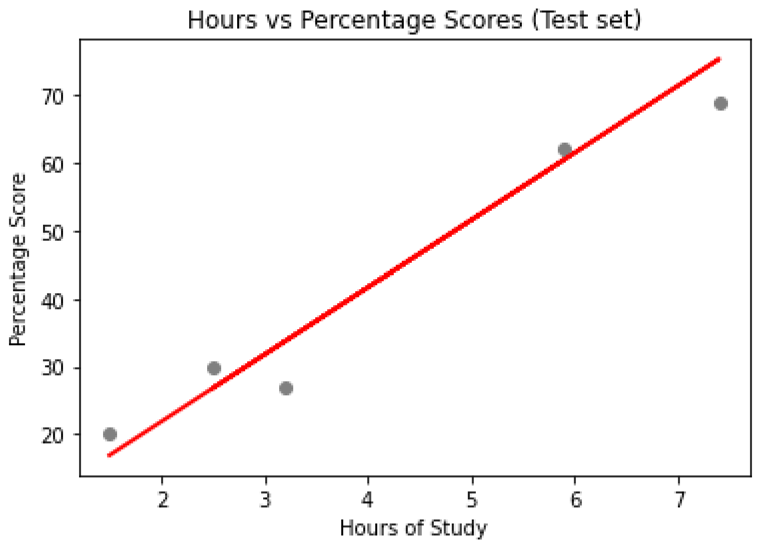
```
   Actual  Predicted
0      20  16.884145
1      27  33.732261
2      69  75.357018
3      30  26.794801
4      62  60.491033
```

This will give us a table of the actual percentage scores and the predicted scores for the testing data.

We can also visualize the predicted values and the regression line on the scatter plot:

In [8]:

```
plt.scatter(X_test, y_test, color='gray')
plt.plot(X_test, y_pred, color='red', linewidth=2)
plt.title('Hours vs Percentage Scores (Test set)')
plt.xlabel('Hours of Study')
plt.ylabel('Percentage Score')
plt.show()
```



The resulting scatter plot with the regression line shows how well the model fits the testing data:

Overall, this is how we can perform simple linear regression to predict the percentage of a student based on the number of study hours.

We can use the linear regression model that we trained earlier to predict the percentage score of a student who studies for 9.25 hours per day

In [9]:

```
hours = [[9.25]]
pred_score = regressor.predict(hours)
print("Predicted Score = {:.2f}%".format(pred_score[0]))
```

```
Predicted Score = 93.69%
```