Dinesh Jain

Task 1

gripseptember2021

Prediction using supervised ml

We need to predict the percentage of the student based on the number of study hours per day.

In [42]:
```
!pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\hp\appdata\local\programs\python\p
ython39\lib\site-packages (1.3.3)
Requirement already satisfied: pytz>=2017.3 in c:\users\hp\appdata\local\programs\py
thon\python39\lib\site-packages (from pandas) (2021.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\hp\appdata\local\p
rograms\python\python39\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: numpy>=1.17.3 in c:\users\hp\appdata\local\programs\p
ython\python39\lib\site-packages (from pandas) (1.21.2)
Requirement already satisfied: six>=1.5 in c:\users\hp\appdata\local\programs\python
\python39\lib\site-packages (from python-dateutil>=2.7.3->pandas) (1.16.0)
```

In [43]:
```
import pandas as pd
```

In [44]:
```
!pip install numpy
```

```
Requirement already satisfied: numpy in c:\users\hp\appdata\local\programs\python\py
thon39\lib\site-packages (1.21.2)
```

In [45]:
```
!pip install matplotlib
```

```
Requirement already satisfied: matplotlib in c:\users\hp\appdata\local\programs\pyth
on\python39\lib\site-packages (3.4.3)
Requirement already satisfied: cycler>=0.10 in c:\users\hp\appdata\local\programs\py
thon\python39\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\hp\appdata\local\progra
ms\python\python39\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: numpy>=1.16 in c:\users\hp\appdata\local\programs\pyt
hon\python39\lib\site-packages (from matplotlib) (1.21.2)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\hp\appdata\local\program
s\python\python39\lib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: pillow>=6.2.0 in c:\users\hp\appdata\local\programs\p
ython\python39\lib\site-packages (from matplotlib) (8.3.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\hp\appdata\local\pro
grams\python\python39\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six in c:\users\hp\appdata\local\programs\python\pyth
on39\lib\site-packages (from cycler>=0.10->matplotlib) (1.16.0)
```


In [46]:
```
import numpy as np
```

```
import matplotlib.pyplot as plt
%matplotlib inline
```

In [47]:

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

```
Data imported
```

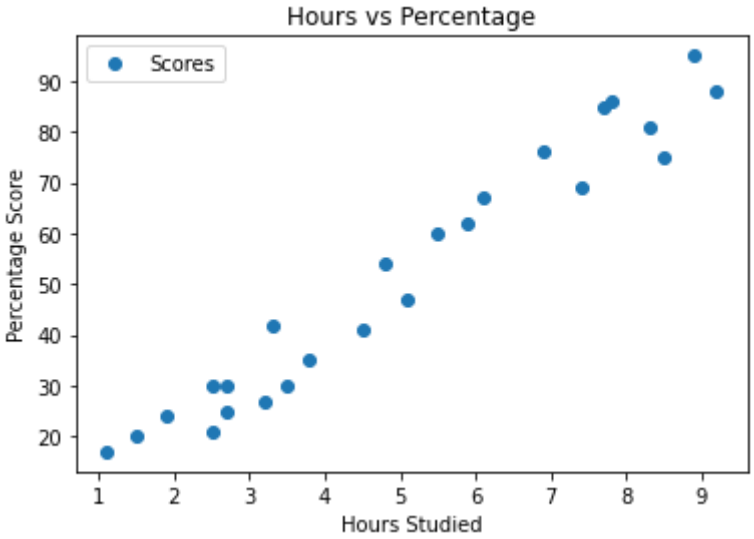
Out[47]:

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

Visualization

In [48]:

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



In [49]:

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

In [50]:

```
!pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in c:\users\hp\appdata\local\programs\py
thon\python39\lib\site-packages (0.24.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\hp\appdata\local\pro
                                                es (from scikit-learn) (2.2.0)
Requirement already satisfied: joblib>=0.11 in c:\users\hp\appdata\local\programs\py
```

```
thon\python39\lib\site-packages (from scikit-learn) (1.0.1)
Requirement already satisfied: scipy>=0.19.1 in c:\users\hp\appdata\local\programs\p
ython\python39\lib\site-packages (from scikit-learn) (1.7.1)
Requirement already satisfied: numpy>=1.13.3 in c:\users\hp\appdata\local\programs\p
ython\python39\lib\site-packages (from scikit-learn) (1.21.2)
```

Training and plotting

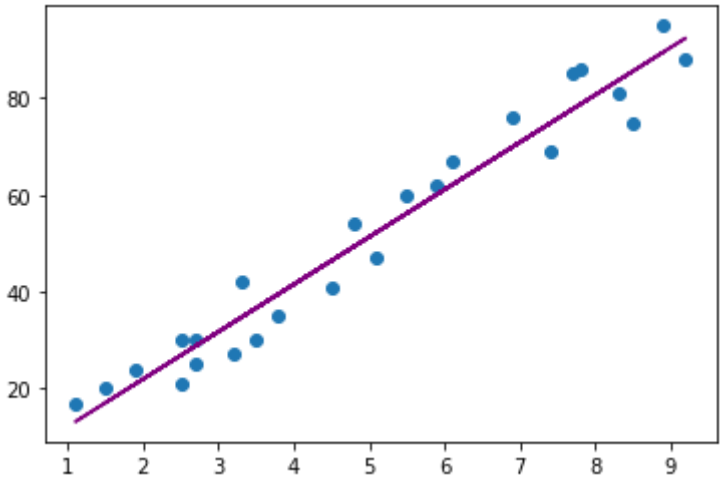
In [51]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
 test_size=0.3, random_state=0)
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
print("Training complete.")
```

```
Training complete.
```

In [52]:

```
# Plotting the regression line
line = regressor.coef_*X+regressor.intercept_
# Plotting for the test data
plt.scatter(X, y)
plt.plot(X, line,color = 'purple');
plt.show()
```



In [53]:

```
y_pred = regressor.predict(X_test)# Predicting the scores
print(y_pred)
```

```
[17.05366541 33.69422878 74.80620886 26.8422321  60.12335883 39.56736879
 20.96909209 78.72163554]
```

In [54]:

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

Out[54]:

	Actual	Predicted
0	20	17.053665
1	27	33.694229
2	69	74.806209

	Actual	Predicted
4	62	60.123359
5	35	39.567369
6	24	20.969092
7	86	78.721636

In [55]:
```
# You can also test with your own data
s_data = np.array(9.25)
s_data = s_data.reshape(-1,1)
own_pred = regressor.predict(s_data)
print("No of Hours = {}".format(9.25))
print("Predicted Score = {}".format(own_pred[0]))
```

```
No of Hours = 9.25
Predicted Score = 92.91505723477056
```

Predicted Score = 92.91505723477056

For 9.25 hours

Hence, we have used the supervised learning to predict the score of the student based on the study hours per day.

Error metrics

In [57]:
```
from sklearn import metrics
print('Mean Absolute Error:',
 metrics.mean_absolute_error(y_test, y_pred))
```

```
Mean Absolute Error: 4.419727808027652
```

Conclusion: We used linear regression model to predict score of student provided study of hours