jupyter Student Score Prediction Spark Foundation proj… Last Checkpoint: 15 hours ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 ○

Utkarsh Tripathi

Student Percentage Prediction by Spark Foundation

Data--> https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/student_scores%20-%20student_scores.csv

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
import missingno
import math
```

In [2]:
```
dt = pd.read_csv('https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/student_scores%20-%2
dt
```

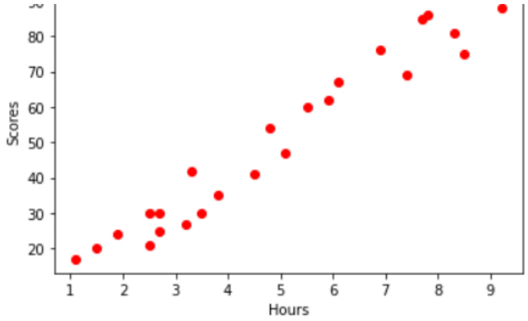
Out[2]:

	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
10	7.7	85
11	5.9	62
12	4.5	41
13	3.3	42
14	1.1	17
15	8.9	95
16	2.5	30
17	1.9	24
18	6.1	67
19	7.4	69
20	2.7	30
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

Data Analysis

In [3]:
```
plt.xlabel('Hours')
plt.ylabel('Scores')
plt.scatter(dt['Hours'], dt['Scores'], color = 'r')

plt.legend()
plt.show()
```

```
No handles with labels found to put in legend.
```



Feature Engineeing

Converting Decimal Values into Data Frame

Assigning X & Y Values

```
In [23]: x = pd.DataFrame(dt['Hours'])
         y = dt['Scores']
```

```
In [24]: from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(x,y, test_size= 0.4)
```

```
In [25]: from sklearn.linear_model import LinearRegression
```

```
In [26]: lr_model = LinearRegression()
```

Training to the model

```
In [36]: lr_model.fit(x_train,y_train)
```

```
Out[36]: LinearRegression()
```

```
In [37]: lr_model.score(x_train, y_train)
```

```
Out[37]: 0.9615149004850574
```

```
In [38]: lr_model.score(x_test, y_test)
```

```
Out[38]: 0.915298979620659
```

```
In [39]: lr_model.predict([[8]])
```

```
Out[39]: array([83.82620009])
```

```
In [40]: lr_model.coef_
```

```
Out[40]: array([10.37685061])
```

```
In [41]: lr_model.intercept_
```

```
Out[41]: 0.8113952445042827
```

```
In [ ]:
```

Exporting Model in File for Model Deployment

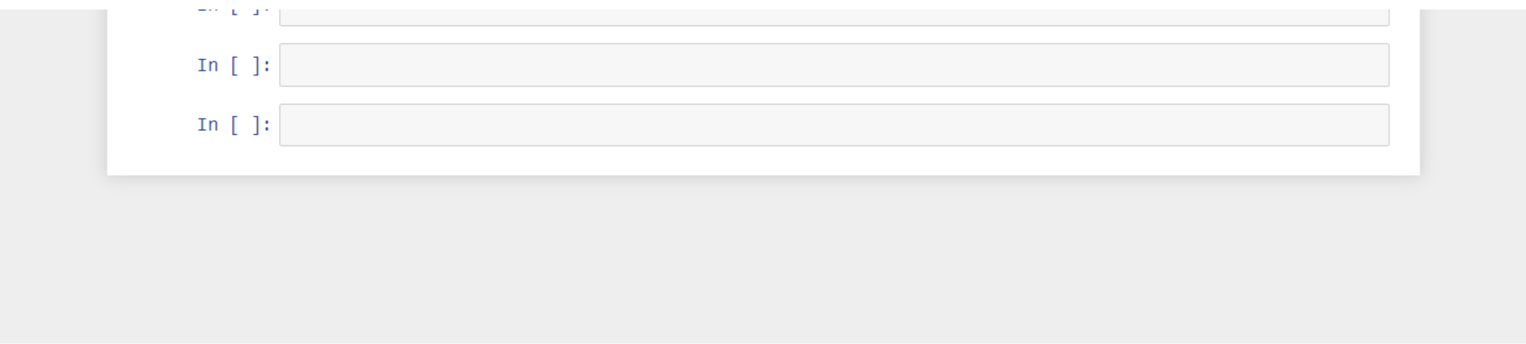
```
In [42]: import pickle

         pickle.dump(lr_model, open('Students Score Prediction', 'wb'))
```

```
In [43]: loaded_model = pickle.load(open('Students Score Prediction', 'rb'))
```

```
In [44]: loaded_model.predict([[7]])
```

```
Out[44]: array([73.44934948])
```

```
In [ ]:
```



```
In [ ]:

In [ ]:

In [ ]:
```