internship-task-1

May 3, 2022

1 internship-task-1

Use the "Run" button to execute the code.

```
[42]: !pip install jovian --upgrade --quiet
```

```
[43]: import jovian
```

```
[44]: # Execute this to save new versions of the notebook
      jovian.commit(project="internship-task-1")
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "parishabhatia12/internship-task-1" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/parishabhatia12/internship-
task-1
```

```
[44]: 'https://jovian.ai/parishabhatia12/internship-task-1'
```

```
[5]: import numpy as np
     import matplotlib.pyplot as plt
     import pandas as pd
     from sklearn import linear_model
```

```
[24]: df = pd.read_csv("https://raw.githubusercontent.com/AdiPersonalWorks/Random/
      →master/student_scores%20-%20student_scores.csv")
```
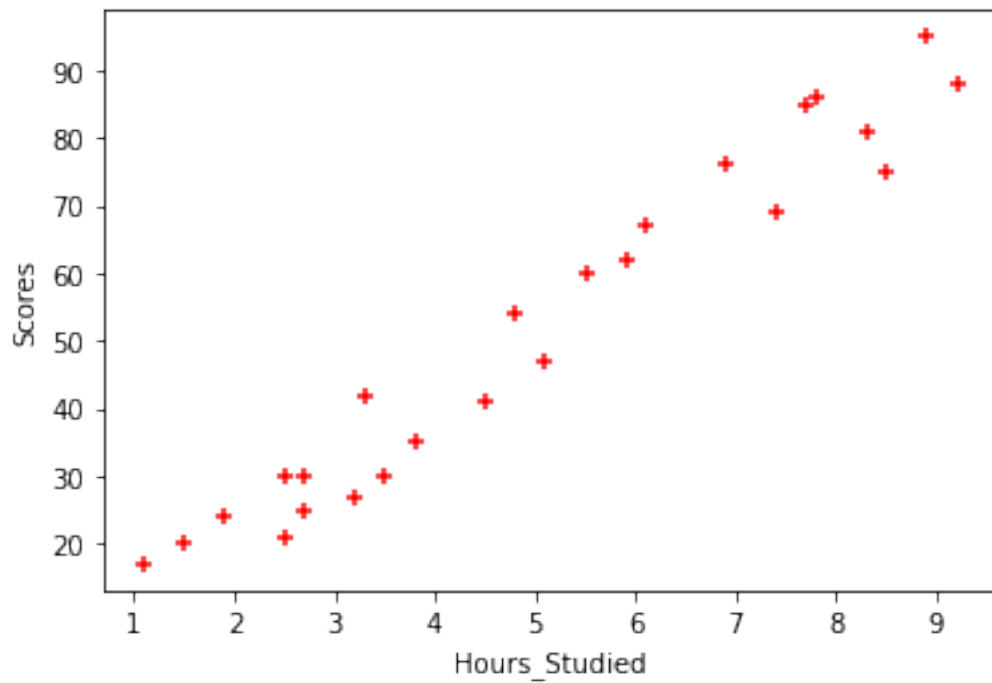
```
[25]: df
```

```
[25]:    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
```

[26]:
```python
%matplotlib inline
plt.scatter(df.Hours,df.Scores,color = 'red',marker = '+')
plt.xlabel('Hours_Studied')
plt.ylabel('Scores')
```

[26]: Text(0, 0.5, 'Scores')

```
[34]: reg = linear_model.LinearRegression(copy_X=True,fit_intercept=True,n_jobs=1)
      reg.fit(df[['Hours']],df.Scores)
```

```
[34]: LinearRegression(n_jobs=1)
```

```
[28]: reg.coef_
```

```
[28]: array([9.77580339])
```

```
[29]: reg.intercept_
```

```
[29]: 2.48367340537321
```

```
[40]: def predict(intercept,coefficient,Hours_Studied):
          Scores = intercept + (coefficient*Hours_Studied)
          return Scores
```

```
[41]: predict(2.48367340537321,9.77580339,9.25)
```

```
[41]: 92.90985476287321
```

```
[ ]:
```