

Spark Intern - Task 1

March 14, 2022

```
[6]: # importing library

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

```
[7]: # Reading data

data = pd.read_csv('https://raw.githubusercontent.com/AdiPersonalWorks/Random/
↳master/student_scores%20-%20student_scores.csv')
```

```
[8]: data.head(10)
```

```
[8]:
```

	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

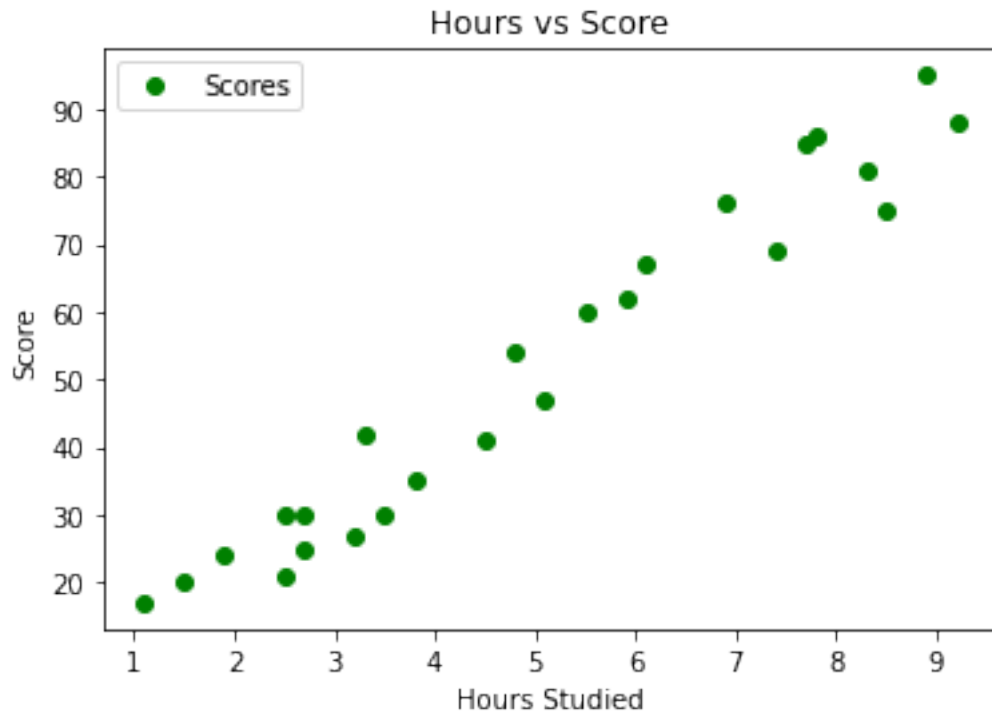
```
[5]: data.isna().sum()
```

```
[5]: Hours      0
      Scores    0
      dtype: int64
```

```
[81]: # plotting dataset

data.plot(x='Hours',y='Scores',style='o',color='green')
plt.grid(color='white')
plt.title('Hours vs Score')
plt.xlabel('Hours Studied')
plt.ylabel('Score')
```

```
plt.show()
```



```
[82]: # Converting array into data's
```

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

```
[83]: from sklearn.model_selection import train_test_split
```

```
[84]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

```
[85]: from sklearn.linear_model import LinearRegression
```

```
regs = LinearRegression()  
regs.fit(X_train, y_train)  
  
print('Training Completed')
```

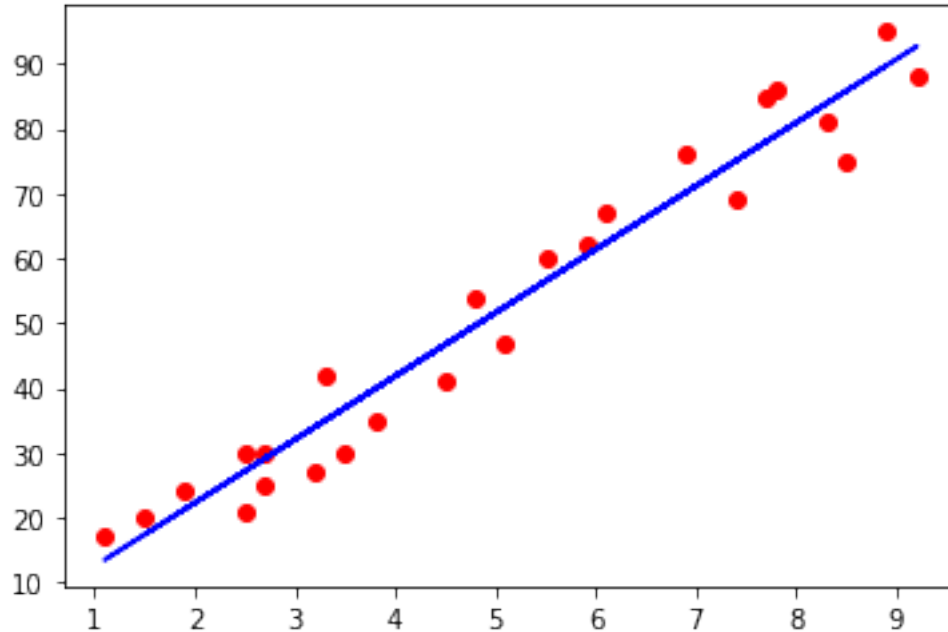
Training Completed

```
[98]: # plotting regression and co-efficient
```

```
line = regs.coef_*X+regs.intercept_
```

```
plt.scatter(X,y,color='red')
plt.plot(X,line,color='b')
```

```
[98]: []
```



```
[94]: # Predicting the scores
```

```
print(X_test)
y_pred = regs.predict(X_test)
```

[[4.8]
[2.7]
[8.3]
[5.9]
[5.1]]

```
[88]: # Comparing Actual vs Predicted
```

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

[88]:	Actual	Predicted
0	54	49.678081
1	25	29.149848
2	81	83.891803
3	62	60.430965

4 47 52.610686

```
[91]: # Prediction on 9.25 hours
```

```
hours = 9.25
own_pred = regs.predict([[hours]])
print("No of Hours = {}".format(hours))
print("Predicted Score = {}".format(own_pred[0]))
```

No of Hours = 9.25

Predicted Score = 93.17838494231938

```
[93]: # Evaluating the model
```

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

Mean Absolute Error: 3.7086581663630853