Day2

January 8, 2019

Note: Considering that the 'nvconvert' module cannot convert notebook with Chinese characters, Notes are gonna written in English from now on

0.1 Step 1:Preorcess the data

- import modules
- import Dataset
- replace missing data with mean or max/min
- split dataset
- feature scaling

```
In [28]: import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         # show figures in current window
         %matplotlib inline

         dataset = pd.read_csv('../datasets/studentscores.csv')
         X = dataset.iloc[:, :1].values
         Y = dataset.iloc[:, 1].values
         print("Original data X: {}, Y:{}".format(X.shape, Y.shape))

         from sklearn.cross_validation import train_test_split
         X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.25, random_stat
         print(X_train.shape)
```

```
Original data X: (25, 1), Y:(25,)
(18, 1)
```

0.2 Step 2: Fitting socre(Y)-hour(X) relation with Linear Regeression

- Y: student score
- X: learning hours for each student

```
In [29]: from sklearn.linear_model import LinearRegression
         model = LinearRegression()
         model = model.fit(X_train, Y_trian)
```

0.3 Step 3: Predict the result

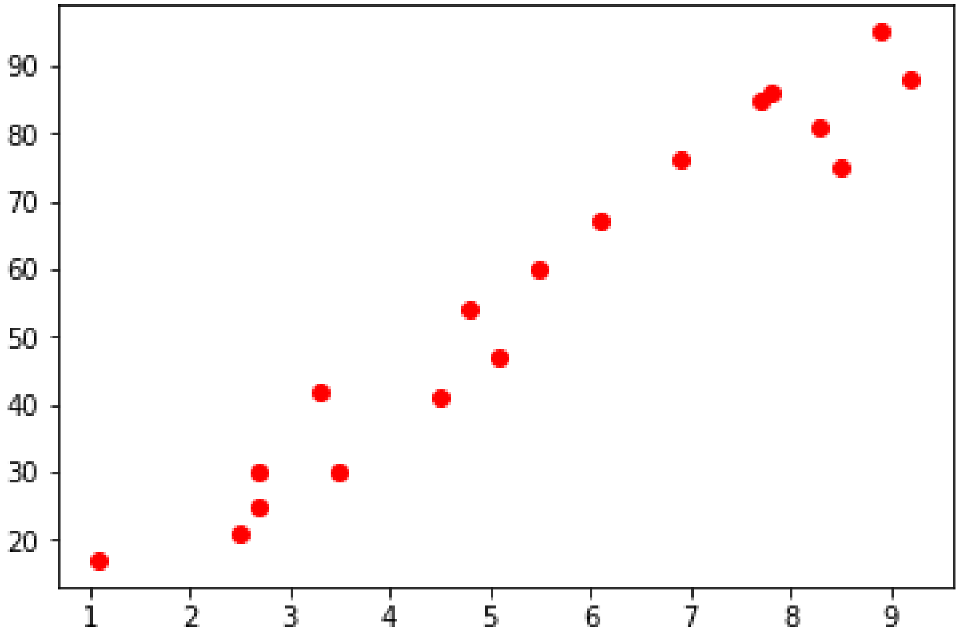
```
In [30]: Y_pred = model.predict(X_test)
```

0.4 Step 4: Visualization

- Visualising the training results

```
In [31]: plt.scatter(X_train, Y_train, color='red')
```

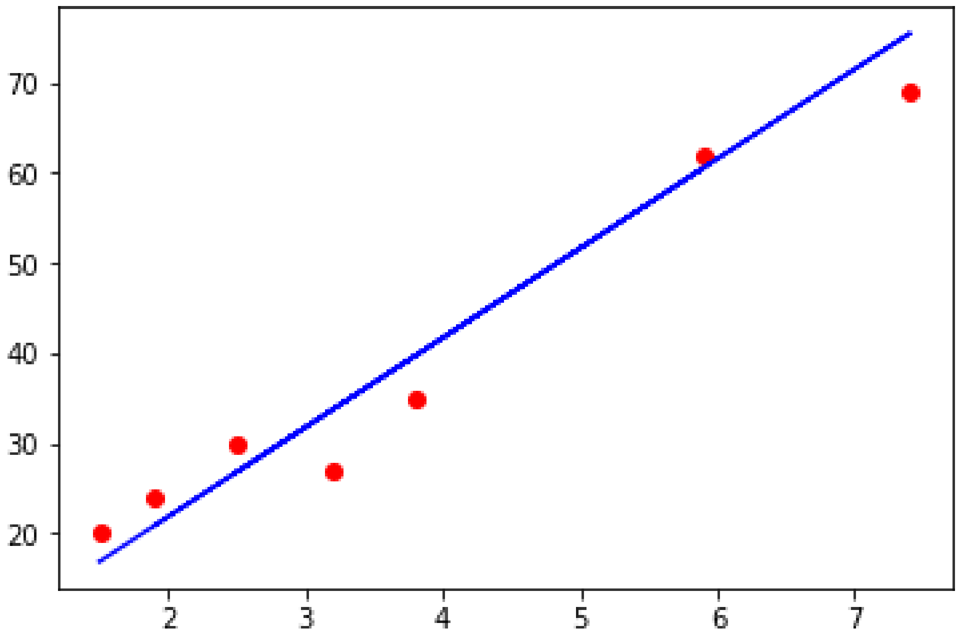
```
Out[31]: <matplotlib.collections.PathCollection at 0x7fbd283022e8>
```



- Visualizing the test results

```
In [32]: plt.scatter(X_test, Y_test, color='red')
         plt.plot(X_test, model.predict(X_test), color='blue')
```

```
Out[32]: [<matplotlib.lines.Line2D at 0x7fbd2825ab00>]
```


70
60
50
40
30
20
2
3
4
5
6
7