

공부시간과 성적의 관계

공부시간	성적	예측값
2	81	83.6
4	93	88.2
6	91	92.8
8	97	97.4

fake_a_b는 임의의 예측값을 넣으면된다

아래는 공식으로 순차적으로 풀기위해

기본에서 계산처리한값들

공식

- 공부시간(x)평균 : $(2 + 4 + 6 + 8)/4 = 5$
- 성적(y)평균 : $(81 + 93 + 91 + 97)/4 = 90.5$

$$1. \text{기울기 } a = \frac{(x-x\text{평균})(y-y\text{평균})\text{의 합}}{(x-x\text{평균})^2\text{의 합}}$$

$$\Rightarrow a = \frac{(2-5)(81-90.5)+(4-5)(93-90.5)+(6-5)(91-90.5)+(8-5)(97-90.5)}{((2-5)^2+(4-5)^2+(6-5)^2+(8-5)^2)}$$

$$\Rightarrow \frac{46}{20}$$

$$\Rightarrow \text{기울기 } a = 2.3$$

$$2. y\text{절편의 값 } b = y\text{의 평균} - (x\text{의 평균} \times \text{기울기 } a)$$

$$\Rightarrow b = 90.5 - (5 \times 2.3)$$

$$\Rightarrow b = 90.5 - 11.5$$

$$\Rightarrow y\text{절편의 값 } b = 79$$

$$3. y = ax + b$$

$$\Rightarrow y = 2.3x + 79$$

$$\Rightarrow y = 2.3 \times 2 + 79 = 83.6$$

$$\Rightarrow y = 2.3 \times 4 + 79 = 88.2$$

$$\Rightarrow y = 2.3 \times 6 + 79 = 92.8$$

$$\Rightarrow y = 2.3 \times 8 + 79 = 97.4$$

4. 오차값

$$\Rightarrow \text{오차} = \text{예측값} - \text{실제값}$$

$$\Rightarrow 83.6 - 81 = 1.0$$

$$\Rightarrow 88.2 - 93 = 25.0$$

$$\Rightarrow 92.8 - 91 = 9.0$$

$$\Rightarrow 97.4 - 97 = 9.0$$

5 평균제곱오차(MSE)

$$\Rightarrow \sum_i^n (y_i - \hat{y}_i)^2$$

$$\Rightarrow (83.6 - 81)^2 = 6.7599999999999705$$

$$\Rightarrow (88.2 - 93)^2 = 23.039999999999974$$

$$\Rightarrow (92.8 - 91)^2 = 3.2399999999999896$$

$$\Rightarrow (97.4 - 97)^2 = 0.16000000000000456$$

$$\Rightarrow \frac{1}{n} \sum (y_i - \hat{y}_i)^2$$

$$(6.7599999999999705 + 23.039999999999974 + 3.2399999999999896 + 0.16000000000000456)/4$$

$$\text{평균제곱오차(MSE)} = 8.299999999999985$$

In [1]:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 fake_a_b = [2.3, 79.0]
5
6 data = [[2, 81], [4, 93], [6, 91], [8, 97]]
7
8 ## i[0]은 첫번째값(a), i[1]은 두번째값(y)
9 x = [i[0] for i in data]
10 y = [i[1] for i in data]
11
12 print("x => ", x)
13 print("y => ", y)
```

x => [2, 4, 6, 8]
y => [81, 93, 91, 97]

In [2]:

```
1 ## y = ax + b 에 대한 결과 처리 함수, EX) 예측값A * X값 + 예측값B = 3 * 2 + 76
2 def predict(x):
3     return (fake_a_b[0] * x) + fake_a_b[1]
```

In [3]:

```
1 ## MES 평균제곱오차, EX) Y값 점수 - 예측값, 81 - 82 = 1,0 = 1,0 * 1,0 = 1.0
2 def mse(y, y_hat):
3     return np.mean(((y - y_hat) ** 2))
```

In [4]:

```
1 # MSE 평균오차 값
2 def mse_val(y, preidct_res):
3     return mse(np.array(y), np.array(predict_res))
```

In [5]:

```

1 # 예측값 배열
2 predict_res = []
3
4 for i in range(len(x)):
5     predict_res.append(predict(x[i]))
6     print("시간 : ", x[i],
7           " 성적 : ", y[i],
8           " 예측값 : ", predict_res[i],
9           " 오차값 : ", mse(y[i], predict_res[i])
10         )

```

```

시간 : 2 성적 : 81 예측값 : 83.6 오차값 : 6.7599999999999705
시간 : 4 성적 : 93 예측값 : 88.2 오차값 : 23.039999999999974
시간 : 6 성적 : 91 예측값 : 92.8 오차값 : 3.2399999999999896
시간 : 8 성적 : 97 예측값 : 97.4 오차값 : 0.16000000000000456

```

In [6]:

```
1 print("MSE 값 : ", mse_val(y, predict_res))
```

```
MSE 값 : 8.2999999999999985
```

In [7]:

```

1 # 그래프 처리
2 pl.plot(x, y, label='BASE')
3 pl.scatter(x, y)
4
5 pl.plot(x, predict_res)
6 pl.scatter(x, predict_res)
7
8 pl.legend()
9 pl.show()

```

