

HW #13

2017029970 우원진

#코드

```
import random
import numpy as np

point = []
for i in range(-5, 7):
    x = i
    y = 2 * x - 1 + np.random.normal(0, 2)
    point.append((x, y))
```

주어진 x값들에 먼저 $y = 2x - 1 + N(0, 2)$ 을 계산하여 sample을 완성했습니다.

```
def best_fitting():
    rand_point = []
    while len(rand_point) < 6:
        idx = random.randrange(0, 12)
        if idx not in rand_point:
            rand_point.append(idx)
    matrix = []
    y_vector = []
    for i in rand_point:
        x, y = point[i]
        matrix.append([x, 1])
        y_vector.append([y])
    A = np.matrix(matrix)
    B = np.matrix(y_vector)

    X = np.linalg.inv(A.transpose() * A) * A.transpose() * B
    X = np.asarray(X)
    a = X[0][0]
    b = X[1][0]

    error = 0
    for i in rand_point:
        x, y = point[i]
        y_best = a * x + b
        error += (y_best - y) * (y_best - y)
    return (a, b, error)

best = (999999, 99999, 999999999)
for i in range(10000):
    (a, b, error) = best_fitting()
    if error < best[2]:
        best = (a, b, error)

print("finding (a, b, error) for 6 samples about minimum error : ", best)
```

그리고 위의 사진과 같이 12개의 sample들 중 6개의 sample을 무작위로 뽑아서 그 sample들로 부터 Least Square Error 를 통해 구한 a, b 값에 대한 Error들을 계산했고, 그 Error들 중 가장 작은 값에 해당하는 a, b를 구했습니다. Sample을 12개중 6개를 뽑을 때는 $12C_6$ 이므로 총 924가지의 경우의 수가 있습니다. 그래서 중복 된다는 것도 생각을 해 총 10000번을 돌리면 웬만한 경우의 수는 다 해볼 수 있을 것이라 생각해서 10000번을 돌렸습니다.

```

# 샘플 모두를 사용한 least_square
mat_all = []
y_all = []
for (x, y) in point:
    mat_all.append([x, 1])
    y_all.append([y])
mat_all = np.matrix(mat_all)
y_all = np.matrix(y_all)

X_all = np.linalg.inv(mat_all.transpose() * mat_all) * mat_all.transpose() * y_all
X_all = np.asarray(X_all)
a_all = X_all[0][0]
b_all = X_all[1][0]

error_all = 0
for (x, y) in point:
    y_all_best = a_all * x + b_all
    error_all += (y_all_best - y) * (y_all_best - y)
print("Least Square Error for all Samples(12) (a,b,c) : ", a_all, b_all, error_all)

```

그리고 12개의 sample 모두에 대한 Least Square Error를 통해 a, b, error를 얻는 코드입니다.

#결과

```

uwonjin-ui-MacBookPro:HW13 woowonjin$ python3 hw13.py
finding (a, b, error) for 6 samples about minimum error : (2.1041677559112255, -0.4135398327318991, 0.24637803159346686)
Least Square Error for all Samples(12) (a,b,c) : 1.9538761177613513 -0.80176200127316 26.72290666856227

```