

HW#4 <Curve fitting using Least Square>

2016024766 김서현

0. 주어진 좌표와 랜덤하게 좌표 set을 뽑는 부분

```
data = np.array([[-2.9, 35.4],
                 [-2.1, 19.7],
                 [-0.9, 5.7],
                 [1.1, 2.1],
                 [0.1, 1.2],
                 [1.9, 8.7],
                 [3.1, 25.7],
                 [4.0, 41.5]])

data1 = random.sample([0, 1, 2, 3, 4, 5, 6, 7], 6) #random하게 점 6개를 뽑음
data2 = random.sample([0, 1, 2, 3, 4, 5, 6, 7], 6)
```

주어진 좌표들을 numpy를 이용해 저장했고, random.sample() 함수를 이용해서 8개의 좌표 중 6개 좌표의 인덱스를 뽑았습니다.

1. 행렬 A를 구하고 pseudo-inverse를 통해 Least Square를 하는 부분

```
A1 = np.ones((6, 3))
idx = 0
for i in data1:
    A1[idx, 0] = data[i, 0]**2
    A1[idx, 1] = data[i, 0]
    A1[idx, 2] = data[i, 0]
    idx += 1
X1 = np.dot(np.linalg.pinv(A1), data[data1:], 1))

A2 = np.ones((6, 3))
idx = 0
for i in data2:
    A2[idx, 0] = data[i, 0]**2
    A2[idx, 1] = data[i, 0]
    A2[idx, 2] = data[i, 0]
    idx += 1
X2 = np.dot(np.linalg.pinv(A2), data[data2:], 1))
```

랜덤하게 뽑힌 인덱스에 맞게 좌표들을 구해 행렬 A를 만들었습니다. $y=ax^2+bx+c$ 의 형태로 fitting을 하기 때문에 위와 같이 A를 구성했습니다. 행렬 A에 대해서 pseudo-inverse를 구하고, 그 행렬과 y좌표값들을 행렬곱연산을 해서 a, b, c의 값들을 추정했습니다. 랜덤하게 좌표 set을 두 번 뽑았기 때문에 이 과정도 두 번 하였습니다.

2. 그래프로 그려보는 부분

```
def f1(a):
    return X1[0]*a*a+X1[1]*a+X1[2]

def f2(a):
    return X2[0]*a*a+X2[1]*a+X2[2]

x1 = np.arange(-10, 10, 0.01)
x2 = np.arange(-10, 10, 0.01)

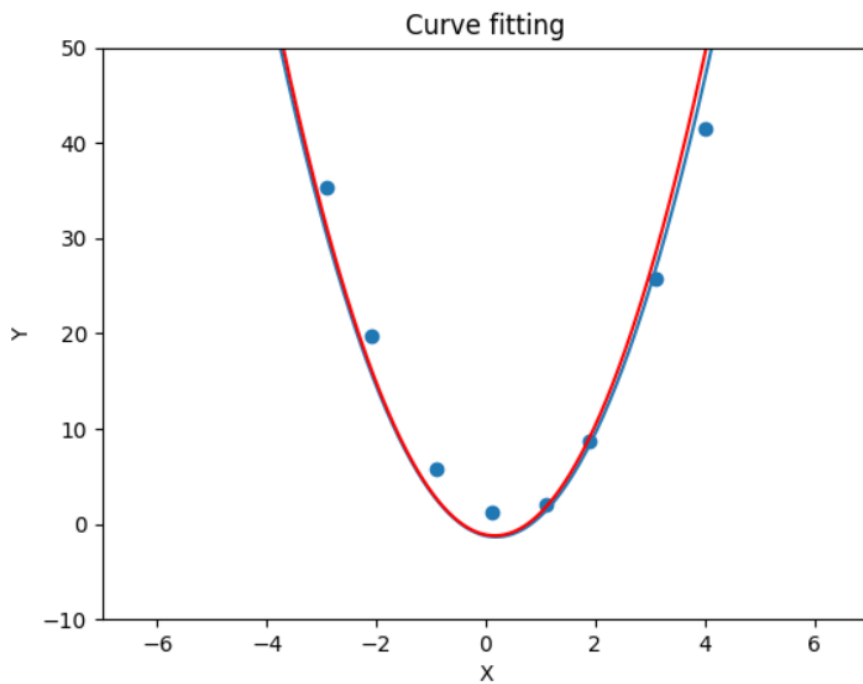
plt.scatter(data[:, 0], data[:, 1])      #그래프를 그림
plt.title("Curve fitting")
plt.xlabel("X")
plt.ylabel("Y")
plt.axis([-7, 7, -10, 50])
plt.plot(x1, f1(x1), x2, f2(x2), 'r-')
plt.show()
```

위의 코드를 통해 두 그래프를 그려보았습니다.

3. 결과 및 분석

```
data1: [6, 7, 0, 5, 2, 1]
data2: [1, 6, 0, 5, 2, 3]
A1: [[ 9.61  3.1  3.1 ]
      [16.   4.   4. ]
      [ 8.41 -2.9 -2.9 ]
      [ 3.61  1.9  1.9 ]
      [ 0.81 -0.9 -0.9 ]
      [ 4.41 -2.1 -2.1 ]]
A2: [[ 4.41 -2.1 -2.1 ]
      [ 9.61  3.1  3.1 ]
      [ 8.41 -2.9 -2.9 ]
      [ 3.61  1.9  1.9 ]
      [ 0.81 -0.9 -0.9 ]
      [ 1.21  1.1  1.1 ]]
X1: [ 3.30766462 -1.23876595 -1.23876595]
X2: [ 3.43089412 -1.12273843 -1.12273843]
```

실행할 때마다 랜덤하게 뽑히는 data set이 달라져서 값이 조금씩 변했습니다. 위 결과가 나왔을 때에는 $y=ax^2+bx+c$ 의 a값은 약 0.13, b값은 약 0.11, c값은 약 0.11정도 차이를 보였습니다.



그래프를 그려보았을 때에는 두 그래프가 약간 차이가 나긴 하지만 거의 동일하게 그려지는 것을 확인할 수 있었습니다. 여러 번 랜덤하게 뽑아서 그래프를 그려보았는데 시각적으로 완전히 겹치게 그려지는 경우도 있었고, 위의 그래프보다 조금 더 차이나는 경우도 있었습니다. 좌표 set이 편차가 클수록 curve fitting의 결과도 많이 차이나는 것을 확인할 수 있었습니다.