

# Pair Programing

4주차

1조  
201984014  
성도윤  
201904126  
허준혁

## Q1. 선형 회귀 코드 분석

A1. numpy패키지를 추가하고 x에 0과 2사이의 무작위 100개의 데이터 포인트를 가진 2차원 배열을 생성한다.

-> y는 x와 선형 관계를 맺는다

->  $4 + 3 * X$

-> 무작위 오차를 추가해서 값에 더해줌으로써 노이즈를 시뮬레이션 한다

## Q2. LinearRegression 클래스가 어떤것인가?

A2. 선형 회귀 모델을 구현하기 위한 클래스

-> 통계 및 머신러닝에서 사용되는 간단한 회귀 분석 기법

-> 모델 초기화, 모델 학습, 예측, 평가, 모델 저장 및 불러오기 등의 기능들을 제공한다

## Q3. 학습률 eta값을 변경하면 어떻게 될까?

A3. 학습 속도가 빨라질 것 같다

-> 이유는???

-> 더 큰 업데이트가 발생하기 때문

## Q4. Interactions값을 100만번으로 올리면 어떻게 될까?

A4. 결과 출력까지 6.9초가 걸린다...

-> 1억번을 하면?

-> 10분이 넘어간다...

-> 적절한 값을 찾는게 중요한 것 같다

## 문제\_1. 확률적 경사 하강법 실습

실습\_1. n\_epochs만큼 for문을 돌린다 (학습률)

-> 학습 데이터 샘플 수 만큼 for문 작성

: for epoch in range(n\_epochs)

-> 첫번째 epoch에서 처음 20개의 데이터 포인트에 대해서만 학습 과정을 시각화하기위한 코드 작성

: for epoch == 0 and i < 20

-> 무작위 포인터를 지정한다

: np.random.randint(m)

-> 경사를 계산한다

-> 평균 제곱 오차를 최소화하기 위한 공식 사용

:  $2 * x_i.T.dot(x_i.dot(theta) * y_i)$

-> 모델 파라미터를 경사률과 학습률을 곱하여 업데이트

:  $theta = theta - eta * gradients$

-> 출력하려면?

: plt.show()

Q5. epoch = 100, dataset = 1000, minibat = 50  
일 때, 가중치 업데이트는 몇 번 변경될까?

A5. epoch마다 1000개의 데이터를 minibat으로 나누  
면 20번의 minibat 업데이트가 발생한다

-> epoch 1회당 가중치 업데이트 횟수는 20회

-> epoch가 100번 진행된다면 총 2000번 발생

Q6. 가중치 역할 규제 선형 회귀 모델

-> 릿지회귀

: 모델의 복잡성을 줄이고 과적합을 방지

-> 라쏘회귀

: 특성 선택의 역할, 불필요한 특성 제거

-> 엘라스틱넷

: 릿지회귀와 라쏘회귀의 장점을 결합