

4주라 기계학승

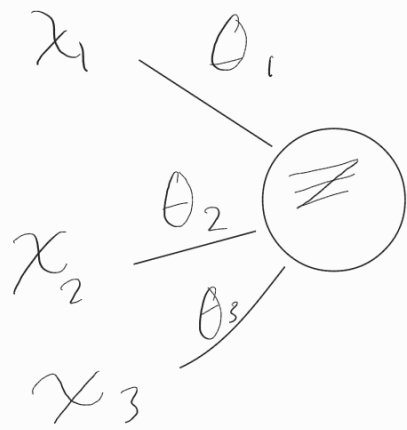
202184026

서동현

201904025

김성호

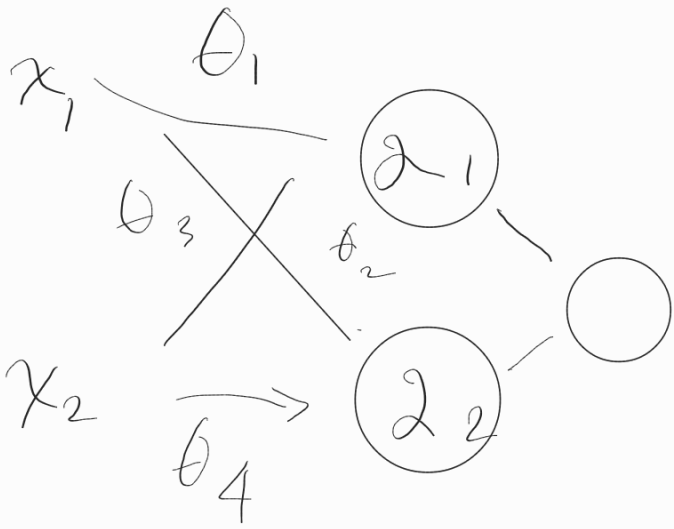
선형 회귀 모델 예제



$$Z = \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

$$= \sum_{i=0}^n \theta_i x_i$$

$$= \theta x$$



$$z_1 = \theta_1 x_1 + \theta_3 x_2$$

$$z_2 = \theta_2 x_1 + \theta_4 x_2$$

$$\begin{bmatrix} \theta_1 & \theta_3 \\ \theta_2 & \theta_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & -2 \end{bmatrix}$$

$$\downarrow$$

$$\begin{bmatrix} 2 & 4 \\ -3 & 7 \end{bmatrix}$$

⋮

① 선형 회귀 모델은 어디에 사용 되는지? <서동현>
<김성욱>

머신 러닝에서 큰 목적은 실제 데이터를 바탕으로 모델을 생성해서 만약 다른 입력 값을 넣었을 때 발생할 결과값을 예측하는 데에 있는데 가장 직관적이고 간단한 모델인 선을 찾는 분석하는 방법을 선형 회귀라고 해 따라서 입력값과 결과값에 대한 관계를 선으로 분석하고 싶을 때 사용하는 거 같다.

② 배치 경사 하강법에서 굳이 전체 데이터를 통해 훈련을 해야하는가? <서동현>

<김성욱>

비용 함수를 미분해서 미분값의 변화를 관찰하는데 전체 데이터가 아닌 일부 데이터만 훈련시키면 미분할 수 있는 양이 적어 미분값의 변화를 관찰하기 어려워서 그런거 같다.

③ 확률적 경사 하강법에서 불인정 함수
<시동형> 어떠한 방법으로 인경화 시킬 수 있을까?
<김성욱>
잘 모르겠다.

서로 코딩하는 실력이

부족해서 이해하는
시간을 가졌습니다.

//코랩

n_epochs = 50 # epoch를 50으로 잡았다.
t0, t1 = 5, 50 # 학습 스케줄 하이퍼파라미터.
m = 100 # 학습 데이터는 100개 (???)

학습 스케줄이 있어야한다 why? 확률적 경사이기 때문에 샘플을 확률적으로 랜덤으로 뽑기 위해 필요하다.

```
def learning_schedule(t):  
    return t0 / (t + t1)
```

theta를 random.randn --> 표준 정규분포 확률을 따르는 난수를 return
theta = np.random.randn(2,1)

#for ~~ in range(~~~) : for 반복문 사용

for epoch in range(n_epochs):

for i in range(m):

if epoch == 0 and i < 20: # ?????

y_predict = X_new_b.dot(theta) # 행렬의 곱(X_new_b에 theta 곱)

style = "b-" if i > 0 else "r--" # i>0 "b-"파랑 색, 아니면 "r--" 빨간 색

plt.plot(X_new, y_predict, style) # X축 :X_new Y축 : y_predict을 style로

plot 그려라

random_index = np.random.randint(m) # random.randint : 0~m-1 사이의
랜덤한 정수 1개

xi = X_b[random_index:random_index+1] # xi는 X_b배열에서 random_index
의 값

yi = y[random_index:random_index+1] # yi는 y배열에서 random_index의
값

gradients = 2 * xi.T.dot(xi.dot(theta) - yi) # .dot, xi, yi, theta를 통해서 기울기를 구한다.

eta = learning_schedule(epoch * m + i) # eta를 learning_schedule함수를 통해서 구한다. 매개변수는 epoch * m + 1이다.

theta = theta - eta * gradients # theta값을 기울기와 eta를 통해서 다시 구한다.

theta_path_sgd.append(theta) #.append를 통해서 theta_path_sgd에
theta값을 추가한다.

문제

DataSet = 1000

epoch = 100

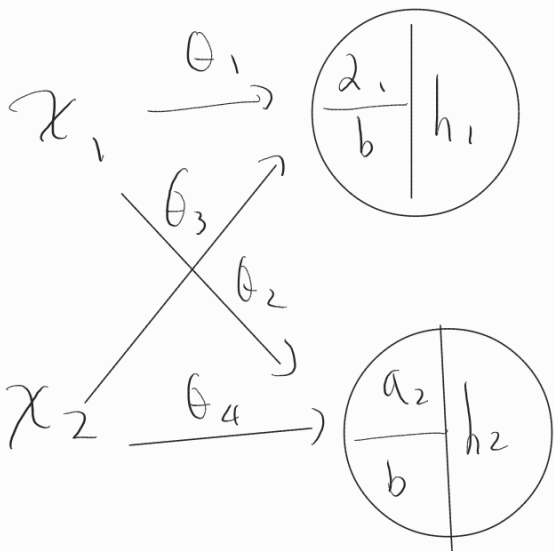
minibatch = 50

기억: DataSet을 50개로 쪼개면
(mini batch)

20개 값이 나오는 데 거기서
Epoch 1000번씩 돌리면
2000번이 나온다.

시행: DataSet이 1000개에서 100번 수행하고
미세배치 크기가 50이니 100×50
5000 번이라고 생각하자.

2지스틱 회귀



시행

$$z_1 = \theta_1 x_1 + \theta_3 x_2$$

$\sigma(z_1)$

$$z_2 = \theta_2 x_1 + \theta_4 x_2$$