import tensorflow as tf

tf.enable\_eager\_execution()

# Data

x\_data = [1, 2, 3, 4, 5]

y\_data = [1, 2, 3, 4, 5]

#데이터 준비 x\_data=input, y\_data=output

# W, b initialize

W = tf.Variable(2.9)

b = tf.Variable(0.5)

# variable로 w,b값정의(임의의 값)

learning\_rate = 0.01

for i in range(100+1): # W, b update

#i는 0부터 101까지 변해 감 101번 update

# Gradient descent/경사하강 알고리즘(W,b를 찾는 알고리즘)-

with tf.GradientTape() as tape:

hypothesis = W \* x\_data + b

cost = tf.reduce\_mean(tf.square(hypothesis - y\_data))

#변수들에 대한 정보를 테이프에 기록후

W\_grad, b\_grad = tape.gradient(cost, [W, b])

# gradient는 cost, [W, b]함수에 대해서 변수들에 대한 개별 미분값을 구하여 반환함

W.assign\_sub(learning\_rate \* W\_grad)

b.assign\_sub(learning\_rate \* b\_grad)

#assign sub를 사용하여 w와 b를 업데이트 함 /뺀값을 다시 그 값에 넣어줌(a-=b)

#learnin rate는 grad를 얼마만큼 반영할지를 결정함

if i % 10 == 0:

print("{:5}|{:10.4f}|{:10.4}|{:10.6f}".format(i, W.numpy(), b.numpy(), cost))

w와 b,cost값이 얼마만큼 변하는지 중간에 확인(i값이 10의 배수가 될 때 마다)