

(1) import	<code>import tensorflow.compat.v1 as tf</code>
(2) training data set	<p>trainging할 데이터 확보 및 설정</p> <pre>x_data = [1, 2, 3], y_data = [1, 2, 3]</pre> <p>그래프를 실행시키는 단계에서 값을 던져준다.</p>
(3) placeholder	<pre>x = tf.placeholder(dtype=float32) y = tf.placeholder(dtype=float32)</pre> <p>가중치(Weight)와 출력값(bias) 설정</p>
(3) Weight & bias	<pre>W = tf.Variable(tf.random_normal([1]), name="weight") # 랜덤으로 값 불러오기 b = tf.Variable(tf.random_normal([1]), name="bias")</pre> <p>가설 설정 (training data set에 잘 맞는 직선을 긋는 과정)</p>
(4) Hypothesis	$H = W * x + b$ <p>가설과 training data 간의 값의 차를 계산하는 함수</p> <p>=&gt; cost funtction의 값을 최소로 만드는 가중치와 출력값을 찾는 것이 목표</p>
(5) cost function	<pre>cost = tf.reduce_mean(tf.square(H - y)) # H-y의 제곱의 평균 optimizer = tf.train.GradientDescentOptimizer(learning_rate=0.01) train = optimizer.minimize(cost)</pre>
(6) Session 실행 & Variable 초기화	<pre>sess = tf.Session() # 실행 session.run(tf.gloval_variables_initializer())</pre> <p>for step in range(1, 3000):</p>
(7) 학습	<pre>_, w_val, b_val, cost_val = sess.run([train, W, b, cost],                                      feed_dict={x:x_data, y:y_data}) # feed dict = 값 불러오기</pre>