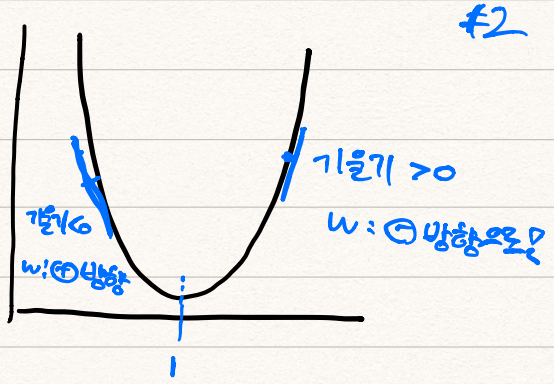
※ Simplified hypothesis

$$H(x) = Wx$$

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx_i - y_i)^2$$



#2

$$W := W - \alpha \frac{1}{m}\sum_{i=1}^{m}(Wx_i - y_i)x_i$$

learning_rate = 0.1

gradient = tf.reduce_mean((W*X-Y)*X)

descent = W - learning_rate * gradient

update = W.assign(descent)

tensorflow는 바로 대입연산자 X.

assign을 써줘야해.

tensorflow를 사용한다면, 일일히 미분할 필요 X. #3

```
optimizer = tf.train.GradientDescentOptimizer(learning_rate=0.1)
train = optimizer.minimize(cost)
```