

머신러닝이 무엇인가?

- 프로그램인데 학습하는 프로그램

학습 종류

- 지도 학습 (Supervised learning)
- 비 지도 학습 (Unsupervised learning)

지도학습 Supervised Learning

- 이름표 붙혀 주고 학습시킴
- training data Set에 따른 종류
 1. regression (회귀): 숫자와의 상관관계
ex) 공부시간과 시험 점수
 1. binary classification (이항 구분): 두가지로
ex) 공부시간과 패스 / 논패스
 1. multi label classification (여러개 구분): 숫자 아닌 것들
ex) 공부 시간과 성적 (A ~ F)



Linear Regression (선형 회귀)

- 실생활의 데이터는 선형적인 경우가 많다.
- hypothesis 와 cost 를 이용해 **적절한 모델을** 도출해야함

어떤 모델을 좋은 모델이라 할 수 있나?

- 우리가 예측한 모델과 실제 데이터 값의 거리를 비교 (**Cost function**) 해서 차이가 가장 적은 것이 좋은 모델임

Cost function

- 모델과 실제 데이터의 거리를 계산하는 함수

$$(H(x) - y)^2$$

- 예측함수

$$H(x) = Wx + b$$

- cost 함수 식

$$\text{cost} = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$$

- cost 함수를 code 로 나타내기

$$\text{cost} \times (W, b) = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$$

$$\text{code} \Rightarrow \text{cost} = \text{tf.reduce_mean}(\text{tf.square} \times (\text{hypothesis} - y))$$

- 목표: W와 b 의 값을 최소화 시키는 것!

$$\underset{W, b}{\text{minimize}} \times \text{cost} \times (W, b)$$



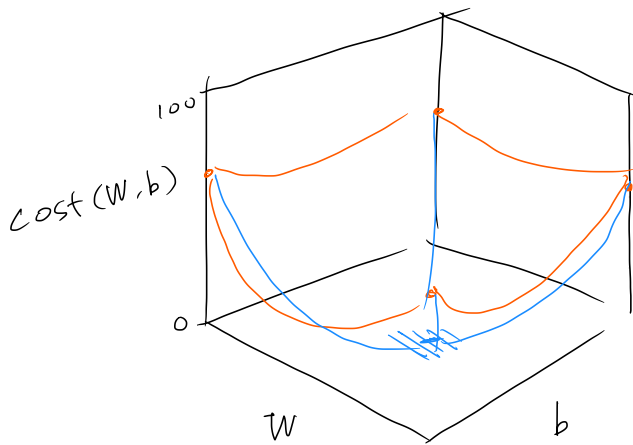
Minimize Cost

Gradient descent algorithm : 경사도를 이용해 최소 값을 찾기 좋은 알고리즘

$$W: ? = W - \alpha \frac{1}{m} \sum_{i=1}^m (Wx^{(i)} - y^{(i)})x^{(i)}$$

경사도를 구하는 방법 = 미분

Convex function : 어느점에서 시작하던 우리가 원하는 값을 얻을 수 있음



우리의 cost function이 Convex
형태로 나타나게 설계해야함!