



Choose  $\theta$  s.t.  $h(x) \approx y$  for training data

$$h_\theta(x) = h(x)$$

That Minimizes  $\rightarrow$   $\underset{\substack{\downarrow \text{수식} \\ \text{적용}}}{\sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2} = J(\theta)$  cost function / loss function

Gradient descent 경사하강법

Start with some  $\theta$  (say  $\theta = \vec{0}$ )

keep changing  $\theta$  to reduce  $J(\theta)$



iteration 한 번씩 하면서 새  $\theta$ 를 찾음 1 step → update the parameter  $\theta$

assignment (♡)

$$\theta_j := \theta_j - \alpha \cdot \frac{\partial}{\partial \theta_j} J(\theta)$$

↴  
 Empirical error function, step function, 보통 미지수의 개수는 2~5개  
 learning rate (LR) over shoot → can pass the global minimum  
 LR가 커면 두 번째 진보 → slow, not efficient

$$\begin{aligned}
 \frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} (h_{\theta}(x) - y)^2 \\
 &= 2 \cdot \frac{1}{2} (h_{\theta}(x) - y) \cdot \frac{\partial}{\partial \theta_j} (h_{\theta}(x) - y) \\
 &= (h_{\theta}(x) - y) \cdot \frac{\partial}{\partial \theta_j} (\theta_0 x_0 + \theta_1 x_1 + \dots + \theta_n x_n - y) \\
 &= (h_{\theta}(x) - y) \cdot x_j
 \end{aligned}$$

↴  $x_j$ 는  $x_j$ 가 되고  
 $\theta_j$ 는 가짜지 않은 항은 0임

$$\theta_j := \theta_j - \alpha \cdot \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \cdot x_j$$

Repeat until convergence

Batch Gradient Decent

- process all training data as a "batch"
- cons: when the data set is too large, (when M is very big)  
1 single step becomes very slow

### Stochastic Gradient Decent

Repeat { (for every j)}

For  $i=1$  to  $m$  {

$$\theta_j := \theta_j - \alpha (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

}

parameters will oscillate and may not converge to the global-global min.  
but it's faster when the data set is very very large.

(Batch  
Stochastic) Gradient Decent: iterative (multiple steps).  
used for 'Generalized linear models', NN... etc

For the special case of linear regression, jump 1 step to the global optimum

Normal Equations (works only for linear regression)

직접 행렬 곱셈을 통해 최적의 해를 찾는다.

$$\nabla_{\theta} J(\theta) = \begin{bmatrix} \frac{\partial J}{\partial \theta_0} \\ \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \end{bmatrix} \quad \text{3-dimensional vector}$$

Given

$$A \in \mathbb{R}^{2 \times 2} \quad A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad f: \mathbb{R}^{2 \times 2} \rightarrow ?$$

if  $f(A) = A_{11} + A_{12}^2$  then  $f\left(\begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}\right) = 5 + 6^2$

$$\nabla_A f(A) = \begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \frac{\partial f}{\partial A_{12}} \\ \frac{\partial f}{\partial A_{21}} & \frac{\partial f}{\partial A_{22}} \end{bmatrix} = \begin{bmatrix} 1 & 2A_{12} \\ 0 & 0 \end{bmatrix} \quad \text{각 element에 대한 편미분 } (f(A) \text{은 plug-in})$$

Normal Eq. ⑥의 성질을 이용한 성질

If  $A$  is a square matrix  $A \in \mathbb{R}^{n \times n}$

$$\text{tr } A = \text{sum of the diagonal entries} = \sum_{i=1}^n A_{ii}$$

'trace of  $A$ '라고 함

$$\textcircled{1} \quad \text{tr } A = \text{tr } A^T$$

$$\textcircled{6} \quad z^T z = \sum_i z_i^2 \quad (\text{sum of squares of elements})$$

$$\textcircled{2} \quad f(A) = \text{tr } AB \rightarrow \nabla_A f(A) = B^T$$

$$\textcircled{3} \quad \text{tr } AB = \text{tr } BA$$

$$\textcircled{4} \quad \text{tr } ABC = \text{tr } CAB \quad (\text{cyclic permutation property})$$

$$\textcircled{5} \quad \nabla_A \text{tr } AAT^C = CA + C^TA \quad \text{analogous to } \frac{d}{da} a^2 c = 2ac$$

$$J(\theta) = \frac{1}{2} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \Rightarrow \frac{1}{2} (x\theta - y)^T (x\theta - y) \text{의 증명과정:}$$

(1) take the training examples and stack them up in rows

$$X\theta = \begin{bmatrix} \cdots (X^1)^T \cdots \\ \cdots (X^2)^T \cdots \\ \vdots \\ \cdots (X^n)^T \cdots \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} = \begin{bmatrix} X^{(1)\top} \theta \\ X^{(2)\top} \theta \\ \vdots \\ X^{(n)\top} \theta \end{bmatrix} = \begin{bmatrix} h_{\theta}(x^{(1)}) \\ \vdots \\ h_{\theta}(x^{(n)}) \end{bmatrix}$$

$$(2) \vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix} \quad (3) \vec{x}\theta - \vec{y} = \begin{bmatrix} h_{\theta}(x^{(1)}) - y^{(1)} \\ \vdots \\ h_{\theta}(x^{(n)}) - y^{(n)} \end{bmatrix} \quad (\text{difference between predictions} \Leftrightarrow \text{actual label})$$

위 항목의 성질 ⑥ 이용하면, 하단 줄의 모든 원소의 제곱의 합은  $\vec{z}^T \vec{z}$ 와 같다.

$$\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 = (\vec{x}\theta - \vec{y})^T (\vec{x}\theta - \vec{y})$$

$$\begin{aligned}
 \nabla_{\theta} J(\theta) &= \nabla_{\theta} \frac{1}{2} (\mathbf{x}\theta - \mathbf{y})^T (\mathbf{x}\theta - \mathbf{y}) \\
 &= \frac{1}{2} \nabla_{\theta} (\theta^T \mathbf{x}^T - \mathbf{y}^T)(\mathbf{x}\theta - \mathbf{y}) \quad \downarrow \text{단순전개} \\
 &= \frac{1}{2} \nabla_{\theta} [\theta^T \mathbf{x}^T \mathbf{x} \theta - \theta^T \mathbf{x}^T \mathbf{y} - \mathbf{y}^T \mathbf{x} \theta + \mathbf{y}^T \mathbf{y}] \quad \downarrow \text{Andrew Ng 강의 노트에 행렬만 } \\
 &= \frac{1}{2} [\mathbf{x}^T \mathbf{x} \theta + \mathbf{x}^T \mathbf{x} \theta - \mathbf{x}^T \mathbf{y} - \mathbf{x}^T \mathbf{y}] \\
 &\quad \mathbf{x}^T \mathbf{x} \theta - \mathbf{x}^T \mathbf{y} \text{ set to } \vec{0} \\
 &\quad \mathbf{x}^T \mathbf{x} \theta = \mathbf{x}^T \mathbf{y} \\
 &\therefore \text{optimal } \theta = (\mathbf{x}^T \mathbf{x})^{-1} \mathbf{x}^T \mathbf{y}
 \end{aligned}$$