$$h(x) = \sum_{j=0}^{n} \theta_j x_j = \theta^T X \qquad \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix}$$

$(x^{(i)}, y^{(i)}) \rightarrow i^{th}$ training example $\qquad X = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_m \end{bmatrix}$

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

$f(x) = x^2$

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

$f'(x) = 0$

cost func^n

Task $\Rightarrow$ Choose θ s.t. $J(\theta)$ is minimum.

$$\frac{dx^2}{dx} = 2x$$

$$2x = 0$$

$$x = 0$$

① LMS Algo (Gradient descent)

Starts with θ (say $\theta = \vec{0}$)

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

$:=$ assignment operator

α learning rate (size of step towards a minima)

$\theta - 0.01 \times 2\theta$

$\theta - 0.02\theta$

$1.98\theta \cdot 0.01$

$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{\partial}{\partial \theta_j} \frac{1}{2} (h_\theta(x) - y)^2$$

$$= \frac{1}{\cancel{2}} \cdot \cancel{2} (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^{n} \theta_i x_i - y \right)$$

$$= (h_\theta(x) - y) \cdot (x_j)$$

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

Repeat this until convergence

(for $j = 0, 1, \ldots, n$) $\{$

$$\theta_j := \theta_j - \alpha \sum_{i=0}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

$\}$