# Gradient Descent algorithm

Remember: hypoth. funct. $h_\theta(x) = \theta_0 + \theta_1(x)$

cost funct. $J(\theta_0,\theta_1) = \frac{1}{2m}\sum^{m}(h_\theta(x_i)-y_i)^2$

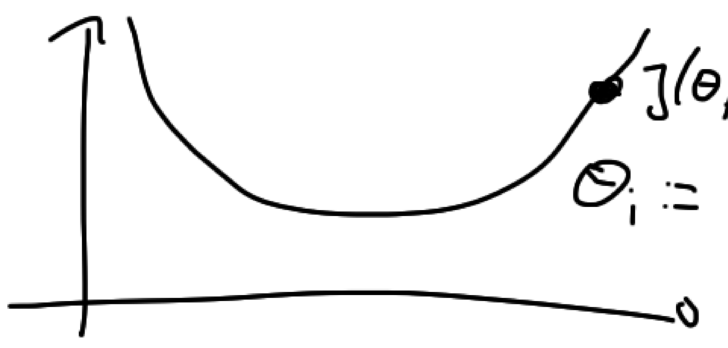
goal => minimize cost function

repeat until convergence {

$$\theta_j := \theta_j - \underbrace{\alpha}_{\text{learning rate}} \frac{\partial}{\partial\theta_j} J(\theta_0,\theta_1)$$

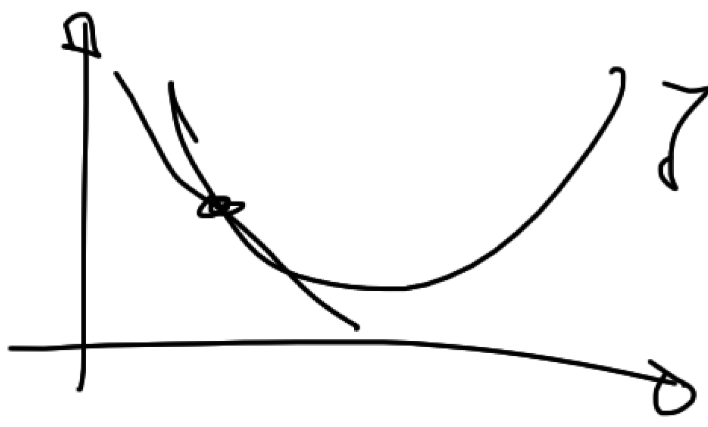
}

$\min_{\theta_1} J(\theta_1) \quad \theta_1 \in \mathbb{R}$



$J(\theta_1)$, $\theta_1 \in \mathbb{R}$

$\theta_1 := \theta_1 - \alpha \underbrace{\frac{\partial}{\partial\theta_1} J(\theta_1)}_{\geq 0} \Rightarrow \theta_1 := \theta_1 - \alpha \cdot$ pos. number



$J(\theta_1) \Rightarrow \frac{\partial}{\partial\theta} J(\theta) < 0 \Rightarrow \theta_1 := \theta_1 + \alpha \cdot$ (negative number)

## Use Gradient descent to minimize cost funct

$$\frac{\partial}{\partial\theta_j} J(\theta_1,\theta_0) = \frac{\partial}{\partial\theta_j}\frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x_i)-y_i)^2 = \frac{\partial}{\partial\theta_j}\frac{1}{2m}\sum_{i=1}^{m}(\theta_0+\theta_1 x_i - y_i)^2$$

$\Rightarrow$ calculate this for $\theta_0$ & $\theta_1$ $\Rightarrow$

$$\begin{cases} j=0: & \frac{\partial}{\partial\theta_0} J(\theta_0,\theta_1) = \frac{1}{m}\sum(h_\theta(x_i)-y_i) \\ j=1: & \frac{\partial}{\partial\theta_1} J(\theta_0,\theta_1) = \frac{1}{m}\sum(h_\theta(x_i)-y(i)) \cdot x(i) \end{cases}$$

this leads to following algorithm: repeat until convergence {

$$\theta_0 := \theta_0 - \alpha\frac{1}{m}\sum^{m}(h_\theta(x_i)-y_i)$$

$$\theta_1 := \theta_1 - \alpha\frac{1}{m}\sum(h_\theta(x_i)-y_i)\cdot x_i$$

}

## Batch Gradient Descent:

"Batch": each step of gradient descent uses all the training examples $\sum_{i=1}^{m}(h_\theta x_i - y_i)$

Let $\theta_0 = \theta_1 \Rightarrow$

$\theta_0 := \theta_0 + \frac{1}{m}\sum(h_\theta(x)-y)$

$\theta_1 := \theta_1 + \frac{1}{m}\sum(h_\theta(x)-y)x$