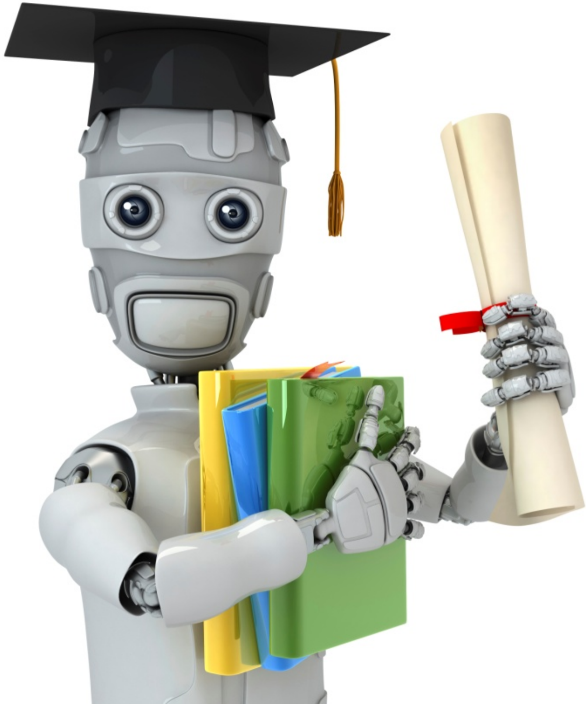


Machine Learning

Linear regression with one variable

Gradient descent intuition

Gradient descent algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial\theta_j} J(\theta_0, \theta_1)}$$

(simultaneously update $j = 0$ and $j = 1$)

}

Learning Rate

rate

derivative

$\min_{\theta_1} J(\theta_1)$

$\theta_1 \in \mathbb{R}$.

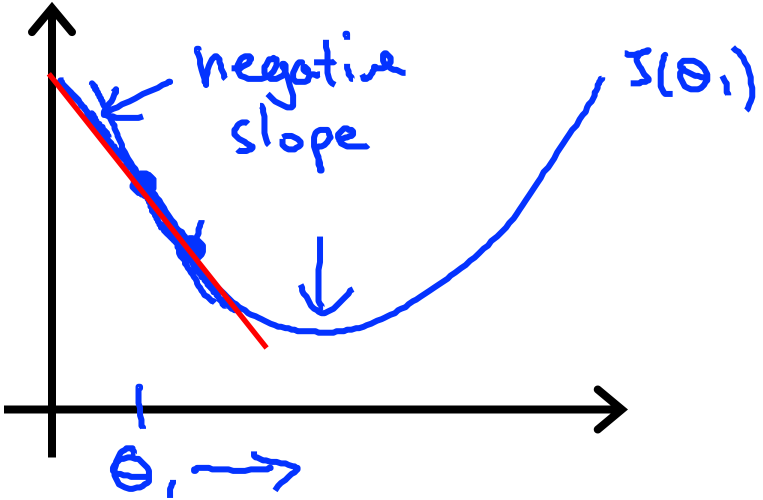
$(\theta_1 \in \mathbb{R})$

$\frac{\partial}{\partial \theta_1}$

$$\theta_1 := \theta_1 - \alpha \boxed{\frac{d}{d\theta_1} J(\theta_1)}$$

≥ 0

$$\theta_1 := \theta_1 - \underline{\alpha} \cdot (\text{positive number})$$



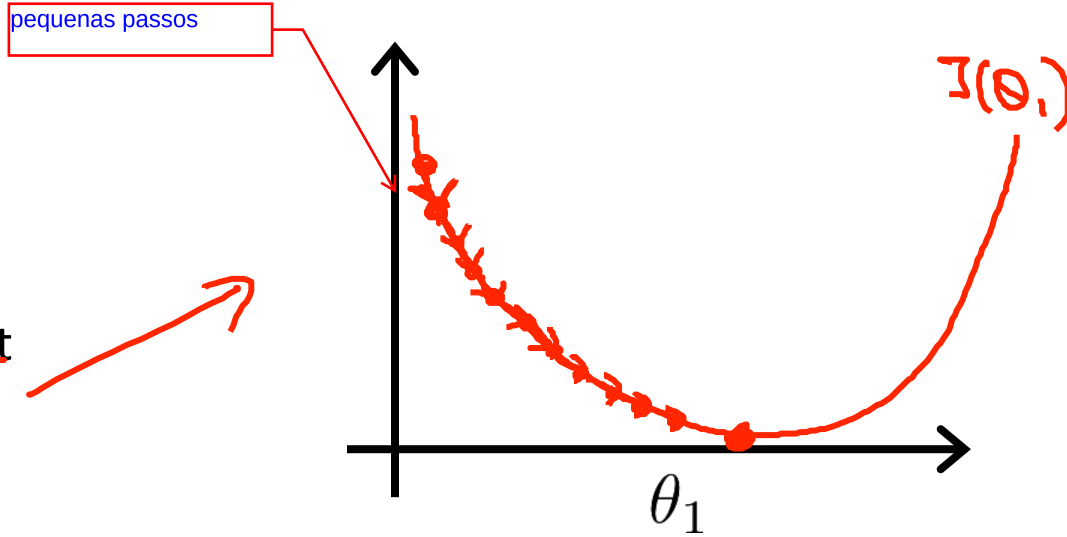
$$\underline{\frac{d}{d\theta_1} J(\theta_1)}$$

≤ 0

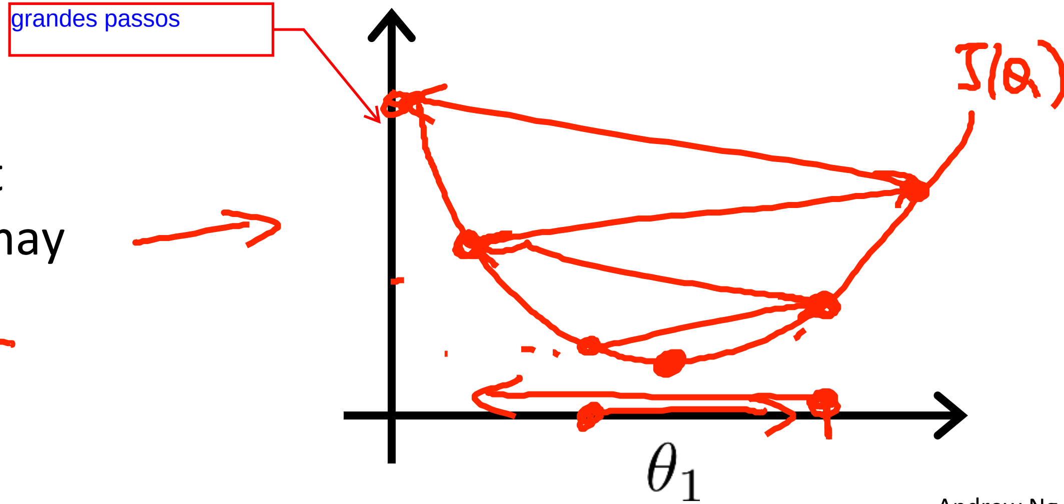
$$\theta_1 := \underline{\theta_1} - \alpha\,(\text{negative number})$$

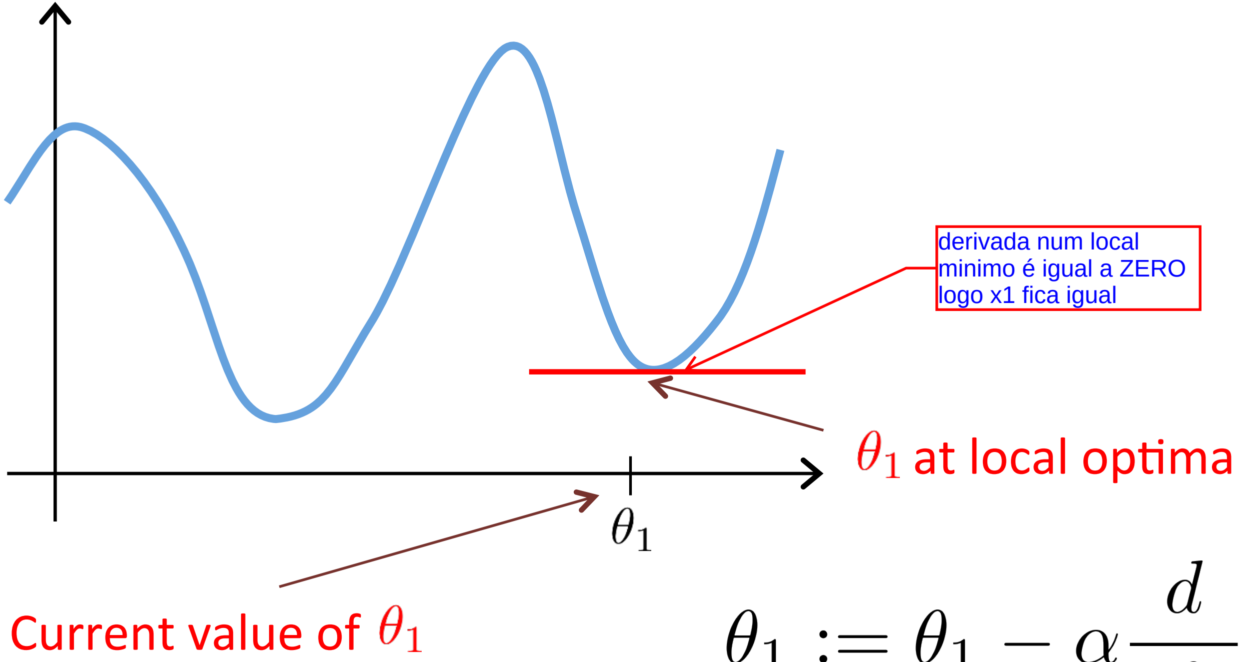
$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

- If α is too small, gradient descent can be slow.



- If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.





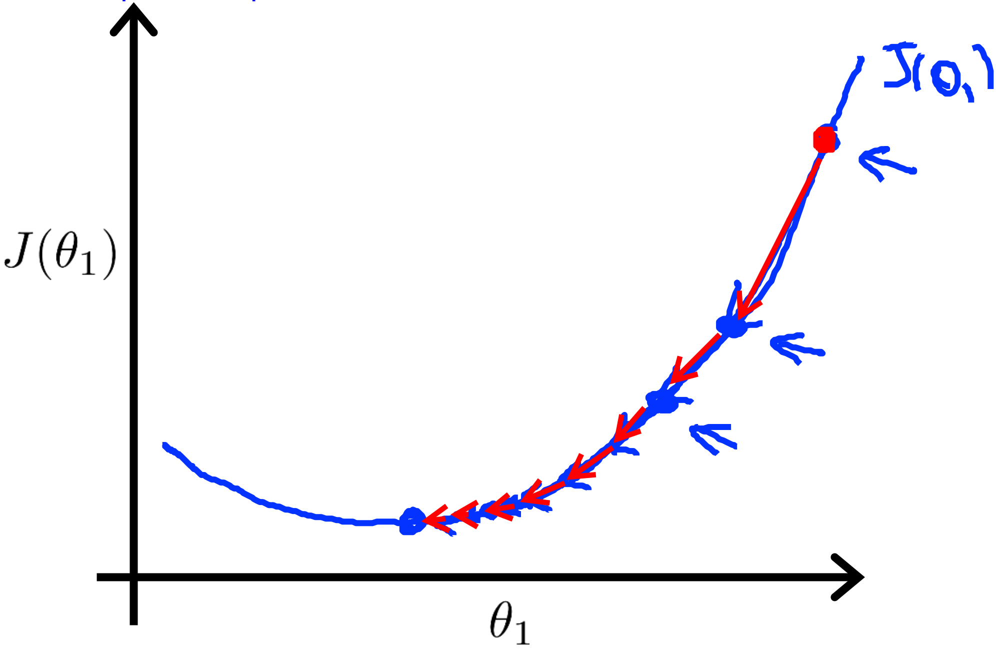
$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

x1 = xi - x*0

Gradient descent can converge to a local minimum, even with the learning rate α fixed.

$$\theta_1 := \theta_1 - \alpha \boxed{\frac{d}{d\theta_1} J(\theta_1)}$$

a medida que nos aproximamos do minimo aderivada écada vez menor



As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.