



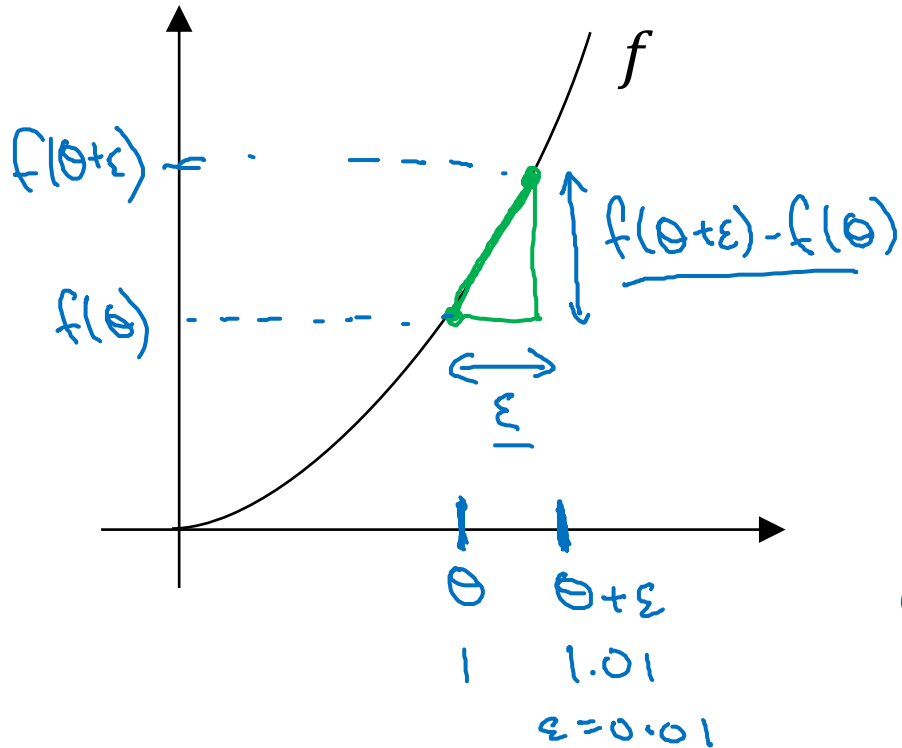
deeplearning.ai

Setting up your
optimization problem

Numerical approximation
of gradients

Checking your derivative computation

I $f(\theta) = \theta^3$
 $\theta \in \mathbb{R}.$



$$g(\theta) = \frac{d}{d\theta} f(\theta) = f'(\theta)$$

$g(\theta) = 3\theta^2$

$g(\theta) = 3 \cdot (1)^2 = 3$
 when $\theta = 1$

$\frac{dw}{db}$

$$\frac{f(\theta + \epsilon) - f(\theta)}{\epsilon} \approx g(\theta)$$

$$\frac{(1.01)^3 - 1^3}{0.01} = 3.0301 \approx 3$$

Annotations for the calculation above:

- 3.1 and 3.2 are written below 3.0301 .
- 0.0301 is written below 3.0301 .
- Arrows indicate the approximation process: $3.0301 \approx 3$.

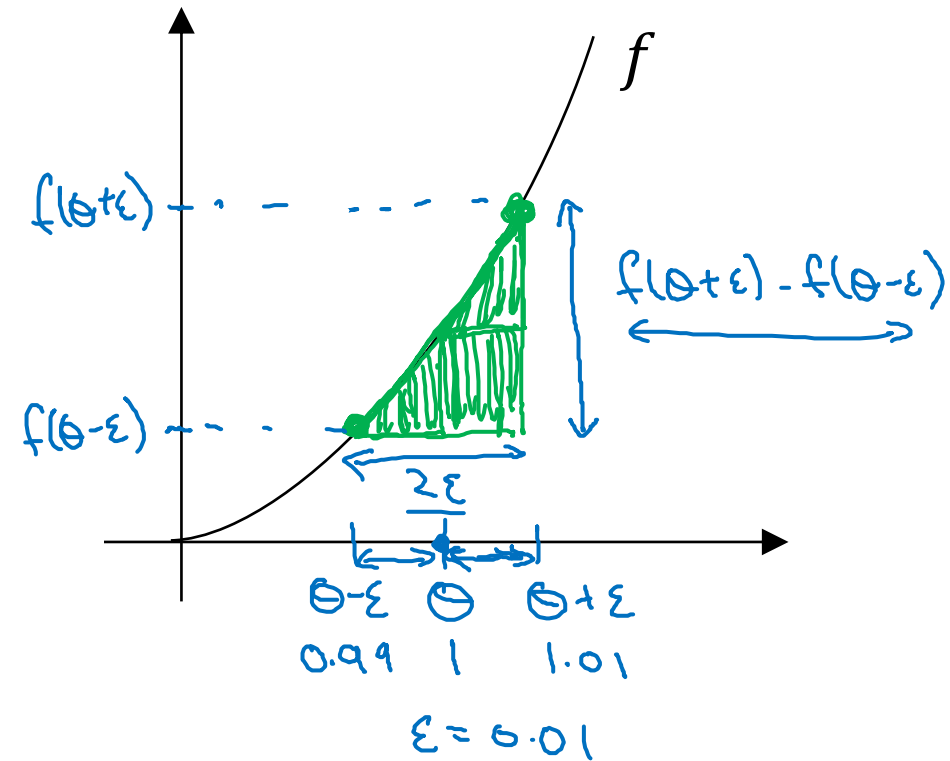
$\theta = 1$

$\theta + \epsilon = 1.01$

Checking your derivative computation

$$\underline{f(\theta) = \theta^3}$$

Using two sided difference way of approximating the derivative



$$\left[\frac{f(\theta + \epsilon) - f(\theta - \epsilon)}{2\epsilon} \approx \underline{g(\theta)} \right]$$

$$\frac{(1.01)^3 - (0.99)^3}{2(0.01)} = 3.0001 \approx 3$$

$$g(\theta) = 3\theta^2 = 3$$

approx error: 0.0001

(prev slide: 3.0301. error: 0.03)

Two sided difference formula is much more accurate.

$$\left\{ \begin{array}{l} f'(\theta) = \lim_{\epsilon \rightarrow 0} \frac{f(\theta + \epsilon) - f(\theta - \epsilon)}{2\epsilon} \quad \begin{array}{l} O(\epsilon^2) \\ 0.01 \\ \underline{0.0001} \end{array} \quad \left| \quad \frac{f(\theta + \epsilon) - f(\theta)}{\epsilon} \quad \begin{array}{l} \text{error: } O(\epsilon) \\ 0.01 \end{array} \end{array} \right.$$