deeplearning.ai

Setting up your optimization problem

Numerical approximation of gradients

Checking your derivative computation

$f(\theta) = \theta^3$

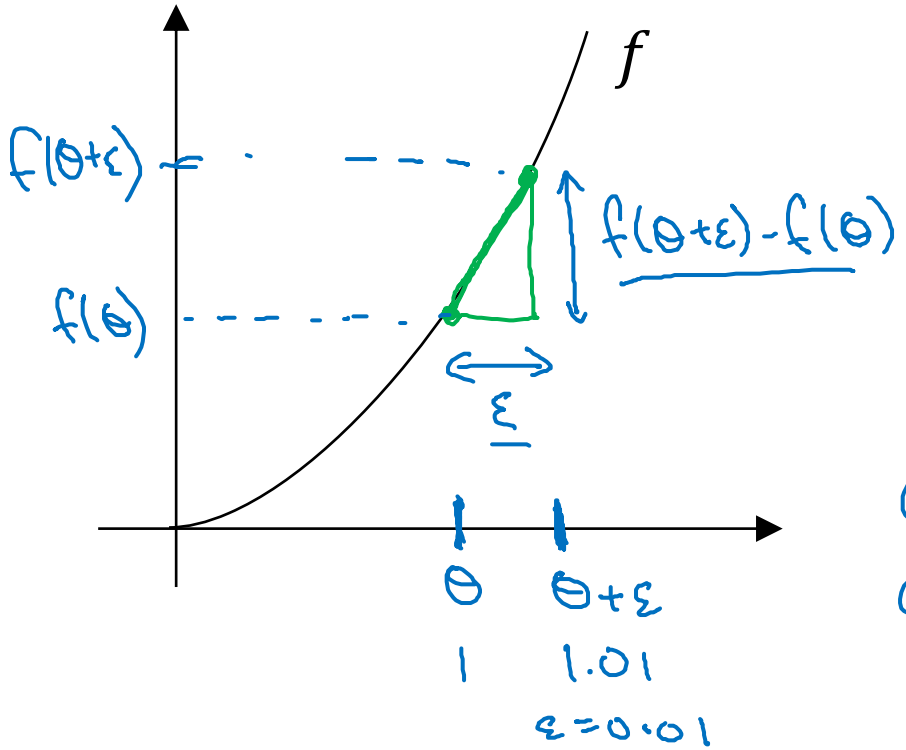
$\theta \in \mathbb{R}$.

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

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

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

$\frac{dw}{db}$



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

$\theta = 1$

$\theta + \varepsilon = 1.01$

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

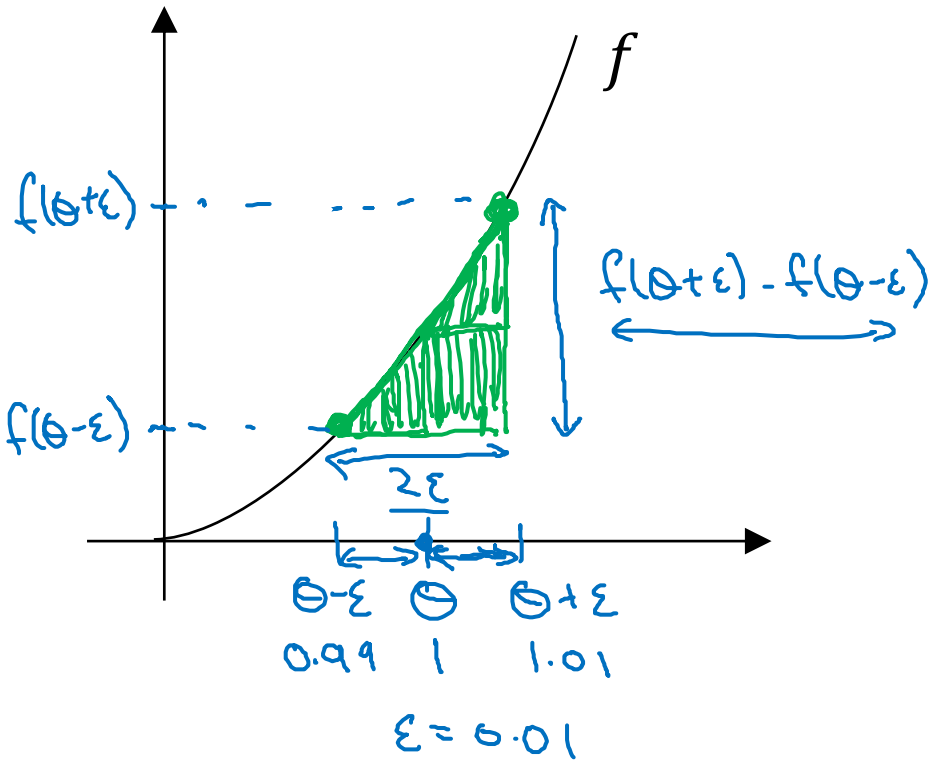
0.0301

3.1

3.2

Checking your derivative computation

$f(\theta) = \theta^3$



$$\frac{f(\theta+\varepsilon) - f(\theta-\varepsilon)}{2\varepsilon} \approx g(\theta)$$

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

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

approx error: 0.0001

(prev slide: 3.0301. error: 0.03)

$f'(\theta) = \lim_{\varepsilon \to 0} \frac{f(\theta+\varepsilon) - f(\theta-\varepsilon)}{2\varepsilon}$ $O(\varepsilon^2)$ 0.01 0.0001

$\frac{f(\theta+\varepsilon) - f(\theta)}{\varepsilon}$ error: $O(\varepsilon)$ 0.01

Andrew Ng

当你实现梯度逆传播时 你会发现一个测试 名叫梯度检验 可以帮助你确保 你的梯度逆传播的实现是正确的 因为有时写下这些方程后 你无法百分百确定 你的所有细节都做对了 实现了正确的逆传播 所以 为了之后能介绍梯度检验 我们先谈谈如何在数值上近似计算梯度 在下一个视频中 我们再讨论如何实现 梯度检验 从而确保逆传播的实现是正确的 我们来看一下函数f 我把它画在了这里 这个f(θ)等于θ立方 我们从某些θ取值出发 比如θ=1 这次我们不再将θ向右扰动到θ+ε 我们将同时向左右扰动 得到θ−ε和θ+ε 所以这里是1 这里是1.01 这里是0.99 ε与之前一样是0.01 事实上 相比于取这个小三角形 并计算高和宽 你还能得到更好的估计值 如果你选取这个点f(θ−ε)和这个点 然后计算这个大三角形的高和宽 由于一些技术原因 我在此不作解释 这个大三角形的 高和宽会给你一个θ点处梯度更好的近似值 很容易发现 如果不取右上的这个小三角形 你可以认为你有两个小三角形 右上的这个和左下的这个 你相当于同时考虑这两个三角形 通过使用这个大的绿三角形 所以你这次取了双侧的差值而不是单侧的差值 我们来写成数学形式 这个点是f(θ+ε) 而这个点是f(θ−ε) 所以这个大的绿三角形的高是f(θ+ε)−f(θ−ε) 所以这个大的绿三角形的高是f(θ+ε)−f(θ−ε) 至于宽度 这里是1个ε 这里是第2个ε 所以这个绿色三角形的宽就是2ε 所以这里的高度就是 首先是高度 也就是f(θ+ε)−f(θ−ε)除以宽度 也就是2ε 我写在这里了 这个很可能与g(θ)很接近 代入这些值 记住f(θ)是θ立方 这里θ+ε也就是1.01 我取它的立方 然后减去0.99的立方 再除以2∗0.01 你可以暂停视频并在计算器上算一下 你会得到结果是3.0001 前一个幻灯片中 我们知道g(θ) 也就是3θ^2 而θ是1 所以这两个值十分接近 近似误差是0.0001 在上一个幻灯片中 我们取过单侧的差值 θ和θ+ε之间的结果是3.0301 所以近似误差就是0.03而不是0.0001 所以用这个取双侧差值的方法 来近似导数 你会发现结果非常接近3 所以这将让你更加自信 g(θ)很可能就是求f导数的正确实现 当你把这个方法用于梯度检验和逆传播时 它运行起来很可能比用单侧差值要慢两倍 但从实践的角度 我认为这个方法值得一用 因为它精确很多 我再说一些选修的理论知识 给你们之中比较熟悉微积分的人 如果你听不懂我即将说到的内容也没有关系 事实上 导数的正式定义 就是对于很小的ε 计算[f(θ+ε)−f(θ−ε)]/(2θ) 就是对于很小的ε 计算[f(θ+ε)−f(θ−ε)]/(2θ) 而导数的正式定义就是右边这个公式 当ε趋近于0时的极限 极限的定义就是微积分课上学的那样 但我这里不再详述 对于一个非零的ε值 你可以证明这个近似的误差 在ε平方这个阶上 ε是个很小的数 如果ε是0.01 就像这里 那么ε平方就是0.0001 这个大O记号就表示误差就是某个常数乘以这个 这就是我们的近似误差 这个例子中的大O的常数恰好就是1 相比而言 如果我们用这边的另一个公式 误差就在ε这个阶上 当ε是一个小于1的数时 ε就比ε平方大很多 这也就是为什么 这个公式不如左边这个公式精确 这也就是为什么我们做梯度检验时采用双侧差值 你计算f(θ+ε)−f(θ−ε)再除以2ε 而不使用这个不够精确的单侧差值 如果你不理解我最后说的两点 所有东西都写在这儿了 Don't worry about it.Don't worry about it. 对微积分和数值近似比较熟悉的人 可以多学一些 简单来说就是双侧差值的公式更加精确 我们在下一个梯度检验视频中就会用到这个 所以你们学习了如何取双侧差值 来在数值上验证是否给定的函数g(θ) 是函数f的导数的正确实现 我们来看看如何使用这个来验证是否 你的逆传播的实现是正确的 还是说里面有错误要剔除掉