关于梯度下降的算法原理

我们可以从之前的求某 $\min\{\dots\}$ 的最小值来说。在CS229中，我们之前使用的 $L(\omega)$ 被称为min函数，也就是对我们来说，我们可以重新定义

对于一个线性模型

$$h_\omega(x) = \omega^T x + b \qquad (d+1\text{为}\omega\text{的向量}+1)$$

我们令 $$J(\omega) = \frac{1}{d+1}\sum_{i=1}^{m}\left(h_\omega(x^{(i)}) - y^{(i)}\right)^2$$

那么 $\min\{J(\omega)\}$ J 是关于 ω 的函数

求 J 的最小值，就可以得到 ω 的最优解

之前，我们发现可以通过其它方法计算

但是今天我们将会采用更加通用(流行)的方法过降. 梯度下降 [Gradient Descent]

所以听到 GD 的时候，它确实有时在说 梯度下降 [梯度下降]

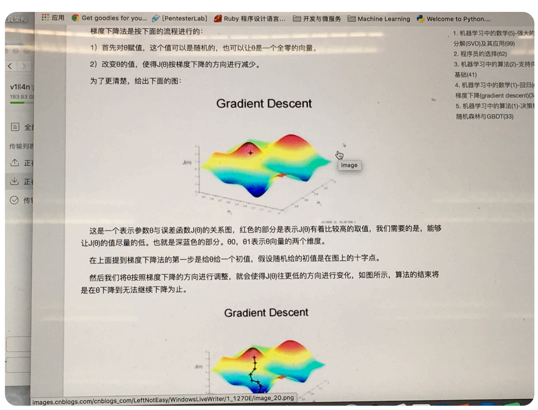
好的. 我们开始.

$J(w)$ 函数我们称之为成本函数 Cost Function

$J(w)$ 必须可微数

$\Delta J(w)$ 表在 w 处的微分

PS: Cost Function 这个翻译怪怪的
但是 Cost 这个词一定要记住



基本步骤：

① 对 ω 赋值 [初始值]

② 对成本函数 J(ω) 进行微分运算

$$\Delta(\omega) = \frac{\partial J(\omega)}{\partial \omega}$$

Δ(ω) 的方向为正 表上升

Δ(ω) 的方向为负 表下降

③ 选择方向 [梯度向下 为负]

确定学习率 α (Step)，乘以学习率(步长)

$\underline{\omega = \omega - \alpha\Delta(\omega)}$ ⇒ 更新 ω

④ 继续学习……

关于 GD 的算法：BGD SGD 与 MBGD