Gradient Descent Optimization Algorithms (01)

机器学习中的优化器

Grissom, 2019/07/14
http://sunbo2019.github.io

目录

Q: 什么是“优化算法”？
A: 就是求 [**参数取什么值时，损失函数能取到最小值**] 的方法。

基于**梯度下降思想**的算法，常用的有如下几种：

- SGD（梯度下降）
  1. Batch Gradient Descent （每次迭代都使用全部数据的“批量梯度下降”）
  2. Stochastic Gradient Descent （每次迭代只取一条数据的“随机梯度下降”）
  3. Mini-batch Gradient Descent （每次迭代取一小批数据的“批量梯度下降”）

- Momentum（基于梯度，引入物理中的“动能（惯性）”来使SGD的收敛过程不那么剧烈）
- Adagrad（基于梯度，自动调整学习率）
- Adam（Adagrad + Momentum，目前使用最多、效果最好、最流行）

那么，什么是“梯度下降”？ 为什么要求梯度下降？

有些函数的一阶导，是可以直接解出的，比如：

$$f(x) = x^2 + 3 \quad \Rightarrow \quad f'(x) = 2x = 0 \quad \Rightarrow \quad x = 0$$

有些函数的一阶导，是**难以直接求出**的，比如：

$$f(x) = x^4 + x^{\frac{5}{3}} \quad \Rightarrow \quad f'(x) = 4x^3 + \frac{5}{3}x^{\frac{2}{3}} = 0 \quad \Rightarrow \quad x = ?$$

那么，这时应该怎么办呢？
使用“梯度下降”法：通过每次**往梯度变小的方向走一小步**来逼近最优解。

[若原函数的一阶导函数是凸函数，那么一定有全局最优解；
若原函数的一阶导函数不是凸函数，那么只能找到局部最优解]

具体怎么做呢？（1）

step 01: 随机给定一个x值。

step 02: 往**一阶导数的梯度**变小的方向走一小步（学习率），并得到新位置上的x值。

step 03: 判断：

1) 新、旧x值是否有变化。（若无变化，说明梯度不再更新，满足终止条件）

2) 计算新x值时，对应的梯度是否为0。

（若为零，说明取此x值时，原函数可以取得最小值，满足终止条件）

具体怎么做呢？（2）

以 $f'(x) = 4x^3 + \frac{5}{3}x^{\frac{2}{3}} = 0$ 为例，

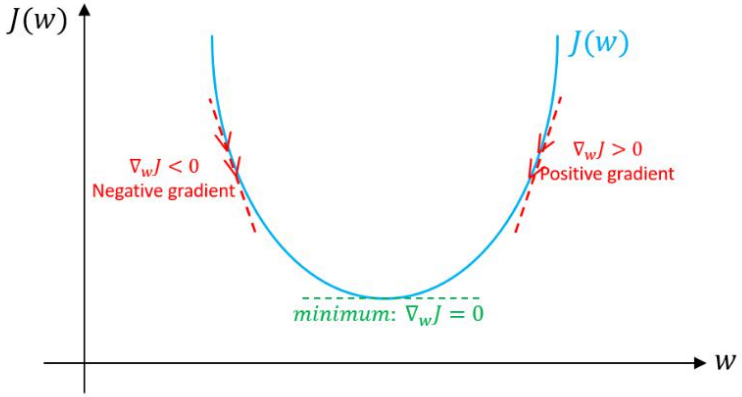
step1: 随机给定x = 2，并设学习率为0.1
step2: x = 2 - 0.1 * f'(2) = -1.46
x = -1.46 - 0.1 * f'(-1.46) = -0.107
x = -0.107 - 0.1 * f'(-0.107) = -0.08
x = -0.08 - 0.1 * f'(-0.08) = -0.06
...
x = -0.006 - 0.1 * f'(-0.006) = -0.003
...
x = 0 - 0.1 * f'(0) = 0   => 满足终止条件，退出

求得解：当 x = 0 时，原式可以取得最优值。

本质上，就是**通过梯度来找到一个附近的、新的X值**。计算这个新位置上的梯度后，再利用它去寻找一个新的X值。如此迭代，直到满足退出条件。简方之，**梯度 <-> X值，迭代**。

理解时，可能存在的疑问

疑问1： $x_j \leftarrow x_j - \alpha f'(x)$ **，这里总是“减”吗？如果这里的导数是负的呢？**

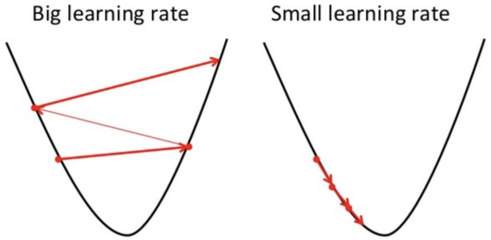


如图所示，如果是在左侧，其梯度为负数，$x_j - \alpha f'(x)$ 值增大
=> 新的 x_j 值是在往最优值逼近

如果是在右侧，其梯度为正数，$x_j - \alpha f'(x)$ 值减小
=> 新的 x_j 值是在往最优值逼近

所以，此表达式总是成立！

疑问2：步长设多少合适？过大或达小会怎样？

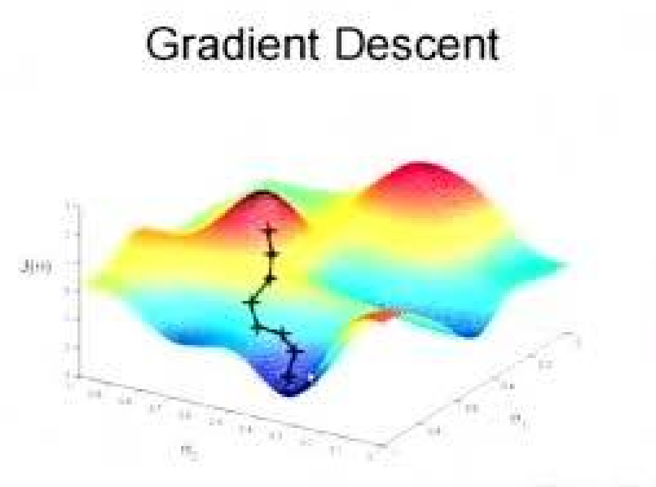


如果步长过大，会错过最优值，造成梯度下降时来回震荡。
如果步长过小，迭代过程会很漫长，梯度收敛慢。
步长设多少合适，需要多用几个值去试。哪个值效果最好就用哪个！

多元函数的梯度下降（1）

先明确两个概念：

偏导数：多元函数沿坐标轴的变化率
方向导数：多元函数沿任意方向的变化率



假设有一个多元函数f(x)，且在空间中有一点x。现在我们从点X出发，沿着方向V前进一个步长。我们希望：**下降的速度最快！**
即：f(x) - f(x+v)的值最大！

对 f(x + v)在x处进行Taylor一阶展开，得到：

$$f(x+v) \approx f(x) + \nabla f(x)^T v \quad \rightarrow \quad f(x) - f(x+v) \approx -\nabla f(x)^T v$$

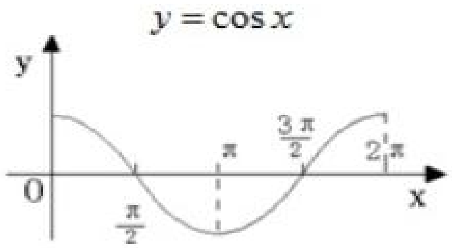
这里的 $\nabla f(x)$ 是f(x)的偏导数，v是方向导数。那么，如何才能让 $-\nabla f(x)^T v$ 的值最大呢？

多元函数的梯度下降（2）

根据 **向量点积** 公式：

$$\vec{a} \bullet \vec{b} = |\vec{a}||\vec{b}| \cos\theta$$

根据 三角函数 公式：



解释了
为什么朝着梯度的反方向前进，函数值下降最快

若要使得 $-\nabla f(x)^T v$ 的取值最大，那么方**向导数V**只能与**偏导数**成**反方向**！（180度角！）