最速下降法

前言

红色石头对于为什么局部下降最快的方向就是梯度的负方向? 讲的非常清晰明了，我这就是从他那学习的，写这个就是希望自己能从头到尾捋一遍谨防眼高手低。

最速下降法公式

首先介绍完整的公式，假设$f(x)$其最速下降法公式为

$$x_{k+1} = x_k - \lambda \nabla f\left(x_k\right), \tag{1}$$

其中，k表示迭代步数，∇f表示梯度，λ表示迭代的步长。

公式推导

该公式是基于泰勒公式的一阶近似推导的，对$f(x)$在$x = x_0$进行泰勒一阶展开可得，

$$f(x) \approx f(x_0) + \nabla f(x_0)(x - x_0)。 \tag{2}$$

对公式（2）移项可得，

$$f(x) - f(x_0) \approx \nabla f(x_0)(x - x_0)。 \tag{3}$$

如果想要$f(x)$变小，则$f(x) - f(x_0) < 0$（这只能放在一维来理解，高维向量是没有大小的）,所以则有

$$\nabla f(x_0)(x - x_0) < 0。 \tag{4}$$

若要$f(x)$变得更小，则公式（4）应该更小，由向量点乘公式可知当两向量方向相反时，其值是最小的为-1。所以有

$$\frac{x - x_0}{|x - x_0|} = -\frac{\nabla f(x_0)}{|\nabla f(x_0)|} \tag{5}$$

$$x = x_0 - \frac{|x - x_0|}{|\nabla f(x_0)|} \nabla f(x_0) \tag{6}$$

因为x是未知的，所以$|x - x_0|$也是未知的，因此就需要通过人为来设定步长，最终的公式为

$$x_{k+1} = x_k - \lambda \nabla f(x_0) \tag{7}$$

步长λ的求解

如果步长大，迭代快但在最优点可能来回迭代；如果步长小，则搜索速度会很慢。一般采用精确步长搜索法或一维搜索方法（更多步长求解算法可见凸优化系列二:确定步长一维搜索算法）。
精确步长搜索法
该方法求解的是最优步长，但并不容易求解

$$\lambda_k = arg\min_{\lambda_k} f\left(x_k - \lambda_k \nabla f\left(x_k\right)\right) \tag{8}$$

非精确步长搜索法
黄金分割法：该方法是通过固定比率缩短区间来求解的，只能适用于区间内存在单一极小值点的情况。详细介绍过程一维搜索的最优方法(黄金分割法)。 设在区间$[a,b]$内存在任意两点p,q满足$a<p<q<b$则有

$$p=b-0.618(b-a) \tag{9-1}$$

$$q=a+0.618(b-a) \tag{9-2}$$

经过比较$f(p)$和$f(q)$的大小后确定新区间为$[a,q]$或$[p,b]$,然后重复上述步骤。

PS

1.适用场景 对误差函数或优化函数进行求解，当无解析解或解析解求解难度大，适用该迭代优化算法。

2.最速下降法优缺点
优点：简单、计算量小

缺点：最速下降法相邻两次搜索方向正交，靠近极小值时收敛速度减慢。

3.问题
（1）再推导的过程中即采用低维的思路又采用高维的思路，总感觉不严谨，差点意思。后续学好了再做补充。
（2）确定步长算法中如果不是一元函数的步长该怎么确定?

4.其他 markdown源码和pdf参见tusha-github。