

维基百科

自由的百科全书

应用于最优化的牛顿法

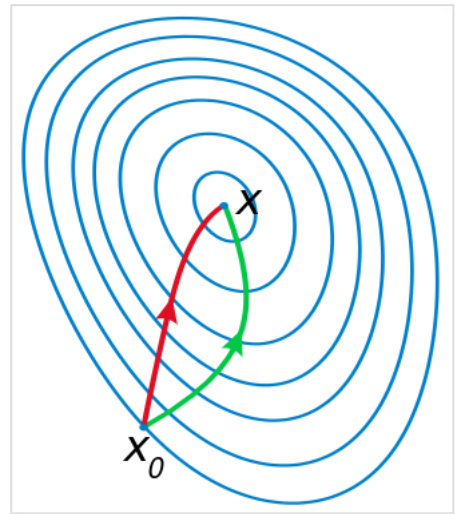
维基百科，自由的百科全书

牛顿法是微积分学中, 通过迭代以求解可微函数***f***的零点的一种算法 (即求***x***使得***f*(*x*) = 0**). 而在优化中, 牛顿法通常被运用于求解一个二次可微函数***f***的一阶导数***f'***的零点 (即求***x***使得***f'*(*x*) = 0**), 同时也是***f***的驻点. 因此从另一个角度而言, 应用于优化的牛顿法是搜索函数***f*(*x*)**的最小值或最大值的一种算法。

一维问题的牛顿法主要步骤如下: 取一个点***x*₀**为初值, 依如下公式迭代:

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)},$$

直至满足一定条件 (如***f'*(*x*_{*n*}) = 0**或***x*_{*n*+1} − *x*_{*n*} < ε**, 其中ε为一个给定的足够小的常量) 后, 算法终止。



最速下降法 (绿色) 与牛顿法 (红色) 在求最小值问题上的比较 (带有步长). 可见牛顿法根据曲率选择了一条“快捷方式”。

方法描述

在一维问题中, 牛顿法将构造一个以***x*₀**为首项, 收敛到***x*^{*}**的数列{***x*_{*n*}**}, 其中***x*^{*}**使得***f'*(*x*^{*}) = 0**成立.

f*(*x*)**在x* = *x*_{*n*}**处的二阶泰勒展开式***f*_{*T*}(*x*)**为:

$$f_T(x) = f_T(x_n + \Delta x) \approx f(x_n) + f'(x_n)\Delta x + \frac{1}{2}f''(x_n)\Delta x^2.$$

我们希望找到**Δ*x***使***x*_{*n*} + Δ*x***为***f*_{*T*}(*x*)**的一个驻点。则将上式对**Δ*x***进行求导:

$$0 = \frac{d}{d\Delta x}(f(x_n) + f'(x_n)\Delta x + \frac{1}{2}f''(x_n)\Delta x^2) = f'(x_n) + f''(x_n)\Delta x.$$

上述方程的解**Δ*x*** = $-\frac{f'(x_n)}{f''(x_n)}$ 满足

$$x_{n+1} = x_n + \Delta x = x_n - \frac{f'(x_n)}{f''(x_n)}$$

收敛于***f*_{*T*}(*x*)**的驻点***x*^{*}**.

几何意义

牛顿法的几何意义为: 在每一次迭代中, 均以一个二次函数去逼近***f*(*x*)**. 具体而言: 在一维问题中, 已知***x*_{*n*}**, ***f*(*x*_{*n*})**, ***f'*(*x*_{*n*})**及***f''*(*x*_{*n*})**, 设二次函数表达式为***a**x*² + *b**x* + *c***, 依下列方程求解未知数***a***, ***b***, ***c***,

$$\begin{aligned}ax_n^2 + bx_n + c &= f(x_n), \\2ax_n + b &= f'(x_n), \\2a &= f''(x_n).\end{aligned}$$

然后二次函数 $ax^2 + bx + c$ 的极值点即为下一个迭代点，

$$x_{n+1} = -\frac{b}{2a}.$$

而在高维问题中，上述的极值点也可以是鞍点。值得一提的是，如果 $f(\boldsymbol{x})$ 恰为一个二次函数，则其极值点只需一次迭代中即可找到。

高维问题求解

上述的一维问题的迭代法可以被推广至多维问题。只需将导数替换为梯度 ($\nabla f(\boldsymbol{x})$)，并将二阶导数的倒数替换为Hessian矩阵的逆矩阵 ($\mathbf{H}f(\boldsymbol{x})$)，即：

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - [\mathbf{H}f(\boldsymbol{x}_n)]^{-1} \nabla f(\boldsymbol{x}_n), n \geq 0.$$

通常，使用牛顿法时会加入一个步长变量 $\gamma \in (0, 1)$ 作微调以使每一步迭代都满足Wolfe条件，即，

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \gamma [\mathbf{H}f(\boldsymbol{x}_n)]^{-1} \nabla f(\boldsymbol{x}_n).$$

这个方法被称为无约束牛顿法，通常用于第一步之后的迭代。

只要牛顿法适用，其收敛于最小值或最大值的速度均颇快于最速下降法。事实上，对于每一个极小值，都存在一个邻域 \boldsymbol{N} 使得，只要Hessian矩阵是可逆的且是一个关于 $\boldsymbol{x} \in \boldsymbol{N}$ 的Lipschitz连续函数，以 $\boldsymbol{x}_0 \in \boldsymbol{N}$ 为初值，步长 $\gamma = 1$ 的牛顿法是二次收敛的。

求一个高维问题的Hessian矩阵的逆矩阵是一件颇费工夫的事情。在实际应用中，通常会用向量 $\Delta \boldsymbol{x} = \boldsymbol{x}_{n+1} - \boldsymbol{x}_n$ 作为线性方程组

$$[\mathbf{H}f(\boldsymbol{x}_n)] \Delta \boldsymbol{x} = -\nabla f(\boldsymbol{x}_n)$$

的解。这个求解过程中，透过使用各种矩阵分解方法同近似求解方法，求解速度可以大大提升。然而，这些矩阵分解方法或近似求解方法的使用需要满足一定条件；例如，Cholesky分解同共轭梯度法只有在 $\mathbf{H}f(\boldsymbol{x})$ 是正定矩阵时才适用。这看似是一个限制，但有时也能充当检验答案的工具；例如，在一个最小化问题 ($\min f(\boldsymbol{x})$) 中，求出一个 \boldsymbol{x}' 使得 $\nabla f(\boldsymbol{x}') = \mathbf{0}$ 但 $\mathbf{H}f(\boldsymbol{x})$ 不是正定矩阵，那么 $(\boldsymbol{x}', f(\boldsymbol{x}'))$ 只是 $f(\boldsymbol{x})$ 的一个鞍点而非极小值点。

另一方面，一个有约束的问题的求解过程可能会遇到当前解陷入一个鞍点的情况，这时的Hessian矩阵是对称不定的；此时则要使用其他方法，例如Cholesky分解的LDL^T变形或共轭梯度法等的方法，来迭代得出 \boldsymbol{x}_{n+1} 。

此外，为规避求Hessian矩阵的繁琐，也存在多种拟牛顿法，通过调整梯度以求出Hessian矩阵的近似。

如果Hessian矩阵 $\mathbf{H}f(\mathbf{x})$ 接近一个奇异矩阵, 则其逆矩阵会变得数值不稳定且迭代不会收敛. 此种情形下, 前人探索出了很多成功的方法来解决这个问题. 目标之一是通过引入修正矩阵 \mathbf{B}_n 使得 $\mathbf{H}f(\mathbf{x}_n) := \mathbf{H}f(\mathbf{x}_n) + \mathbf{B}_n$ 是对称正定的. 其中一种方法是将 $\mathbf{H}f(\mathbf{x}_n)$ 对角化, 选择 \mathbf{B}_n 使 $\mathbf{H}f(\mathbf{x}_n) + \mathbf{B}_n$ 有相同的特征向量, 但每一个 $\mathbf{H}f(\mathbf{x}_n)$ 的负特征值都被替换成 $\epsilon > 0$.

一个应用于莱文贝格- 马夸特方法 (其中用到了近似的Hessian矩阵) 的方法是引入一个带系数的单位矩阵 $\mu \mathbf{I}$, 系数在每一步迭代中调整. 对于较大的 μ 及较小的Hessian矩阵, 迭代将变得与以 μ^{-1} 为步长的最速下降法相似, 这将使得迭代收敛变慢, 但在Hessian矩阵不定或半定的情况下, 收敛更稳定.

参阅

- [拟牛顿法](#)
- [最速下降法](#)
- [高斯-牛顿算法](#)
- [莱文贝格- 马夸特方法](#)
- [置信域方法](#)
- [优化](#)
- [Nelder-Mead方法](#)

参考文献

- Avriel, Mordecai. [Nonlinear Programming: Analysis and Methods](#). Dover Publishing. 2003. ISBN 0-486-43227-0.
- Bonnans, J. Frédéric; Gilbert, J. Charles; Lemaréchal, Claude; Sagastizábal, Claudia A. [Numerical optimization: Theoretical and practical aspects](#). Universitext Second revised ed. of translation of 1997 French. Berlin: Springer-Verlag. 2006: xiv+490 [2017-08-07]. ISBN 3-540-35445-X. MR 2265882. doi:10.1007/978-3-540-35447-5. （原始内容存档于2013-07-19） .
- Fletcher, Roger. [Practical methods of optimization](#) 2nd. New York: John Wiley & Sons. 1987. ISBN 978-0-471-91547-8.
- Nocedal, Jorge; Wright, Stephen J. [Numerical Optimization](#). Springer-Verlag. 1999. ISBN 0-387-98793-2.

外部链接

- [Korenblum, Daniel. Newton-Raphson visualization \(1D\). Blocks. Aug 29, 2015 \[2017-08-07\]. ffe9653768cb80dfc0da. （原始内容存档于2014-07-14） .](#)

取自“<https://zh.wikipedia.org/w/index.php?title=應用於最優化的牛頓法&oldid=78788633>”

■