3.2 最速下降法

下降算法是求解无约束优化问题(3.0.1)的流行算法，那么一个自然的问题是：沿什么方向，目标函数值下降最快？

假设函数f一阶连续可微. 由命题1.4.1易得：负梯度方向$-g_k \neq 0$是f在x^k处的一个下降方向. 下面证明：负梯度方向$-g_k \neq 0$是f在x^k处的最速下降方向. 由

$$f(x^k + \lambda d^k) - f(x^k) = \lambda g_k^\top d^k + o(\|\lambda d^k\|)$$

可以看到：目标函数f沿方向d^k的变化率是$g_k^\top d^k$. 因此，最速下降方向是优化问题

$$\min_{d \neq 0} \; g_k^\top \left(\frac{d}{\|d\|} \right)$$

的最优解. 注意

$$g_k^\top \left(\frac{d}{\|d\|} \right) = -\|g_k\| \left\| \frac{d}{\|d\|} \right\| \cos\theta_k = -\|g_k\| \cos\theta_k,$$

其中θ_k是$-g_k$与d之间的夹角. 显然，当$\theta_k = 0$(即$\cos\theta_k = 1$)时，$g_k^\top \left(\frac{d}{\|d\|} \right)$最小. 此时有

$$\frac{d}{\|d\|} = -\frac{g_k}{\|g_k\|}.$$

因此，负梯度方向$-g_k \neq 0$是f在x^k处的最速下降方向.

3.2 最速下降法

最速下降法是以负梯度方向作为下降方向的极小化算法，又称梯度法. 它是求解无约束优化问题中最简单的算法. 下面给出求解无约束优化问题(3.0.1)的最速下降算法.

算法 3.2.1 （最速下降算法） 选择初始点$x^0 \in \mathbb{R}^n$. 置$k := 0$.

步*1* 计算$d^k = -g_k$. 如果$g_k = 0$, 算法终止.

步*2* 由线搜索确定步长$\lambda_k > 0$, 使得$f(x^k + \lambda_k d^k) < f(x^k)$.

步*3* 置$x^{k+1} := x^k + \lambda_k d^k$和$k := k + 1$, 转步*1*.

注. (1) 在理论分析中，取$g_k = 0$作为算法的终止规则，但在实际计算中，终止规则$g_k = 0$由$\|g_k\| \le \epsilon$来代替，其中$\epsilon > 0$是容许误差（后面的算法均如此，以后不再赘述）. (2) 当步长$\lambda_k > 0$由精确一维线搜索得到时，对应的算法由法国数学家Cauchy于1847年提出，它是求解无约束优化问题最早的数值方法.

3.2 最速下降法

现在考虑算法3.2.1的全局收敛性. 因为最速下降方向与目标函数的负梯度方向一致，所以有$\|g_k\| = \|d^k\|$. 因此，对所有的k有(3.1.3)式成立. 故由定理3.1.4、定理3.1.5以及定理3.1.6, 容易得到算法3.2.1的全局收敛性如下.

定理 3.2.1 假设$f:\mathbb{R}^n \to \mathbb{R}$连续可微有下界, 且其梯度函数$g$具有*Lipschitz*连续性. 设序列$\{x^k\}$由最速下降算法*3.2.1*产生，其中步长$\lambda_k$由精确一维线搜索得到、或由*Armijo*线搜索得到、或由*Wolfe*线搜索得到. 那么，算法*3.2.1*或者有限终止，或者$\lim_{k\to\infty}\|g_k\| = 0$.

3.2 最速下降法

例 3.2.1 利用精确一维线搜索的最速下降算法求解下述无约束优化问题 $\min\ f(x) = \frac{1}{2}x_1^2 + x_2^2$, 其中初始点取为 $x^0 = (2,1)^\top$.

解 直接计算可得

$$g(x) = \begin{pmatrix} x_1 \\ 2x_2 \end{pmatrix}, \quad G(x) = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}.$$

所以，目标函数f是正定二次函数，并且容易看出: 所求优化问题有唯一极小值点$x^* = (0,0)^\top$.

容易证明：如果f是正定二次函数，那么由精确一维搜索公式确定的步长λ_k满足

$$\lambda_k = -\frac{g_k^\top d^k}{(d^k)^\top G_k d^k}.$$

因此，求解该优化问题的迭代公式为

$$x^{k+1} = x^k - \frac{g_k^\top g_k}{g_k^\top G g_k} g_k.$$

3.2 最速下降法

因为$g_0 = g(x^0) = (2,2)^\top$, 所以由上述迭代公式可得

$$x^1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix} - \frac{(2\ 2)\begin{pmatrix} 2 \\ 2 \end{pmatrix}}{(2\ 2)\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{pmatrix} 2 \\ 2 \end{pmatrix}} \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \frac{1}{3}\begin{pmatrix} 2 \\ -1 \end{pmatrix}.$$

类似迭代计算下去并进行归纳，可得迭代点列为

$$x^k = \left(\frac{1}{3}\right)^k \begin{pmatrix} 2 \\ (-1)^k \end{pmatrix}, \quad k \in \{0, 1, 2, \cdots\}.$$

容易看到：当$k \to \infty$时, 有$x^k \to x^* = 0$, 即算法产生的点列收敛于问题的最优解. □

3.2 最速下降法

由例3.2.1不难看到：采用精确一维线搜索的最速下降算法最多具有线性收敛速度，其收敛速度是很慢的，这似乎与“最速“二字相矛盾. 其实不然，因为沿最速下降方向迭代快只是反映了目标函数在迭代点的局部邻域内沿最速下降方向进行迭代目标函数的下降是最快的，并不意味着整体迭代中目标函数的下降也是最快的. 事实上，采用精确一维线搜索的最速下降算法的整体迭代并不快，这可以由以下分析看出：由精确一维搜索公式容易得到

$$(d^{k+1})^\top d^k = 0, \quad k \in \{0, 1, 2, \cdots\}.$$

这表明：在相继两次迭代中，迭代方向是相互正交的. 因而，采用精确一维线搜索的最速下降算法逼近极小值点的路线是锯齿形的，并且越靠近极小值点步长越小，即越走越慢.

3.2 最速下降法 – 作业

3.4 给定对称正定矩阵$G \in \mathbb{R}^{n\times n}$, 向量$b \in \mathbb{R}^n$以及实数$c \in \mathbb{R}$. 定义二次函数$f(x) = \frac{1}{2}x^\top Gx + b^\top x + c$. 沿射线$x^k + \lambda d^k$ $(d^k \neq 0)$进行一维线搜索：$\min_{\lambda\geq 0} f(x^k + \lambda d^k)$. 试证明：搜索步长为

$$\lambda_k = -\frac{g_k^\top d^k}{(d^k)^\top G d^k}.$$

3.5 试证明: 在用精确一维线搜索的最速下降算法中，相邻两次迭代中的迭代方向是相互正交的.

3.8 设函数$f(x) = \frac{1}{2}x^\top Ax + b^\top x + c$，其中$A$为对称正定矩阵，$b \in \mathbb{R}^n$且$c \in \mathbb{R}$. 又设$x^1$可表示为

$$x^1 = x^* + \mu d,$$

其中x^*是函数f的极小值点, d是矩阵A对应于特征值λ的特征向量. 试证明:

(1) $\nabla f(x^1) = \mu\lambda d$;

(2) 若从x^1出发, 沿着最速下降迭代方向作精确一维线搜索, 则只需一步迭代可达到极小值点x^*.