引言:无约束优化算法

$$\min_{x\in\mathbb{R}^n} \quad f(x).$$

- 线搜索
  1. 先确定搜索方向:梯度类算法、次梯度算法、牛顿算法、拟牛顿算法等.
  2. 按准则进行近似搜索.
- 信赖域
  1. 主要针对$f(x)$ 二阶可微的情形, 在一个给定的区域内使用二阶模型近似原问题.
  2. 不断直接求解该二阶模型从而找到最优值点.

线搜索算法:盲人下山

- 求解$f(x)$ 的最小值点如同盲人下山, 无法一眼望知谷底, 而是:
  1. 首先确定下一步该向哪一方向行走.
  2. 再确定沿着该方向行走多远后停下以便选取下一个下山方向.
- 线搜索类算法的数学表述:

$$x^{k+1} = x^k + \alpha_k d^k.$$

我们称d^k 为迭代点x^k 处的**搜索方向**, α_k 为相应的**步长**.这里要求d^k 是一个**下降方向**, 即$(d^k)^{\mathrm{T}} \nabla f(x^k) < 0$.

- 线搜索类算法的关键是如何选取一个好的方向$d^k \in \mathbb{R}^n$ 以及合适的步长α_k.

α_k 的选取:精确线搜索算法

- 选取d^k 的方法千差万别, 但选取α_k 的方法却非常相似.
- 首先构造一元辅助函数

$$\phi(\alpha) = f(x^k + \alpha d^k),$$

其中d^k 是给定的下降方向, $\alpha > 0$ 是该辅助函数的自变量.

- 线搜索的目标是选取合适的α_k 使得$\phi(\alpha_k)$ 尽可能减小. 这要求:
  1. α_k 应该使得f 充分下降
  2. 不应在寻找α_k 上花费过多的计算量
- 一个自然的想法是寻找α_k 使得

$$\alpha_k = \operatorname*{argmin}_{\alpha>0} \phi(\alpha),$$

即α_k 为最佳步长. 这种线搜索算法被称为**精确线搜索算法**

- 选取α_k 通常需要很大计算量, 在实际应用中较少使用

α_k 的选取:非精确线搜索算法

- 不要求α_k 是$\phi(\alpha)$ 的最小值点，仅要求$\phi(\alpha_k)$ 满足某些不等式. 这种线搜索方法被称为**非精确线搜索算法**.
- 选取α_k 需要满足的要求被称为**线搜索准则**
- 线搜索准则的合适与否直接决定了算法的收敛性，若选取不合适的线搜索准则将会导致算法无法收敛
- 例如，若只要求选取的步长满足迭代点处函数值单调下降，则函数值$f(x^k)$ 的下降量可能不够充分，导致算法无法收敛到极小值点

例子:不合适的线搜索准则导致无法收敛

考虑一维无约束优化问题

$$\min_x \quad f(x) = x^2,$$

迭代初始点$x^0 = 1$. 由于问题是一维的, 下降方向只有$\{-1, +1\}$ 两种. 我们选取$d^k = -\mathrm{sign}(x^k)$, 且只要求选取的步长满足迭代点处函数值单调下降, 即$f(x^k + \alpha_k d^k) < f(x^k)$. 考虑选取如下两种步长:

$$\alpha_{k,1} = \frac{1}{3^{k+1}}, \quad \alpha_{k,2} = 1 + \frac{2}{3^{k+1}},$$

通过简单计算可以得到

$$x_1^k = \frac{1}{2}\left(1 + \frac{1}{3^k}\right), \quad x_2^k = \frac{(-1)^k}{2}\left(1 + \frac{1}{3^k}\right).$$

显然, 序列$\{f(x_1^k)\}$ 和序列$\{f(x_2^k)\}$ 均单调下降, 但序列$\{x_1^k\}$ 收敛的点不是极小值点, 序列$\{x_2^k\}$ 则在原点左右振荡, 不存在极限

Armijo准则

定义 (Armijo 准则)

设d^k 是点x^k 处的下降方向，若

$$f(x^k + \alpha d^k) \leq f(x^k) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k,$$

则称步长α 满足***Armijo* 准则**，其中$c_1 \in (0, 1)$ 是一个常数.

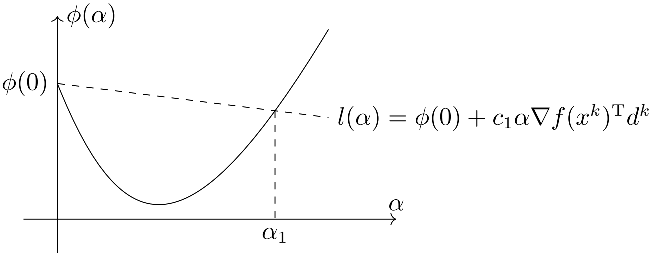

Figure: Armijo 准则

Armijo准则:评注

- 引入Armijo 准则的目的是保证每一步迭代充分下降
- Armijo 准则有直观的几何含义, 它指的是点$(\alpha, \phi(\alpha))$ 必须在直线

$$l(\alpha) = \phi(0) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k$$

  的下方, 上图中区间$[0, \alpha_1]$ 中的点均满足Armijo 准则
- 参数c_1 通常选为一个很小的正数, 例如$c_1 = 10^{-3}$, Armijo准则非常容易得到满足
- Armijo 准则需要配合其他准则以保证迭代的收敛性, 因为$\alpha = 0$ 显然满足Armijo准则, 此时迭代序列中的点固定不变

回退法:以Armijo准则为例

- 给定初值$\hat{\alpha}$, 回退法通过不断以指数方式缩小试探步长, 找到第一个满足Armijo 准则的点
- 回退法选取
$$\alpha_k = \gamma^{j_0}\hat{\alpha},$$
其中
$$j_0 = \min\{j = 0, 1, \cdots \mid f(x^k + \gamma^j \hat{\alpha} d^k) \le f(x^k) + c_1 \gamma^j \hat{\alpha} \nabla f(x^k)^{\mathrm{T}} d^k\},$$
参数$\gamma \in (0, 1)$ 为一个给定的实数

Algorithm 1 线搜索回退法

1: 选择初始步长$\hat{\alpha}$, 参数$\gamma, c \in (0, 1)$. 初始化$\alpha \leftarrow \hat{\alpha}$.
2: **while** $f(x^k + \alpha d^k) > f(x^k) + c\alpha \nabla f(x^k)^{\mathrm{T}} d^k$ **do**
3: 令$\alpha \leftarrow \gamma\alpha$.
4: **end while**
5: 输出$\alpha_k = \alpha$.

回退法:以Armijo准则为例

- 该算法被称为回退法是因为α 的试验值是由大至小的，它可以确保输出的α_k 能尽量地大
- 算法1不会无限进行下去，因为d^k 是一个下降方向，当α 充分小时，Armijo 准则总是成立的
- 实际应用中我们通常也会给α 设置一个下界，防止步长过小

Goldstein 准则

- 为了克服Armijo 准则的缺陷，我们需要引入其他准则来保证每一步的α^k 不会太小
- Armijo准则只要求点$(\alpha, \phi(\alpha))$ 必须处在某直线下方，我们也可使用相同的形式使得该点必须处在另一条直线的上方．这就是Armijo-Goldstein 准则，简称Goldstein 准则

定义 (Goldstein 准则)

设d^k 是点x^k 处的下降方向，若

$$f(x^k + \alpha d^k) \le f(x^k) + c\alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{1a}$$

$$f(x^k + \alpha d^k) \ge f(x^k) + (1-c)\alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{1b}$$

则称步长α 满足***Goldstein* 准则**，其中$c \in \left(0, \frac{1}{2}\right)$．

Goldstein准则

Goldstein 准则有直观的几何含义，它指的是点$(\alpha, \phi(\alpha))$ 必须在两条直线

$$l_1(\alpha) = \phi(0) + c\alpha\nabla f(x^k)^{\mathrm{T}} d^k,$$
$$l_2(\alpha) = \phi(0) + (1-c)\alpha\nabla f(x^k)^{\mathrm{T}} d^k$$

之间. 区间$[\alpha_1, \alpha_2]$ 中的点均满足Goldstein 准则. 同时我们也注意到Goldstein 准则确实去掉了过小的α.

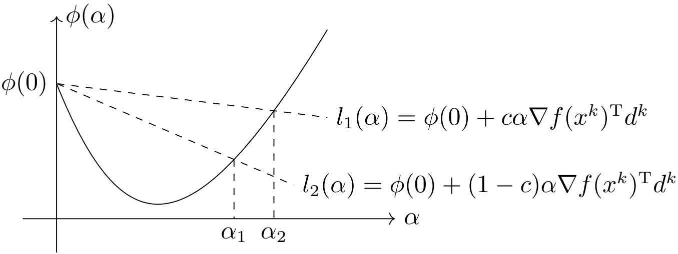

Figure: Goldstein 准则

- Goldstein 准则能够使得函数值充分下降, 但是它可能避开了最优的函数值. 上页图中的一维函数$\phi(\alpha)$ 的最小值点并不在满足Goldstein 准则的区间$[\alpha_1, \alpha_2]$ 中
- 在Wolfe准则中, 第一个不等式即是Armijo 准则, 而第二个不等式则是Wolfe 准则的本质要求

定义 (Wolfe 准则)

设d^k 是点x^k 处的下降方向, 若

$$f(x^k + \alpha d^k) \le f(x^k) + c_1 \alpha \nabla f(x^k)^{\mathrm{T}} d^k, \tag{2a}$$

$$\nabla f(x^k + \alpha d^k)^{\mathrm{T}} d^k \ge c_2 \nabla f(x^k)^{\mathrm{T}} d^k, \tag{2b}$$

则称步长α 满足***Wolfe* 准则**, 其中$c_1, c_2 \in (0, 1)$ 为给定的常数且$c_1 < c_2$.

Wolfe准则

- $\nabla f(x^k+\alpha d^k)^{\mathrm{T}}d^k$ 恰好就是$\phi(\alpha)$ 的导数, Wolfe 准则实际要求$\phi(\alpha)$ 在点α 处切线的斜率不能小于$\phi'(0)$ 的c_2 倍
- $\phi(\alpha)$ 的极小值点α^* 处有$\phi'(\alpha^*)=\nabla f(x^k+\alpha^* d^k)^{\mathrm{T}}d^k=0$, 因此$\alpha^*$ 永远满足条件二. 而选择较小的c_1 可使得α^* 同时满足条件一, 即Wolfe 准则在绝大多数情况下会包含线搜索子问题的精确解

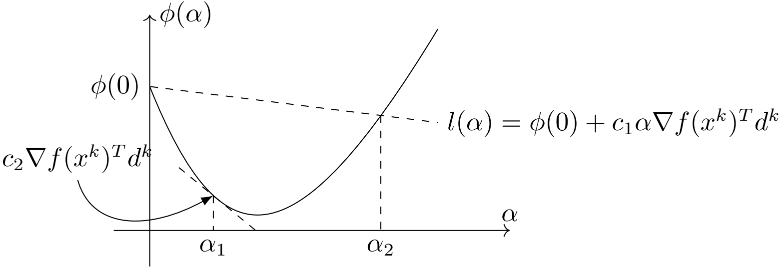

Figure: Wolfe 准则

非单调线搜索准则

定义 (Grippo)

设d^k 是点x^k 处的下降方向, $M>0$ 为给定的正整数. 以下不等式可作为一种线搜索准则:

$$f(x^k+\alpha d^k) \le \max_{0\le j\le \min\{k,M\}} f(x^{k-j}) + c_1\alpha\nabla f(x^k)^{\mathrm{T}} d^k,$$

其中$c_1\in(0,1)$ 为给定的常数.

- 该准则和Armijo 准则非常相似, 区别在于Armijo 准则要求下一次迭代的函数值$f(x^{k+1})$ 相对于本次迭代的函数值$f(x^k)$ 有充分下降, 而该准则只需要下一步函数值相比前面至多M 步以内迭代的函数值有下降就可以了
- 这一准则的要求比Armijo 准则更宽, 它也不要求$f(x^k)$ 的单调性

非单调线搜索准则

另一种非单调线搜索准则的定义更加宽泛.

定义 (Zhang, Hager)

设d^k 是点x^k 处的下降方向, $M > 0$ 为给定的正整数. 以下不等式可作为一种线搜索准则：

$$f(x^k + \alpha d^k) \leq C^k + c_1 \alpha \nabla f(x^k)^\mathrm{T} d^k,$$

其中C^k 满足递推式$C^0 = f(x^0), C^{k+1} = \frac{1}{Q^{k+1}}(\eta Q^k C^k + f(x^{k+1}))$, 序列$\{Q^k\}$ 满足$Q^0 = 1, Q^{k+1} = \eta Q^k + 1$, 参数$\eta, c_1 \in (0, 1)$.

- 变量C^k 实际上是本次搜索准则的参照函数值, 即充分下降性质的起始标准
- 下一步的标准C^{k+1} 则是函数值$f(x^{k+1})$ 和C^k 的凸组合, 并非仅仅依赖于$f(x^{k+1})$, 而凸组合的两个系数由参数η 决定
- 当$\eta = 0$ 时, 此准则就是Armijo 准则

回退法的优缺点

- 只要修改一下算法的终止条件，回退法就可以被用在其他线搜索准则之上，它是最常用的线搜索算法之一.
- 然而，回退法的缺点也很明显：
  1. 它无法保证找到满足Wolfe 准则的步长，即条件二不一定成立，但对一些优化算法而言，找到满足Wolfe 准则的步长是十分必要的
  2. 回退法以指数的方式缩小步长，因此对初值$\hat{\alpha}$ 和参数γ 的选取比较敏感，当γ 过大时每一步试探步长改变量很小，此时回退法效率比较低，当γ 过小时回退法过于激进，导致最终找到的步长太小，错过了选取大步长的机会

基于多项式插值的线搜索算法

- 设初始步长$\hat{\alpha}_0$ 已给定，如果经过验证，$\hat{\alpha}_0$ 不满足Armijo 准则，下一步就需要减小试探步长
- 基于$\phi(0), \phi'(0), \phi(\hat{\alpha}_0)$ 这三个信息构造一个二次插值函数$p_2(\alpha)$
- 寻找二次函数$p_2(\alpha)$ 满足

$$p_2(0) = \phi(0), \quad p_2'(0) = \phi'(0), \quad p_2(\hat{\alpha}_0) = \phi(\hat{\alpha}_0).$$

  由于二次函数只有三个参数，以上三个条件可以唯一决定$p_2(\alpha)$
- $p_2(\alpha)$ 的最小值点恰好位于$(0, \hat{\alpha}_0)$ 内
- 取$p_2(\alpha)$ 的最小值点$\hat{\alpha}_1$ 作为下一个试探点，利用同样的方式不断递归下去直至找到满足Armijo 准则的点
- 基于插值的线搜索算法可以有效减少试探次数，但仍然不能保证找到的步长满足Wolfe 准则

提纲

Zoutendijk定理

定理 (Zoutendijk定理)

考虑一般的迭代格式$x^{k+1} = x^k + \alpha_k d^k$, 其中$d^k$ 是搜索方向, α_k 是步长, 且在迭代过程中*Wolfe* 准则满足. 假设目标函数f 下有界、连续可微且梯度L -利普希茨连续, 即

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|, \quad \forall\, x, y \in \mathbb{R}^n,$$

那么

$$\sum_{k=0}^{\infty} \cos^2\theta_k \|\nabla f(x^k)\|^2 < +\infty,$$

其中$\cos\theta_k$ 为负梯度$-\nabla f(x^k)$ 和下降方向d^k 夹角的余弦, 即

$$\cos\theta_k = \frac{-\nabla f(x^k)^{\mathrm{T}} d^k}{\|\nabla f(x^k)\| \|d^k\|}.$$

这个不等式也被称为***Zoutendijk*** 条件.

Zoutendijk定理的证明

- 由Wolfe准则知$\nabla f(x^{k+1})^\mathrm{T} d^k \geq c_2 f(x^k)^\mathrm{T} d^k$, 故

$$\left(\nabla f(x^{k+1}) - \nabla f(x^k)\right)^\mathrm{T} d^k \geq (c_2 - 1)\nabla f(x^k)^\mathrm{T} d^k.$$

- 由柯西不等式和梯度L-利普希茨连续性质,

$$\left(\nabla f(x^{k+1}) - \nabla f(x^k)\right)^\mathrm{T} d^k \leq \|\nabla f(x^{k+1}) - \nabla f(x^k)\| \|d^k\| \leq \alpha_k L \|d^k\|^2.$$

- 结合上述两式可得

$$\alpha_k \geq \frac{c_2 - 1}{L} \frac{\nabla f(x^k)^\mathrm{T} d^k}{\|d^k\|^2}.$$

- 由Wolfe准则的条件一知$f(x^{k+1}) \leq f(x^k) + c_1 \alpha_k \nabla f(x^k)^\mathrm{T} d^k$, 注意到$\nabla f(x^k)^\mathrm{T} d^k < 0$, 将上式代入得

$$f(x^{k+1}) \leq f(x^k) + c_1 \frac{c_2 - 1}{L} \frac{\left(\nabla f(x^k)^\mathrm{T} d^k\right)^2}{\|d^k\|^2}.$$

Zoutendijk定理的证明

- 根据θ_k 的定义，此不等式可等价表述为

$$f(x^{k+1}) \le f(x^k) + c_1 \frac{c_2 - 1}{L} \cos^2\theta_k \|\nabla f(x^k)\|^2.$$

- 再关于k 求和，我们有

$$f(x^{k+1}) \le f(x^0) - c_1 \frac{1 - c_2}{L} \sum_{j=0}^{k} \cos^2\theta_j \|\nabla f(x^j)\|^2.$$

- 又因为函数f 是下有界的，且由$0 < c_1 < c_2 < 1$ 可知$c_1(1 - c_2) > 0$，因此当$k \to \infty$ 时，

$$\sum_{j=0}^{\infty} \cos^2\theta_j \|\nabla f(x^j)\|^2 < +\infty.$$

Zoutendijk 定理刻画了线搜索准则的性质，配合下降方向d^k 的选取方式我们可以得到最基本的收敛性.

推论 (线搜索算法的收敛性)

对于迭代法$x^{k+1} = x^k + \alpha_k d^k$, 设$\theta_k$ 为每一步负梯度$-\nabla f(x^k)$ 与下降方向d^k 的夹角，并假设对任意的k, 存在常数$\gamma > 0$, 使得

$$\theta_k < \frac{\pi}{2} - \gamma,$$

则在*Zoutendijk*定理成立的条件下，有

$$\lim_{k\to\infty} \nabla f(x^k) = 0.$$

线搜索算法收敛性的证明

- 假设结论不成立，即存在子列$\{k_l\}$ 和正常数$\delta > 0$，使得

$$\|\nabla f(x^{k_l})\| \geq \delta, \quad l = 1, 2, \cdots .$$

- 根据θ_k 的假设，对任意的k，

$$\cos \theta_k > \sin \gamma > 0.$$

- 我们仅考虑Zoutendijk条件中第k_l 项的和，有

$$\begin{aligned}\sum_{k=0}^{\infty} \cos^2 \theta_k \|\nabla f(x^k)\|^2 &\geq \sum_{l=1}^{\infty} \cos^2 \theta_{k_l} \|\nabla f(x^{k_l})\|^2 \\ &\geq \sum_{l=1}^{\infty} (\sin^2 \gamma) \cdot \delta^2 \to +\infty,\end{aligned}$$

- 这显然和Zoutendijk定理矛盾。因此必有

$$\lim_{k\to\infty} \nabla f(x^k) = 0.$$

收敛性分析:评注

- 线搜索算法收敛性建立在Zoutendijk 条件之上, 它的本质要求是$\theta_k < \frac{\pi}{2} - \gamma$, 即每一步的下降方向$d^k$ 和负梯度方向不能趋于正交.
- 几何直观: 当下降方向d^k 和梯度正交时, 根据泰勒展开的一阶近似, 目标函数值$f(x^k)$ 几乎不发生改变. 因此我们要求d^k 与梯度正交方向夹角有一致的下界.
- 不涉及算法收敛速度的分析, 因为算法收敛速度极大地取决于d^k 的选取.