Linear Search

given a point x^k
1. find a descent direction d^k
2. find a stepsize α^k

$$x^{k+1} = x^k + \alpha^k d^k$$

假设在某点，寻找方向 direction 和步长 stepsize 使得最小，如果确定则只需要解决一维最优化问题就可以找到下一个搜索点.
首先选择方向 d^k 通过解决一维最优化问题找到步长 α^k

Descent Direction d^k , $f \in C^1(\mathbb{R}^n), x \in \mathbb{R}^n$ A $d \in \mathbb{R}^n$ is said to be a **descent direction** of f at x $\Longleftarrow$ $[\nabla f(x)]^T d < 0$.

- More generally, if $D \succeq 0$, then $d = -D\nabla f(x)$ is a descent direction.
  $\Longleftrightarrow$ 任一方向 d 只要能分解成一个正定矩阵 D 和负梯度 $-\nabla f(x)$ 的乘积，那么这个方向一定是下降方向
  Proof: $[\nabla f(x)]^T \cdot \big(-D\nabla f(x)\big) = -\big(\nabla f(x)^T D \nabla f(x)\big)$
  $\because \nabla f(x) \neq 0, \therefore < 0$

是不是下降方向就看：$[\nabla f(x)]^T d < 0$

At an x that is **not stationary**,

d = $-\nabla f(x)$ is a descent direction?

yes. $[\nabla f(x)]^T \cdot -\nabla f(x) = -\|\nabla f(x)\|_2^2 < 0$

is the Newton direction $-[\nabla^2 f(x)]^{-1}\nabla f(x)$ a **descent direction?**

A: Not necessary. $\because d = [\nabla^2 f(x)]^{-1}\nabla f(x), \therefore D = [\nabla^2 f(x)]^{-1}\text{?positive definite} \begin{cases} \in & \text{yes} \\ \notin & \text{no} \end{cases}$

	$d^k = -D^k\nabla f(x^k), D \succeq 0$	descent direction	
牛顿法	$-[\nabla^2 f(x^k)]^{-1}\nabla f(x^k)$	一阶导=0 $d^k = -[\nabla^2 f(x^k)]^{-1} \cdot \nabla f(x^k)$ (not necessary) $\text{only } [\nabla^2 f(x^k)] \succeq 0$	仅仅依赖函数值和梯度的信息(即一阶信息)
最速下降法	$-\nabla f(x^k)$	负梯度方向 $d^k = -1I \cdot \nabla f(x^k), \checkmark$	
拟牛顿法	$-B^k\nabla f(x^k)$	$d^k = -B^k \cdot \nabla f(x^k)$ (not necessary) $\text{only } B^k \succeq 0$	
共轭梯度法			

Newton’s method 牛顿迭代法

方法本身：求解非线性方程 $g(x) = 0$ 的近似根 x^*
在 Descent Direction 上的应用：求解 $新 \cdot g(x) = \nabla f(x^*) = 0$

使用**函数的泰勒级数**的前面几项来寻找方程的根。

方法本身

- **背景**
  多数方程不存在求根公式，因此求精确根非常困难，甚至不可能，从而寻找方程的近似根就显得特别重要。方程用二次函数的形式表示出来，我们就可以通过上面的办法大踏步的前进了！由此我们祭出将任意N阶可导函数化为N次多项式的神器：**N阶泰勒展开**
- **思路**
  设 x^* 是 $g(x) = 0$ 的近似根，将 $g(x)$ 在 x^k 附近**用一阶泰勒多项式近似**

$$g(x) = g(x^k) + \nabla g(x^k)^T \cdot (x - x^k) + o(|x - x_0|)$$

  舍去高阶项：$g(x) = g(x^k) + \nabla g(x^k)^T \cdot (x - x^k)$
  将近似根代入：

$$g(x^*) = g(x^k) + \nabla g(x^k)^T \cdot (x^* - x^k) = 0 \tag{1}$$

$$x^* = x^k - \frac{g(x^k)}{g'(x^k)} \tag{2}$$

  不能一步得到，所以需要迭代 ∴迭代公式：$x^{k+1} = x^k - \dfrac{f(x^k)}{f'(x^k)}$

> **Note**
>
> i. 先随机选一个点，
> ii. 然后求出$f(x)$在该点的切线。
> iii. 该切线与x轴相交的点为下一次迭代的值。
> 直至逼近$f(x)=0$的点。

- **停止标准**
  - $|x_{k+1}-x_k|<\epsilon_1$
  - $|f(x)|<\epsilon_2$: $f(x)$很小，小于精度，不能保证x的精度
    局限性：对于某些特殊函数，小区间急速变化
- **几何本质**
  在原函数的某一点处用一个二次函数近似原函数，然后用这个二次函数的极小值点作为原函数的下一个迭代点。 基于当前迭代点的梯度信息进行搜索方向的选择的，牛顿法是通过Hessian矩阵在梯度上进行线性变换得到搜索方向

收敛

fast local convergence 快速的局部收敛 + Quadratic convergence 二阶收敛性
$\iff$ 牛顿法**靠近最优点**时是**二次**收敛的

$\begin{cases} g\in C^2(\mathbb{R}) \\ g(x^*)=0 \\ g'(x^*)\neq 0. \end{cases} \implies \exists\varepsilon>0, |x^0-x^*|<\varepsilon.$ And with Newton's iterate: $x^{k+1}=x^k-\dfrac{g(x^k)}{g'(x^{k+1})}$is well defined.

$$\implies \exists M>0, |x^{k+1}-x^k|\leq M\|x^k-x^*\|_2$$

> **x^0 选的好，那么牛顿法很好用，收敛速度很快，每次迭代之后，如果 x^0 的初始化足够接近一个好的解决方案，那么牛顿方法的定义很好，收敛速度也非常快：每次迭代的正确数字数量大约翻一番。（甚至步长都不需要确定）。所以牛顿法对函数在迭代点处的信息利用更加充分，直观来看，相比于梯度下降法，函数足够正则的情况下牛顿法迭代得更加准确，收敛速率也会更快。**
>
> 当x在以 x^* 为原点，ε 为区间的邻域内进行迭代，所有迭代过来的 x^k 都以二次收敛的速度收敛于 x^*【局部の二次の收敛】，其中$M=\frac{\tau}{2\delta}$

失效

1. x^0选的不好，离x^*很远，$\exists x^k\in(x^0,x^*), g'(x^k)=0$，几何上没有升降的空间，运算上分母为0失效（更远了）
2. due to cycling

在 Descent Direction 上的应用

目标：新 $\cdot g(x) = \nabla f(x^*) = 0$

$\exists x^{k+1}, \nabla f(x^{k+1}) = 0 \implies \nabla f(x) = \nabla f(x^k) + \nabla^2 f(x^k)(x - x^k)$

迭代方程：

参照可得

$$g(x^*) = g(x^k) + \nabla g(x^k)^T \cdot (x^* - x^k) = 0 \tag{1}$$
$$x^* = x^k - \frac{g(x^k)}{g'(x^k)} \tag{2}$$

$$\nabla f(x^*) = \nabla f(x^k) + \nabla^2 f(x^k)^T \cdot (x^* - x^k) = 0 \qquad \text{(牛顿方程)} \tag{3}$$
$$x^* = x^k - \frac{g(x^k)}{g'(x^k)} \tag{4}$$

1. $x^* = x^k - \dfrac{g(x^k)}{g'(x^k)} \qquad (2)$

$$\downarrow$$
$$x^* = x^k - \frac{\nabla f(x^k)}{\nabla^2 f(x^k)}, \qquad d^k = -\frac{\nabla f(x^k)}{\nabla^2 f(x^k)}$$
$$\alpha \equiv 1 \qquad \text{(经典牛顿法)}$$

要求：

1. $\forall k, \nabla^2 f(x^k)$可逆 $\iff \in$ sigular 非奇异矩阵 二阶可微函数
2. 计算$\dfrac{\nabla f(x^k)}{\nabla^2 f(x^k)}$简单

💡 For k=0,1,2..., update $x^{k+1} = x^k - \dfrac{\nabla f(x^k)}{\nabla^2 f(x^k)}$

不要去求解$(\nabla^2 f)^{-1}$然后再乘，而是把$d = \dfrac{\nabla f(x^k)}{\nabla^2 f(x^k)}$，解$\nabla^2 f(x^k) d = \nabla f(x^k)$

⭐ 牛顿法也只是找到一阶导为0，也就是说朝着极值的d^k，不一定是函数值的下降方向，还要verify **通过$\nabla^2 f(x^*)$去验证$X = x^*$是否local minimizer**

⭐ 可以用更少的迭代次数大踏步地前进，并且前进的方向也更趋向于函数的全局最优解（即最值而非极值点），同时也能够摆脱上面梯度下降法中确定α的痛苦

Here we discuss just the local rate-of-convergence properties of Newton's method. We know that for all x in the vicinity of a solution point x∗ such that∇2 f(x∗) is positive definite, the Hessian ∇2 f(x) will also be positive definite. Newton's method will be well defined in this region and will converge quadratically, provided that the step lengths αk are eventually always 1.

缺点

1. 每一步迭代需要求解一个 *n* 维线性方程组，这导致在高维问题中计算量很大.海瑟矩阵 $\nabla^2 f\left(x^k\right)$ 既不容易计算又不容易储存.
2. $\nabla^2 f(x^k)$ 不正定时，由牛顿方程给出的解 *dk* 的性质通常比较差.例如可以验证当海瑟矩阵正定时，*dk* 是一个下降方向，而在其他情况下 *dk* 不一定为下降方向.

Steepest descent 最速下降法

（梯度）**某一点处的梯度方向是函数值增长最快的方向**

💡 Steepest descent with exact line search： 希望得到一个**在该点下降最快的方向**，来使得我们的迭代过程尽可能的高效。 **梯度的反方向就是函数值下降最快的方向**。
计算量大、计算时间长是最速下降法的一个缺点。

存在性证明：负梯度方向就是下降最快方向

(Taylor展开)

$$\because f \in C^1(\mathbb{R}),\ \text{we have:}$$
$$\exists \xi \in \{x+td : t \in (0,1))\}$$
$$f(x+d) = f(x) + [\nabla f(x)]^T d + [\nabla f(\xi) - \nabla f(x)]^T d$$
其中本来二阶导的地方：$\frac{1}{2}d^T\nabla^2 f(\xi) = [\nabla f(\xi) - \nabla f(x)]^T; \xi \text{ depends on } d$

如果

$$\nabla f(x) \neq 0 \iff x \text{ is not a stationary}$$

then，我们取

$$d = -\alpha\nabla f(x) \text{ for some } \alpha > 0,$$

- 为什么 αd 取 $-\alpha\nabla f(x)$ for some $\alpha > 0$,(此处的d范围更缩小一点，指方向
  在没有给定d之前：$f(x+d) = f(x) + \alpha\nabla f(x)^T d$
  $\because$ 我们是给已给函数f和迭代点x，$\nabla f(x)^T \in$ 常量
  $f(x+d) = f(x) + \alpha\nabla f(x)^T d$ 是关于α的函数，要随着α增加而减小，且减少得尽可能快，
  $\therefore d^k = \arg\min_{d^k} \dfrac{\partial f}{\partial \alpha} = \arg\max_{d^k} -\dfrac{\partial f}{\partial \alpha}$
  Recall： Cauchy不等式

$$-\frac{\partial f}{\partial \alpha}=-\nabla f(x)^T d=(-\nabla d(x),d)=\Vert-\nabla f(x)\Vert\cdot\Vert d\Vert\cdot\cos\theta_k$$ θ_k就是搜索方向d和负梯度方向的角度，当$\theta_k=0^\circ$ 时，最大，所以就是最速

$\therefore d=-\nabla f(x)$

then,

$f(x+d)=f(x)+[\nabla f(x)]^Td+[\nabla f(\xi)-\nabla f(x)]^Td\

f(x-\alpha\nabla f(x))=f(x)-\alpha\Vert\nabla f(x)\Vert_2^2-\alpha([\nabla f(\xi)-\nabla f(x)]^T\cdot\nabla f(x))\ \small{\text{其中第2项：}\nabla f(x)\cdot \alpha\nabla f(x)=\alpha\Vert\nabla f(x)\Vert_2^2；\xi \text{ depends on }\alpha}$

所以

第3项：$\alpha([\nabla f(\xi)-\nabla f(x)]^T\cdot\nabla f(x))=0$

$f(x-\alpha\nabla f(x))=f(x)-\alpha\Vert\nabla f(x)\Vert_2^2$

$$\begin{aligned}&\therefore\ \text{for sufficiently small } \alpha>0,\\ &\quad f(x-\alpha\nabla f(x))<f(x)\qquad (\text{是下降方向})\end{aligned}$$

Untitled

我们得出：

$-\mathbf{\nabla f(x)}$ is called the steepest descent direction

Steepest descent with exact line search

Start at $x^0\in R^n$. For each $k=0,1,\dots$

1. Set $d^k=-\nabla f(x^k)\qquad$ (the search direction)
2. pick $\alpha_k\in\arg\min\{f(x^k+\alpha d^k):\alpha>0\}\qquad$ (step size | learning rate)

其中 α_k is chosen according to the exact line search criterion 通过**精确线搜索**确定步长（隐含地假定，对于精确线搜索，存在一个最小化器α_k。）

Steepest descent with constant stepsize

Let $f\in C^2(\mathbb{R}^n)$, $\inf f>-\infty$. Suppose that there $\exists L>0$ so that

$$L\geq\Vert\nabla^2 f(x)\Vert_2,\forall x$$

fix any $\gamma\in(0,2)$ and consider the sequence generated as

$$x^{k+1}=x^k-\frac{\gamma}{L}\nabla f(x^k)$$

then any accumulation point of $\{x^k\}$ is a stationary point of f

步长保守 the constant stepsize，下降缓慢 potentially slow

- proof：

Conjugate gradient method 共轭梯度法

Flops per iteration is *O*(*n*2);

It converges in at most *n* steps;•

It keeps track of *O*(1) vectors of dimension *n* per iteration.

> idea: Modify the steepest descent direction to fit the (ellipse) geometry.

$$\text{Projection onto v} \quad \text{Let } u \in \mathbb{R}^n, v \in \mathbb{R}^n \setminus \{0\}.$$
$$\text{The projection of u onto v} \iff \operatorname{proj}_v(u) := \frac{u^T v}{\|v\|_2^2} v;$$
$$\|w\|_2 = \|u\|_2 \cos\theta = \|u\|_2 \frac{u^T v}{\|u\|_2 \|v\|_2} = \frac{u^T v}{\|v\|_2};$$
$$\text{Unit vector along w is } \frac{v}{\|v\|_2}$$

Untitled

$$\text{Gram-Schmidt process} \quad \text{Given a set of linearly independent vectors } \{v^0, ..., v^k\} \subset \mathbb{R}^n. \text{ Set } w^0 = v^0 \text{ and for each } j = 1, ..., k$$
$$w^k = v^k - \sum_{j=0}^{k-1} \frac{(v^k)^T w^j}{\|w^j\|_2^2} w^j; \qquad \forall i, w^i \neq 0; \forall i \neq j, (w^i)^T w^j = 0$$
$$\operatorname{Span}\{v^0, ..., v^k\} = \operatorname{Span}\{w^0, ..., w^i\}$$

$$\text{Generalized Gram-Schmidt process} \quad \text{Given } A \in \mathbb{R}^n, A \succ 0, \text{ and a set of linearly independent vectors } \{v^0, ..., v^k\} \subset \mathbb{R}^n.$$
$$\text{Set } w^0 = v^0 \text{ and for each } j = 1, ..., k$$
$$w^k = v^k - \sum_{j=0}^{k-1} \frac{(v^k)^T A w^j}{(w^j)^T A w^j} w^j; \qquad \forall i, w^i \neq 0; \forall i \neq j, (w^i)^T A w^j = 0$$
$$\operatorname{Span}\{v^0, ..., v^k\} = \operatorname{Span}\{w^0, ..., w^i\}$$

Conjugate gradient method: Conceptual version

Start at $*x^0 \in \mathbb{R}^{n*}$ and $*d^0 = -\nabla f(x^0) = b - Ax^0*$.

For each *k* = 0,1,2,...,

- If $*d^k = 0*$, terminate.
- Pick α_k **so that: $\alpha_k \in \arg\min\{f(x^k + \alpha d^k) : \alpha \geq 0\}$.

- Set $x^{k+1} = x^k + \alpha_k d^k, d^{k+1} = -\nabla f(x^{k+1}) - \sum_{j=0}^{k} \frac{[-\nabla f(x^{k+1})]^T A d^j}{(d^j)^T A d^j}$

Proof of correctness

Conjugate gradient method: Formal version

Start at $*x^0 \in \mathbb{R}^{n*}$ and $*d^0 = -\nabla f(x^0) = b - Ax^0*$.

For each *k* = 0,1,2,...,

- If $*d^k \;=\; 0*$, terminate.
- Pick α_k **so that: $\alpha_k \in \arg\min\{f(x^k + \alpha d^k) : \alpha \geq 0\}$.
- Set $x^{k+1} = x^k + \alpha_k d^k, d^{k+1} = -\nabla f(x^{k+1}) - \frac{\|\nabla f(x^{k+1})\|_2^2}{\|\nabla f(x^k)\|_2^2} d^k$

Proof of correctness:

Conjugate gradient method: Actual version

1. **迭代过程：**
   Start at $*x^0 \in \mathbb{R}^{n*}$ and $*r^0 = d^0 = -\nabla f(x^0) = b - Ax^0*$.
   For each *k* = 0,1,2,...,
   - If $*\|d^k\|$ $\text{is below a tolerance}*$, terminate.
   - $\alpha_k = \frac{(r^k)^T r^k}{(d^k)^T A d^k}$, $x^{k+1} = x^k + \alpha_k d^k$, $r^{k+1} = r^k - \alpha_k A d^k \qquad \text{(excat line search)}.$
   - ParseError: KaTeX parse error: Unexpected end of input in a macro argument, expected '}' at end of input: …ad\text{(Updated^{k+1}ParseError: KaTeX parse error: Expected 'EOF', got '}' at position 2:)}
2. 优点
   - One matrix-vector multiplication per iteration if $*Ad^{k*}$ is saved.
   - Keeping track of four vectors, $*x^k, r^k, d^k, Ad^{k*}$ saved.

Proof of correctness:

Newton-CG啊，其实挺简单的。传统的牛顿法是每一次迭代都要求Hessian矩阵的逆，这个复杂度就很高，为了避免求矩阵的逆，Newton-CG就用CG共轭梯度法来求解线性方程组，从而避免了求矩阵逆。

Truncated Newton's method (Hessian-Free Optimization)修正牛顿法

ParseError: KaTeX parse error: Unexpected end of input in a macro argument, expected '}' at end of input: … Projection ontoS_+^nParseError: KaTeX parse error: Expected 'EOF', got '}' at position 1: }}} Let $A \in S^n, A = UDU^T$ be its eigenvalue decomposition.

def $A_+ := UD_+U^T$,
其中：D_+ is the diagonal matrix with $(d_+)_{ii} = \max\{d_{ii}, 0\}, \forall i.$
Then A_+ is the unique solution of

$$\arg\min \|Y - A\|_F \text{ s.t.} Y \succeq 0$$

定义

1. Pick $\sigma \in (0,1), \beta \in (0,1), \overline{\alpha_k} \equiv 1$, a small $\eta > 0$ and a huge $M > 0$. Initialize at $x^0 \in \mathbb{R}^n$
2. For $k = 0, 1, 2, \ldots,$
    i. let UDU^T be an eigenvalue decomposition of $\nabla^2 f(x^k)$.
    ii. Let Λ be diagonal with $\lambda_{ii} = \max\{\min\{M, d_{ii}\}, \eta\} \qquad (\text{Project } d_{ii} \text{ on } [\eta, M])$
    iii. Set $D^k := U\Lambda U^T$ and $d^k := -D^k \nabla f(x^k)$..
    iv. Update $x^{k+1} = x^k + \alpha^k d^k$
        α^k is obtained via the Armijo line search by backtracking

Let $f \in C^2(\mathbb{R}^n)$ with $\inf f > 1$ and let $\{x^k\}$ be generated by **the truncated Newton's method.** Then any accumulation point of $\{x^k\}$ is a stationary point of f.

Computational concerns

Untitled

拟牛顿类算法

对于大规模问题，函数的海瑟矩阵计算代价特别大或者难以得到，即便得到海瑟矩阵我们还需要求解一个大规模线性方程组.
它能够在每一步以较小的计算代价生成**近似矩阵**，并且使用**近似矩阵代替海瑟矩阵而产生的迭代序列**仍具有超线性收敛的性质.
不计算海瑟矩阵 $\nabla^2 f(x)$，而是构造其**近似矩阵** $*B^k*$ **或其逆的近似矩阵** $*H^k*$

Basic idea: Secant equations

1. **思路**
    目的：$g(x) = 0, \;\; g \in C^1(\mathbb{R})$
    (Taylor Formula)

    $$g(x^{k+1}) = g(x^k) + \nabla g(x^k)(x^{k+1} - x^k) = 0 \implies x^{k+1} = x^k - \frac{g(x^k)}{\nabla g(x^k)}$$

    但当一阶导$\nabla g(x)$太难求，我们就想到了割线方程 Secant equation。Use finite difference to approximate $\nabla g(x)$
    Secant equations

    $$\nabla g(x^k) \approx \frac{g(x^k) - g(x^{k-1})}{x^k - x^{k-1}} \implies x^{k+1} = x^k - g(x^k)\frac{x^k - x^{k-1}}{g(x^k) - g(x^{k-1})}$$

    - Notes:

a. 这里同时有k+1，k，k-1. initialized at $x^0, x^{-1}, g(x^0) \neq g(x^{-1})$

b. The local convergence rate of the secant method is **typically slower** than Newton's method. However, **the computational cost** per iteration can be smaller when $*g'*$ is hard to compute compared with *g*

Untitled

2. **Example**
   - *Find the square root of 2 using the secant method, starting at* $x^{-1} = 1.4,\ x^0 = 1.5$*, up to 4 decimal places.*

     Untitled

在 descent direction 上的应用

目的：$\nabla f(x) = 0$

same ideas:

$$\nabla g(x^k)(x^k - x^{k-1}) \approx g(x^k) - g(x^{k-1})$$
$$\iff$$
$$\nabla^2 f(x^{k+1})(x^{k+1} - x^k) \approx \nabla f(x^{k+1}) - \nabla f(x^k)$$

Notation:

$$s^k := x^{k+1} - x^k,\ y^k = \nabla f(x^{k+1}) - \nabla f(x^k) \implies \nabla^2 f(x^{k+1}) s^k = y^k$$

成功的关键：我们能够连续不断地构造矩阵$\begin{cases} \text{Method 1: } B^{k+1} \approx \nabla^2 f(x^{k+1}) \\ \text{Method 2 } H^{k+1} \approx \dfrac{1}{\nabla^2 f(x^{k+1})} \end{cases}$ 去拟合海塞矩阵，使得

$\begin{cases} B^{k+1} s^k = y^k \\ H^{k+1} y^k = s^k \end{cases}$，因为我们是要迭代的，所以就是能连续生成迭代

问题：怎么迭代，迭代有什么要求

1. Initialize B^0 (or $*H^0*$) at a **positive definite** matrix.

   $$\text{(proposition of BFGS)}$$
   $$\begin{cases} H_k \succ 0 \\ {y^k}^T s^k > 0 \\ H_{k+1} \text{ is given by BFGS update} \end{cases} \implies H_{k+1} \succ 0$$
   $$\text{Same for B}$$

   - proof:
2. Since $*B^0*$ and $*H^0*$ were **symmetric** to start with, by induction, all $*B^k$*and* $*H^k$ are **symmetric**.
3. **Popular update formula**

Untitled

- Note:

i. DFP and BFGS are rank-2 updates, while SR1 is rank-1 update.
ii. In practice, **BFGS** usually performs better.

- Verify the secant equation for BFGS.

Quasi-Newton method

Given $f \in C^1(\mathbb{R}^n)$.

Initialize at $x^0 \in \mathbb{R}^n$ and $B_0, H_0 \succ 0$, is **symmetric and positive definite**

Quasi-Newton based on B_k

For $k = 0, 1, 2, \ldots$

1. Find $d^k = -B_k^{-1}\nabla f(x^k)$.
2. Update $x^{k+1} = x^k + d^k \times 1$,
3. Set $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$ and $s^k = x^{k+1} - x^k$.
4. Compute B_{k+1} by **Popular update formula**

BFGS

Quasi-Newton based on H_k

For $k = 0, 1, 2, \ldots$

1. Find $d^k = -H_k\nabla f(x^k)$.
2. Update $x^{k+1} = x^k + d^k \times 1$,
3. Set $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$ and $s^k = x^{k+1} - x^k$.
4. Compute H_{k+1} by **Popular update formula**

StepSize α_k

💡 $\text{ given } x^{k} \text{ and }d^k$ 变成了单变量优化：$\alpha^k =\argmin\limits_{\alpha>0}\varphi(\alpha) =f(x^k+\alpha d^k)$ 它是目标函数 *f* (*x*) 在射线 $\{x^k + αd^k:α > 0\}$ 上的限制

分类

exact line search strategy

对于一个一元二次问题，最优解形式为：(求极小值点问题)

$\nabla\varphi(\alpha) = [\nabla f(x^k + \alpha d^k)]^T d^k = 0$

$[\nabla f_{k+1}]^T d^k = 0$

通常需要很大计算量，在实际应用中较少使用

inexact line search strategy

寻找**步长α的一个区间**，通过逐步迭代的方法去寻找**仅仅是满足条件的点**。当搜索结束时，需要满足该步长能够对目标函数带来**充分的下降**。

More practical strategies perform an inexact
line search to identify a step length that achieves adequate reductions in f **at a minimal cost**.

Termination conditions 线搜索准则

为提高非精确算法的搜索效率，需要确定一些termination conditions 去判断是否迭代到 α^*，确保**迭代的收敛性。**

Minimization Rule

$f(x^k + \alpha^k d^k) = \min_{\alpha \geq 0} f(x^k + \alpha d^k)$

Sufficient Decrease condition ****(Armijo condition) 充分下降条件

alone is not sufficient to ensure that the algorithm makes reasonable progress along the given search direction: *α* = 0 显然满足条件，而这意味着迭代序列中的点固定不变，研究这样的步长是没有意义的
是 the Wolfe conditions 1^{st} condition
是 the Goldstein conditions 2^{nd} inequality
是 Backtracking line search 的停止标准stopping criterion， alone is ok

1. (def)
   Let $c_1 \in (0,1), x \in \mathbb{R}^n, d \in \mathbb{R}^n$. Find $\alpha > 0$ so that
   $$f(x^k + \alpha d^k) \leq f(x^k) + c_1 \alpha [\nabla f(x^k)]^T d^k$$
   $\implies \alpha$ satisifies Armijo rule
   其中： d^k is descent direction; $c_1 = 10^{-4}$ is chosen to be quite small;
   Untitled
2. **存在性证明**
   α存在 $\iff$ Armijo rule is not valid，选取符合Armijo rule 确实会使得函数值下降

Let $f \in C^1(\mathbb{R}^n), x \in \mathbb{R}^n, d \in \mathbb{R}^n$ be a descent direction at x. Let $\sigma \in (0,1)$. Then there $\exists \alpha_1 > 0$ so that $\forall \alpha \in [0, \alpha_1], f(x+\alpha d) \le f(x) + \alpha\sigma[\nabla f(x)]^T d$.

- proof：

3. **How to execute Armijo rule in practice**

   Fix $\sigma \in (0,1)$ and $\beta \in (0,1)$. Given $x \in \mathbb{R}^n, d \in \mathbb{R}^n, \overline{\alpha} > 0$. Find the smallest nonnegative integer $j = j_0$ so that

   $f(x + \overline{\alpha}\beta^j d) \le f(x) + \overline{\alpha}\beta^j \sigma[\nabla f(x)]^T d$

   $\text{normally: } \sigma = 10^{-4}, \beta = \dfrac{1}{2}, \overline{\alpha}\beta^{j_0} \text{ is the step size}$

   Note:

   i. d is a descent direction + j is sufficiently large $\rightarrow$ β^j is sufficiently small → Armijo rule satisfied。

   ii. 可证 $\overline{\alpha}\beta^j \text{ is decreasing } \therefore \text{ it is called backtracking}$

   iii. $\overline{\alpha}$ 选择对收敛效率来说很关键

4. **Convergence under Armijo rule**

   Let $f \in C^1(\mathbb{R}^n), \inf f > -\infty$.

   Let $\{\overline{\alpha}_k\} \subset \mathbb{R}$ satisfy $0 < \inf\limits_k \overline{\alpha}_k \le \sup\limits_k \overline{\alpha}_k < \infty$, and

   fix $\sigma \in (0,1), \beta \in (0,1)$.

   Suppose $\{x^k\}$ is generated as $x^{k+1} = x^k + \alpha_k d^k$

   where

   $d^k = -D_k \nabla f(x^k), \qquad \text{if } x^k \text{ is non-stationary, then } d^k \text{ is a descent direction}$

   - $\{D_k\}$ is bounded sequence of positive definite matrices with $D_k - \delta I \succeq 0$ for some independent $\delta > 0$

     $\because D_k - \delta I \succeq 0 \therefore \forall y \in \mathbb{R}^n, y^T(D_k - \delta I)y \ge 0$

     $\therefore y \in \mathbb{R}^n, y^T(D_k)y \ge \delta\|y\|_2^2$

   α_k is generated via the Armijo line search by backtracking with $x = x^k, d = d^k, \overline{\alpha} = \overline{\alpha}_k$

   normally $\sigma = 10^{-4}, \beta = \frac{1}{2}$

   Then any accumulation point of $\{x^k\}$ is a stationary point of f.

   - proof:

   for BFGS:

   $\exists M > 0, \|H_k\|_2 \|H_k^{-1}\|_2 \le M, \forall k$

   $\qquad \Longrightarrow \lim\limits_{k\to\infty} \|_2 = 0$

   $$\cos\theta_k = \frac{{d^k}^T H_k^{-1} d^k}{\|d^k\|_2 \|H_k^{-1} d^k\|_2} \ge \frac{{d^k}^T {H_k}^{-1} d^k}{\|H_k^{-1}\| \|d^k\|_2^2} \ge \frac{\lambda_{\min}(H_k^{-1})}{\|H_k^{-1}\|_2} = \frac{1}{\lambda_{\max}(H_k)\|H_k^{-1}\|_2} = \frac{1}{\|H_k^{-1}\|_2 \|H_k\|_2} \ge \frac{1}{M}$$

5. **Sufficient Decrease and Backtracking approach**

   use **just the sufficient decrease** condition to terminate the line search procedure

   Untitled

Wolfe conditions

(def)
1^{st} ： sufficient decrease condition： $f(x^k+\alpha^k d^k) \le f(x^k) + c_1\alpha^k[\nabla f(x^k)]^T d^k$
2^{nd} ： curvature condition： $\nabla f(x^k+\alpha^k d^k)^T d^k \ge c_2 \nabla f_k^T d^k$
with $0 < c_1 < c_2 < 1$， c_1 usually 10^{-3}, c_2 usually 0.9
φ(α) 在点 α 处切线的斜率不能小于 φ′(0) 的 *c_2* 倍

sufficient decrease condition

curvature condition

1. (def)
$$\begin{array}{ccc} \nabla f(x^k+\alpha^k d^k)^T d^k & \ge & c_2 \nabla f_k^T d^k \\ \| & & \| \\ \nabla\varphi(\alpha^k) & & c_2\nabla\varphi(0) \end{array}$$
其中: $c_2 = 0.9$ in Newton or quasi-Newton method, $c_2 = 0.1$ in a nonlinear conjugate gradient method

Untitled

1. **Wolfe conditions 存在性证明：是有区间能满足 Wolfe conditions**
    - proof：

The strong Wolfe conditions

modify the curvature condition to force α^k to lie in at least a broad neighborhood of a local minimizer or stationary point of ϕ. The only difference with the Wolfe conditions is that we no longer allow the derivative $\varphi\prime(\alpha^k)$ to be too positive.

(def)
1^{st} ： sufficient decrease condition： $f(x^k+\alpha^k d^k) \le f(x^k) + c_1\alpha^k[\nabla f(x^k)]^T d^k$
2^{nd} ： **modified** curvature condition： $|\nabla f(x^k+\alpha^k d^k)^T d^k| \le c_2|\nabla f_k^T d^k|$
with $0 < c_1 < c_2 < 1$

Untitled

Convergence under Wolfe conditions

$$\text{(Zoutendijk's theorem)}$$
$$f \in C^1(\mathbb{R}^n), \inf f > -\infty, x^0 \in \mathbb{R}^n,$$
$$\{x^k\} \text{is a sequence of non-stationary points generated as } x^{k+1} + \alpha_k d^k,$$

$$
\begin{cases}
f \in C^1(\mathbb{R}^n), \inf f > -\infty \text{ (下有界，连续可微)} \\
\exists \ell > 0, \|\nabla f(x) - \nabla f(y)\|_2 \le \ell \|x-y\|_2,\ \forall x, y \in \mathbb{R}^n \text{ (梯度满足L-利普希茨连续)} \\
d^k \text{ is a descent direction} \\
\alpha_k \text{ satisfies the Wolfe conditions (Wolfe)}
\end{cases}
$$

$$
\implies \sum_{k=0}^{\infty} \cos^2\theta_k \|\nabla f(x^k)\|_2^2 < \infty,
$$

$$
\implies \exists \delta, \text{so that } \cos\theta_k = \frac{-[\nabla f(x^k)]^T d^k}{\|\nabla f(x^k)\|_2 \|d^k\|_2} \ge \delta, \forall k (\text{independent of k})
$$

1. $\|\nabla f(x^*)\| = 0 \to \|\nabla f(x^n)\| < \varepsilon$

Goldstein conditions 条件

(def)

$$
f(x^k) + (1-c)\alpha^k[\nabla f(x^k)]^T d^k \le f(x^k + \alpha^k d^k) \le f(x^k) + c\alpha^k [\nabla f(x^k)]^T d^k
$$

with $0 < c < \dfrac{1}{2}$

2^{nd} ≤： sufficient decrease condition

必须在两条直线之间

are often used in **Newton-type methods** but are not well suited for quasi-Newton methods that maintain a positive definite Hessian approximation

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**.

Untitled