第一章

1. Introduction

optimization problem

$$\begin{aligned} \min \ & f(x) \\ \text{s.t.} \ & g(x) \leq 0 \\ & h(x) = 0 \end{aligned}$$

$x = (x_1, \cdots, x_n) \in \mathbb{R}^n$: optimization variables

$f: \mathbb{R}^n \mapsto \mathbb{R}$: objective function

$g: \mathbb{R}^n \mapsto \mathbb{R}^{m_g}$: inequality constraint functions

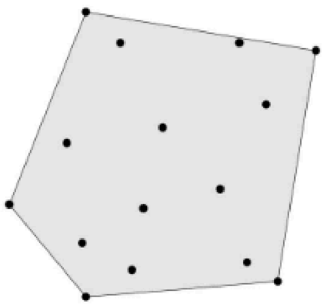
$h: \mathbb{R}^n \mapsto \mathbb{R}^{m_h}$: equality constraint functions

optimal solution x^* has smallest value f among all vectors that satisfy the constraints.

前提:

```
1. f(x) is lower bounded 有下界
1. f(s) has bounded sub-level set 有有界的水平集
```

2. Convex Sets

> ### Convex Sets
>
> Def: A set is convex if every line between its two points stays in the set?
>
> $$\theta x_1 + (1-\theta)x_2, \quad 0 \leq \theta \leq 1$$
>
> More General:
> All **convex combinations**
> lie in set.
>
> $$\sum \theta_i x_i = \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$
> $$\sum \theta_i = \theta_1 + \theta_2 + \theta_3 = 1, \quad \theta_i \geq 0 \forall i$$
>
> 

- 凸组合：表示凸包
- 重心坐标：Barycentric coordinate

  凸集中任意一点的凸表示形式：θ_i表示广义重心坐标

- convex hull：集合中所有点的凸组合，找一个最小的凸集把非凸的集合包裹起来

3. High-Order Info of Functions

- 高阶导数信息

  Gradient $\mathbb{R} \mapsto \mathbb{R}^n$

  Hessian $\mathbb{R} \mapsto \mathbb{R}^{n\times n}$

  Hessian是Gradient的Jacobian

  光滑函数的Hessian是对称的

光滑函数（英语：Smooth function）在数学中特指无穷可导的函数，不存在尖点，也就是说所有的有限阶导数都存在。例如，指数函数就是光滑的，因为指数函数的导数是指数函数本身。

若一函数是连续的，则称其为C^0函数；若函数存在导函数，且其导函数连续，则称为**连续可导**，记为C^1函数；若一函数n阶可导，并且其n阶导函数连续，则为C^n函数（$n \geq 1$）。而光滑函数是对所有n都属于C^n函数，特称其为C^∞ **函数**。

- Function $f(x) = f(x_1, x_2, x_3)$

  Gradient $\nabla f(x) = \begin{pmatrix} \partial_1 f(x) \\ \partial_2 f(x) \\ \partial_3 f(x) \end{pmatrix}$

  Hessian $\nabla^2 f(x) = \begin{pmatrix} \partial_1^2 f(x) & \partial_1\partial_2 f(x) & \partial_1\partial_3 f(x) \\ \partial_2\partial_1 f(x) & \partial_2^2 f(x) & \partial_2\partial_3 f(x) \\ \partial_3\partial_1 f(x) & \partial_3\partial_2 f(x) & \partial_3^2 f(x) \end{pmatrix}$ Symmetric for smooth func

  Approx at zero $f(x) = \underbrace{\overbrace{f(0) + x^T \nabla f(0)}^{\text{Linear approx}} + \frac{1}{2} x^T \nabla^2 f(0) x}_{\text{Quadratic approx}} + O\left(\|x - x_0\|^3\right)$

Jacobian

The extension of the gradient of multidimensional $f(x) : \mathbb{R}^n \to \mathbb{R}^m$:

$$f'(x) = \frac{df}{dx^T} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}$$

$$f(x): X \to Y; \qquad \frac{\partial f(x)}{\partial x} \in G$$

X	Y	G	Name
$\mathbb{R}$	$\mathbb{R}$	$\mathbb{R}$	$f'(x)$ (derivative)
$\mathbb{R}^n$	$\mathbb{R}$	$\mathbb{R}^{\mathrm{n}}$	$\frac{\partial f}{\partial x_i}$ (gradient)
$\mathbb{R}^n$	$\mathbb{R}^m$	$\mathbb{R}^{n\times m}$	$\frac{\partial f_i}{\partial x_j}$ (jacobian)
$\mathbb{R}^{m\times n}$	$\mathbb{R}$	$\mathbb{R}^{m\times n}$	$\frac{\partial f}{\partial x_{ij}}$

- 分子分母布局可以确定结果的行列

The notation of differentials could be useful here

$$\begin{aligned}
&dA = 0 \\
&d(\alpha X) = \alpha(dX) \\
&d(AXB) = A(dX)B \\
&d(X+Y) = dX + dY \\
&d\left(X^\top\right) = (dX)^\top \\
&d(XY) = (dX)Y + X(dY) \\
&d\langle X, Y\rangle = \langle dX, Y\rangle + \langle X, dY\rangle \\
&d\left(\frac{X}{\phi}\right) = \frac{\phi dX - (d\phi)X}{\phi^2} \\
&d\,\mathrm{tr}\,X = I \\
&df(g(x)) = \frac{df}{dg}\cdot dg(x)
\end{aligned}$$

4. Convex Functions

f(x)上任意两点的线段一定在函数上方

local minimum 一定是 global minimum，并且解集一定是凸的

凸函数有凸的次水平集

仿射变换$Ax+b$保留凸性

max保留凸性（f(x)定义为几个凸函数中的最大函数）

Other allowed operations

Set sum $$A + B = \{x + y \mid x \in A, y \in B\}$$

Set product $$A \times B = \{(x, y) \mid x \in A, y \in B\}$$

Point-wise max preserves convexity

$$g(x) = \max_i f_i(x)$$

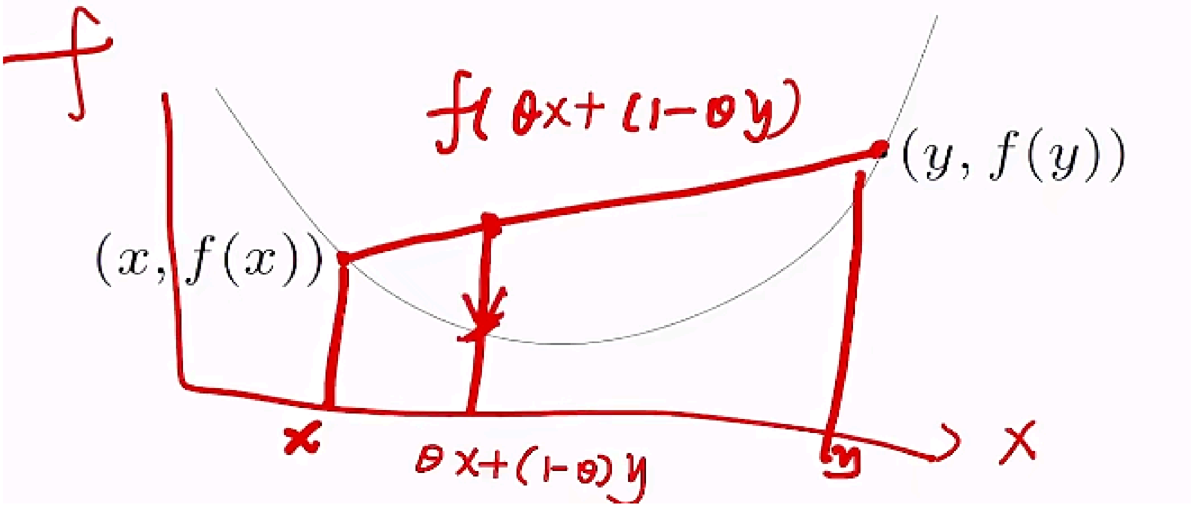
Absolute value $$|x| = \max\{x, -x\}$$

Infinity norm $$\|x\|_\infty = \max_i |x_i|$$

Max eigenvalue $$\|A\|_2 = \max_v v^T A v$$

Jensen's inequality

$$f(\theta x + (1-\theta)y) \le \theta f(x) + (1-\theta) f(y)$$



- 凸函数:

Convex Functions

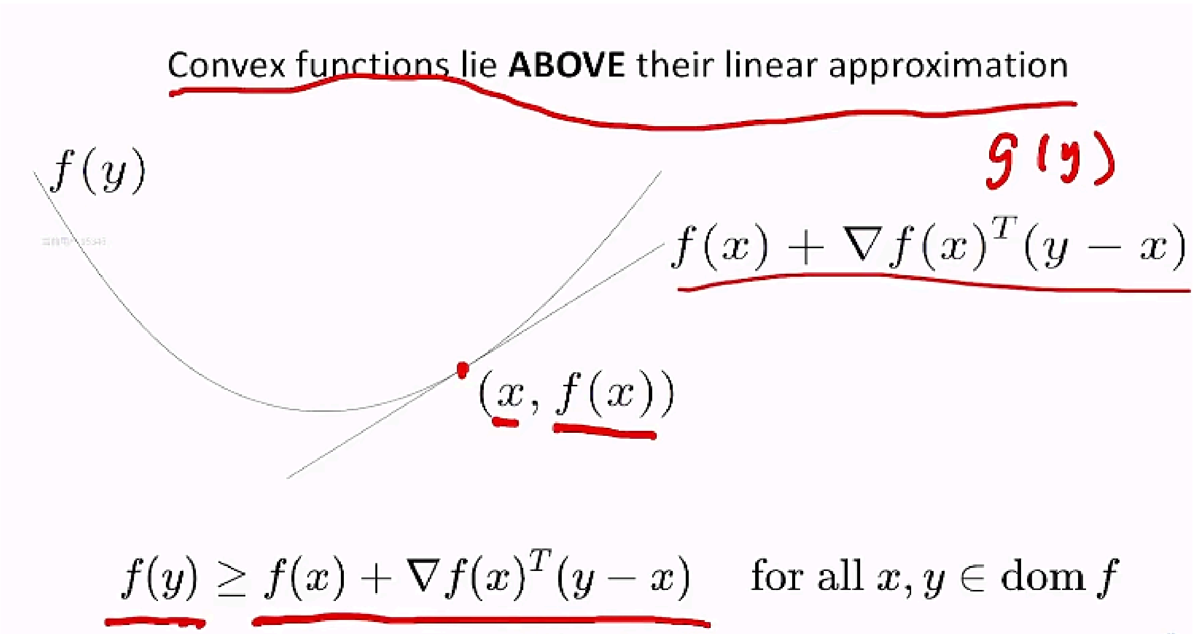
Why are these convex?

Trace	$f(X) = \mathrm{trace}(A^T X)$	Linear operator
Distance over set	$f(x) = \max\limits_{y \in C} \|x - y\|$	Max over convex
Distance to convex set	$f(x) = \min\limits_{y \in C} \|x - y\|$	Special case
Affine Norm	$f(x) = \|b + \sum_i A_i x_i\|_2$	Affine comp

If g(x,y) is convex, then minimizing for y preserves convexity

- 凸函数性质

  整个函数在某点处切线的上方



  梯度为0，全局最优，若函数非凸，则为局部最优

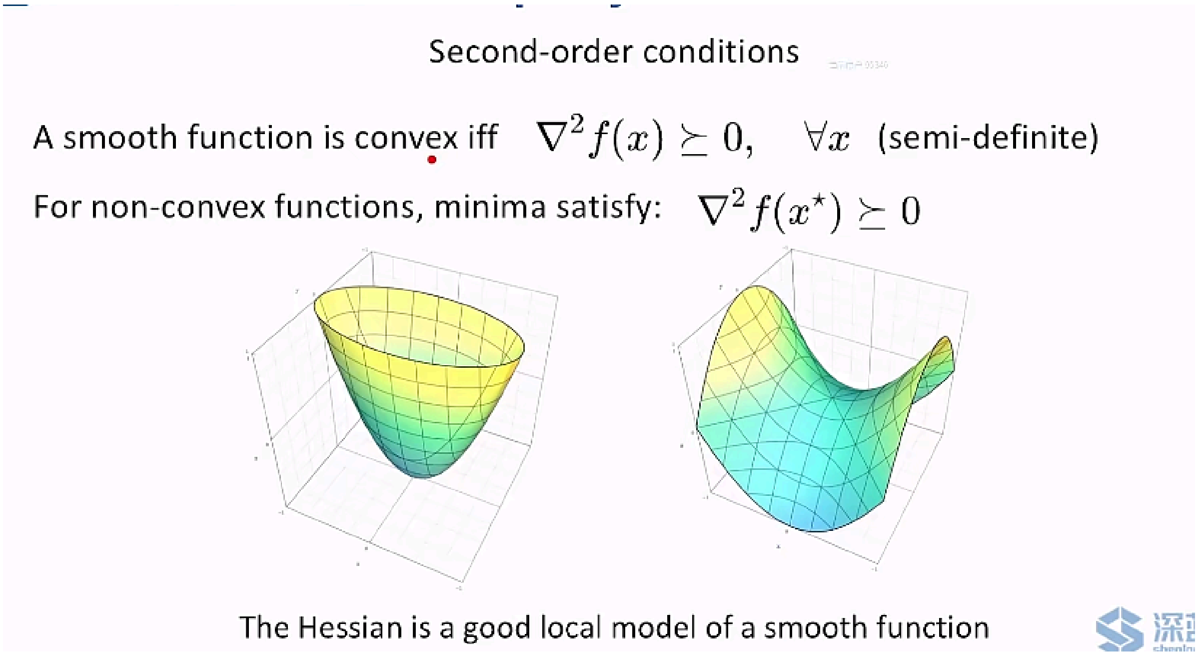
- 若函数光滑

光滑性类别	梯度 $\nabla f(x)$	Hessian $H_f(x)$	混合偏导对称性
C^0（连续）	不一定存在	不一定存在	-
C^1（一阶可微）	存在且连续	可能不存在	可能不对称
C^2（二阶可微）	存在且连续	存在且连续	确保对称
C^k（k 阶可微）	存在且连续	存在且可微	对称（只要 C^2）
C^∞（光滑）	无限阶可微	无限阶可微	对称

1. **Hessian 矩阵的存在性要求函数至少是 C^2**，即所有二阶偏导数都存在且连续。
2. **如果函数仅是二阶可微（但二阶导数不连续），Hessian 仍然存在，但混合偏导不一定对称。**
3. **Hessian 对称性需要 C^2 条件（Schwarz 定理）。**
4. **光滑函数（C^∞）不仅保证 Hessian 矩阵存在，还能保证所有高阶导数存在。**
5. **非光滑函数（如 ReLU）可能没有 Hessian 矩阵**，例如：

$$f(x) = \max(0, x)$$

在 $x = 0$ 处不可导，所以 Hessian 矩阵不存在。



光滑函数或Hessian存在（至少C^2）的函数为凸当且仅当**任意点处**Hessian是半正定

非凸光滑函数**local minima处**的Hessian半正定

- 强凸性质

一个凸函数，Hessian严格正定，则函数为强凸的

1. 得到f(x)下界

Strong convexity

$$f(y) \geq \underbrace{f(x) + (y-x)^T \nabla f(x)}_{\text{holds for any convex func}} + \underbrace{\frac{m}{2}\|y-x\|^2}_{\text{min curvature}}$$

When Hessian exists

$$\begin{aligned} f(y) &\approx f(x) + (y-x)^T \nabla f(x) + \frac{1}{2}(y-x)^T \nabla^2 f(x)(y-x) \\ &\geq f(x) + (y-x)^T \nabla f(x) + \frac{\lambda_{\min}}{2}\|y-x\|^2 \end{aligned}$$

and so

$$\nabla^2 f(x) \succeq mI$$

2. 得到f(x)上界

The Lipschitz constant M of $\nabla f(x)$ satisfies

$$\|\nabla f(x) - \nabla f(y)\| \leq M\|y-x\|$$

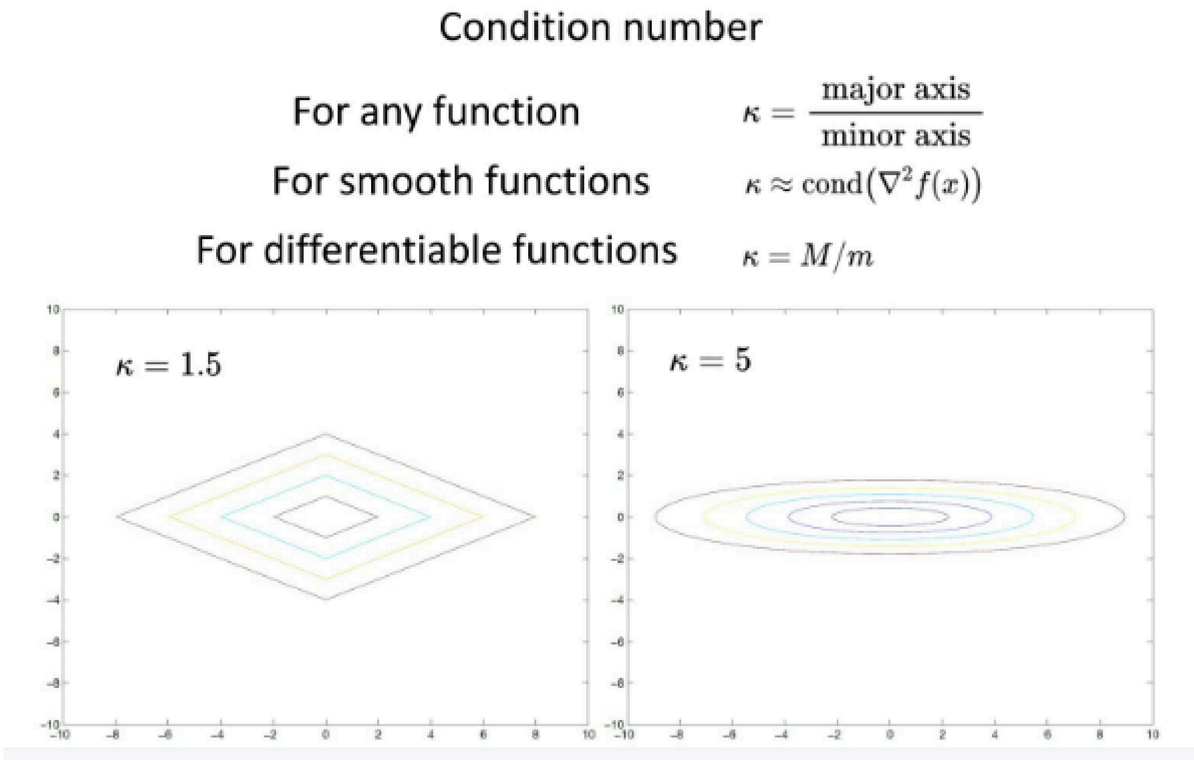
$$f(y) \leq f(x) + (y-x)^T \nabla f(x) + \frac{M}{2}\|y-x\|^2$$

3. 约束函数值的界，m、M可以用来判别函数的条件数

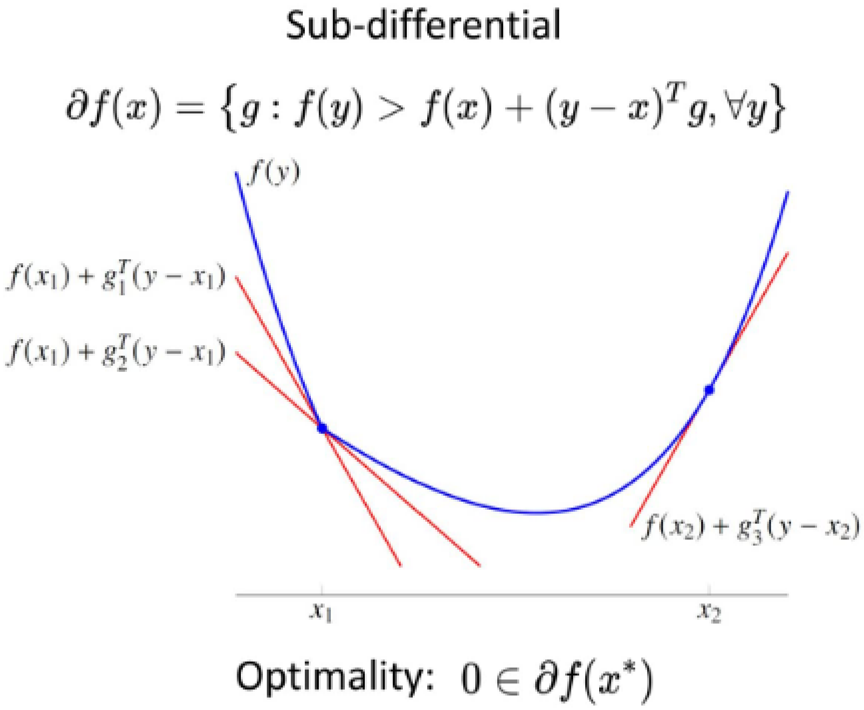
We can bound the objective error in terms of distance from minimizer

$$f(y) - f(x^\star) \geq \frac{m}{2}\|y-x^\star\|^2 \qquad f(y) - f(x^\star) \leq \frac{M}{2}\|y-x^\star\|^2$$

（暂时没懂）条件数



- 凸但是不可微的函数（不可微的凸函数的全局极小值）



1. 可微的点处为切线，连续但不可微的地方为梯度的凸组合（次梯度）
2. 最优性：0属于次梯度（比如f(x)=|x|，0属于{-1,1}的凸组合）
3. **不光滑**的函数沿着次梯度反方向走函数值可能不下降反而上升（对收敛速度造成挑战）需要注意的是**负的次梯度方向**不一定是函数值下降方向，而只有**方向导数 <0 的方向**才是函数值下降方向
4. 如果**知道non-smooth的分界点**时，f(x)下降最快的方向为**non-smooth点处次梯度模长最小向量的反方向（如何证明？）**，可以加快收敛速，防止振荡

- 凸函数的梯度/次梯度单调性

Monotonicity: The (sub) gradient of any convex func is monotone

$$\langle y-x, \nabla f(y)-\nabla f(x)\rangle \geq 0$$

or

$$\langle y-x, g_y-g_x\rangle \geq 0,\ g_x \in \partial f(x),\ g_y \in \partial f(y)$$

This can be obtained by adding two equations below.

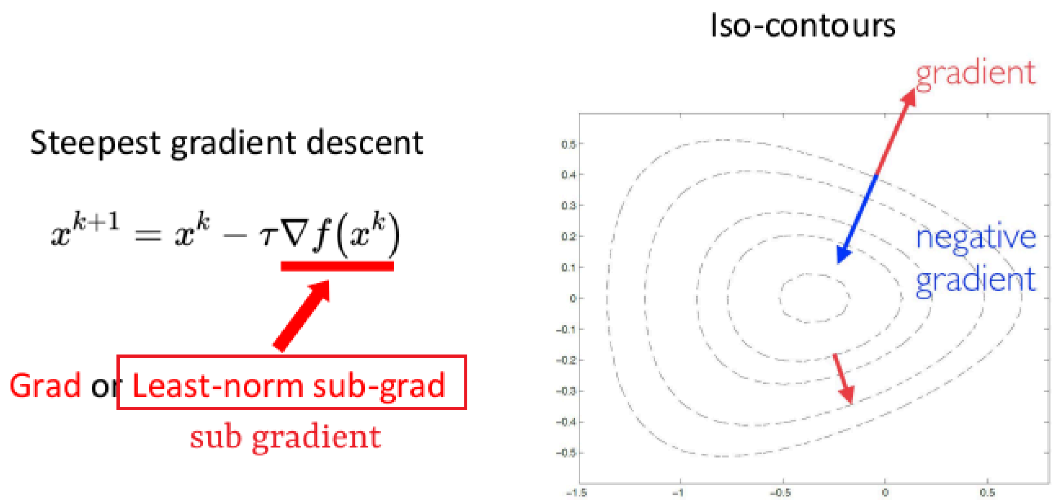
$$f(y) \geq f(x)+\nabla f(x)^T(y-x)$$

$$f(x) \geq f(y)+\nabla f(y)^T(x-y)$$

如果函数严格凸，取严格大于号

5. Unconstrained Optimization for Nonconvex Functions

- **Steepest Gradient Desent 最速梯度下降**



exact line search本质上需要求$minf(x_{k+1})$，采用了当前“最好”的方向，和“最好”的步长因子。因为最好的方向就是梯度的反向，最好的步长因子满足

$$\alpha_k = \mathrm{argmin}_\alpha f(x_k+\alpha d_k)$$

$$\begin{aligned}
&\mathrm{while}\ ||\bigtriangledown f(x_k)|| \geq \epsilon \\
&\qquad d_k = -\bigtriangledown f(x_k) \\
&\qquad \alpha_k = \mathrm{argmin}_\alpha f(x_k+\alpha d_k) \\
&\qquad x_{k+1} = x_k+\alpha_k d_k \\
&\qquad k = k+1 \\
&\mathrm{end}
\end{aligned}$$

每一步仍需要求一个子的优化问题，这个代价一般上比较大，但是对于二次函数可以方便求极值，**二次函数，所以才可能直接求出最优的步长，对于任意非线性函数，最优步长一般是无法求得解析解的，所以才会有Backtracking Line Search方法**。

最速下降法使用的**相邻下降方向是正交**的。在最速下降法中，当次迭代的梯度方向也是和上次迭代梯度方向垂直，但和再之前的梯度方向就不垂直了，所以会有“之”形路线。然后，共轭梯度法要求的是关于矩阵正交，并非直接正交。这是因为每一步都将一个方向走到最优，对于n维空间，那么只需走n步，每一步走的方向都是一个维度。（注意每一步走的方向所代表的维度不一定与坐标轴平行（我们这里说的维度不是坐标轴，是该空间的任何基中的一个方向空间的维数是什么）。但是每个方向之间一定要正交，所以表现出来共轭梯度只需要n步）

1. 如何确定搜索的步长

- Constant step size $\tau = c$ Not intelligent
- Diminishing step size $\tau = c/k$ Robbins-Monro rule for expensive func
- Exact line search $\tau = \arg\min_{\alpha} f\left(x^k + \alpha d\right)$ Generally nontrivial
- Inexact line search $\boxed{\tau \in \left\{\alpha \mid f(x^k) - f(x^k + \alpha d) \geq -c \cdot \alpha d^{\mathrm{T}} \nabla f(x^k)\right\}}$

Armijo condition, easy to satisfy

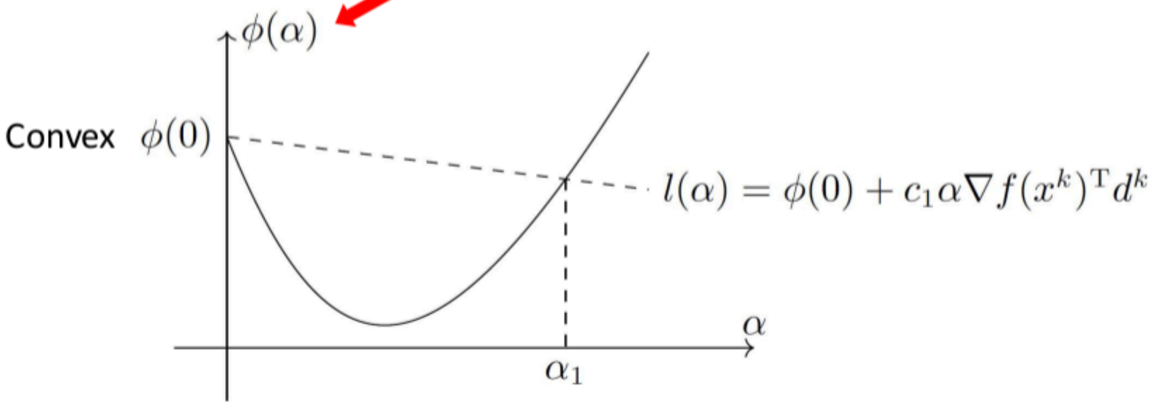
好的步长：使函数充分下降，不能太松弛（收敛很慢）也不能太激进（发散震荡）

2. Backtracking/Armijo Line search(本质上是让$f(x_k + \alpha d)$充分下降)

Armijo condition (sufficient decrease condition)

$$\tau \in \left\{\alpha \mid f(x^k) - \underline{f(x^k + \alpha d)} \geq -c \cdot \alpha d^{\mathrm{T}} \nabla f(x^k)\right\}$$

$$c \in (0, 1)$$



实际使用中可以用Armijo condition（充分下降条件）作为选择步长是否合适的判断准则，用二分法找合适的步长（适用于光滑或者分片光滑，凸或者非凸都可以收敛到local minima）

Choose search direction: $d = -\nabla f(x^k)$

While $f(x^k + \tau d) > f(x^k) + c \cdot \tau d^T \nabla f(x^k)$

$\tau \leftarrow \tau / 2$

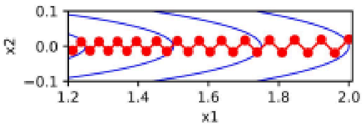
Update iterate $x^{k+1} = x^k + \tau d$

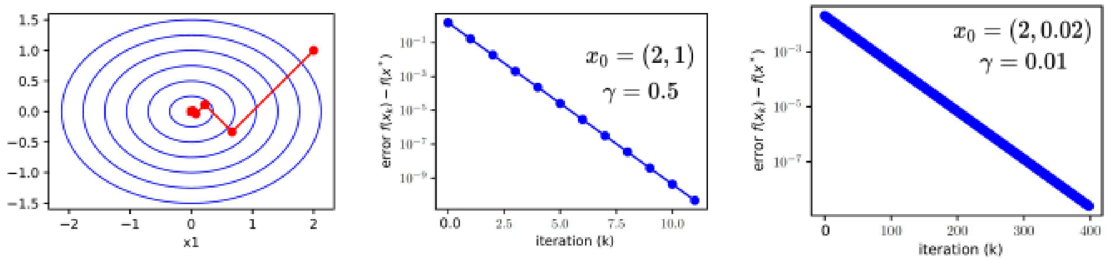
Repeat this until gradient is small
or subdifferential contains zero.

3. 最速下降法的缺点

Drawbacks: Poor conditioning causes performance degeneration

$$f(x_1, x_2) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} = \frac{\gamma}{2} x_1^2 + \frac{1}{2} x_2^2, \quad \boldsymbol{Q} = \mathrm{diag}\{\gamma, 1\}$$





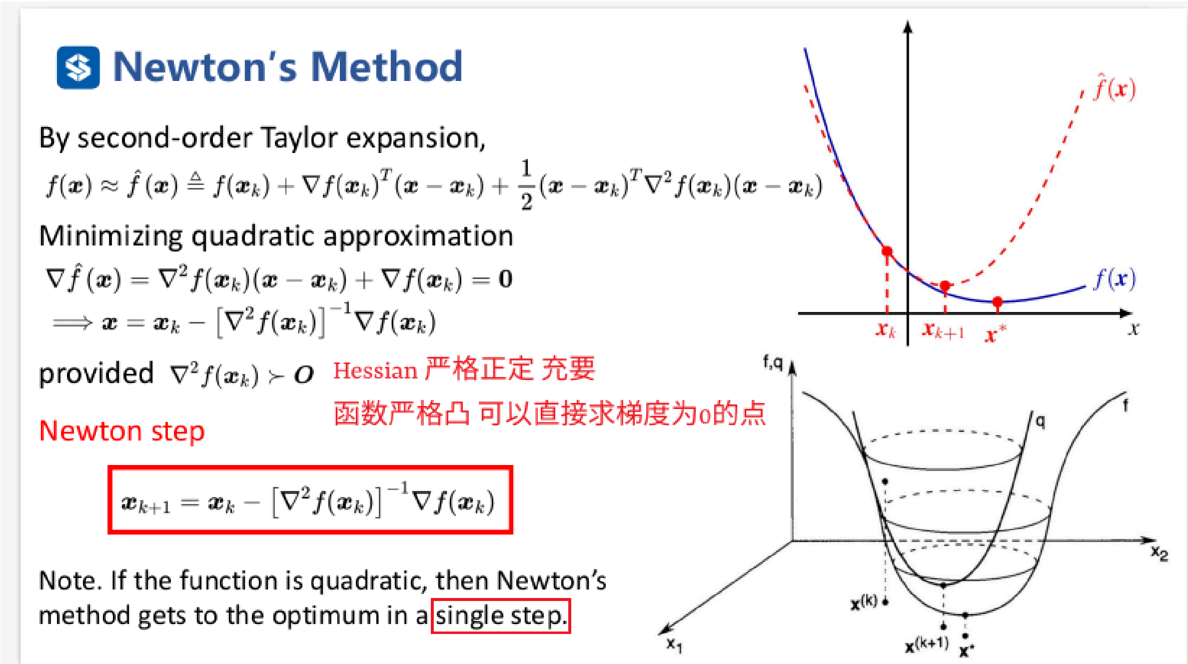
Curvature info is needed!

对于条件数比较敏感，原因是最速梯度下降只用了函数的**一阶信息**，而条件数表明了函数的**高阶信息**，包含了函数的**曲率信息**，表示函数被压的多宽多扁，不利用曲率信息就会不断震荡

6. Modified Damped Newton's Method

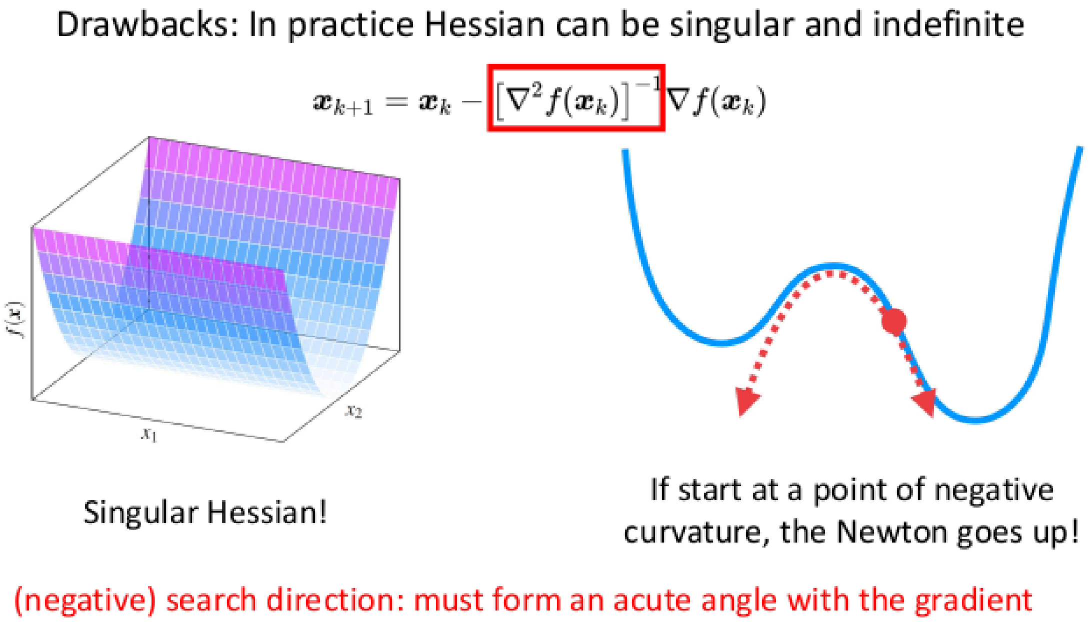
要求：f(x),f'(x),f''(x)连续，Hessian连续，函数可以是非凸

- **Newton's Method**



若f(x)本身是**严格正定**的二次函数，牛顿法可以一步收敛到精确解

缺点：每次迭代需要求一次Hessian矩阵的逆,Hessian必须严格正定，需要让牛顿步长与负梯度方向夹角小于90度



评判数值优化算法的标准：

Three aspect to evaluate a numerical optimization method:

1. Convergence speed (How to measure the rate? In Lec2.).
2. Stability when applied to different functions.
3. Computation work per iteration.

- **Practical Newton's Method**

initialization $\boldsymbol{x} \leftarrow \boldsymbol{x}_0 \in \mathbb{R}^n$
while $\|\nabla f(\boldsymbol{x})\| > \delta$ do
    $\boldsymbol{d} \leftarrow -\boldsymbol{M}^{-1}\nabla f(\boldsymbol{x})$
    $t \leftarrow$ backtracking line search
    $x \leftarrow x + td$
end while
return

- Choose a positive-definite ***M*** that is close to the Hessian 用严格正定M代替Hessian
- Solve the linear system via factorization instead of inversion 用linear solver代替矩阵求逆
- line search does not need grad and Hessian line search 只需要函数值的信息 不需要gradient和hessian

1. Hessian半正定
   - 函数是凸的
     - If function is convex, its Hessian must be PSD. We choose a ***M*** as

$$\boldsymbol{M} = \nabla^2 f(\boldsymbol{x}) + \epsilon \boldsymbol{I}, \boxed{\epsilon = \min(1, \|\nabla f(\boldsymbol{x})\|_\infty)/10}$$

Since ***M*** is PD, the search direction is solved by Cholesky factorization

$$\boldsymbol{M}\boldsymbol{d} = -\nabla f(\boldsymbol{x}), \boxed{\boldsymbol{M} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{T}}}$$

where ***L*** is a lower triangular matrix.

   - 函数是非凸的

- If function is nonconvex, its Hessian can be indefinite. We compute ***M*** on the fly through

Bunch-Kaufman Factorization

$$\boldsymbol{M}\boldsymbol{d} = -\nabla f(\boldsymbol{x}), \ \boldsymbol{M} = \boldsymbol{L}\boldsymbol{B}\boldsymbol{L}^{\mathrm{T}}$$

where ***B*** is block diagonal matrix with block size 1x1 or 2x2. All 2x2 blocks contain nonpositive eigenvalues, which are easy to modify.

把所有2×2矩阵负的特征值都变为ε的极小量

稀疏矩阵Hessian分解可以进一步加快