交通大数据理论与方法

凸规划与梯度下降算法

- 刘志远教授
- Email: zhiyuanl@seu.edu.cn

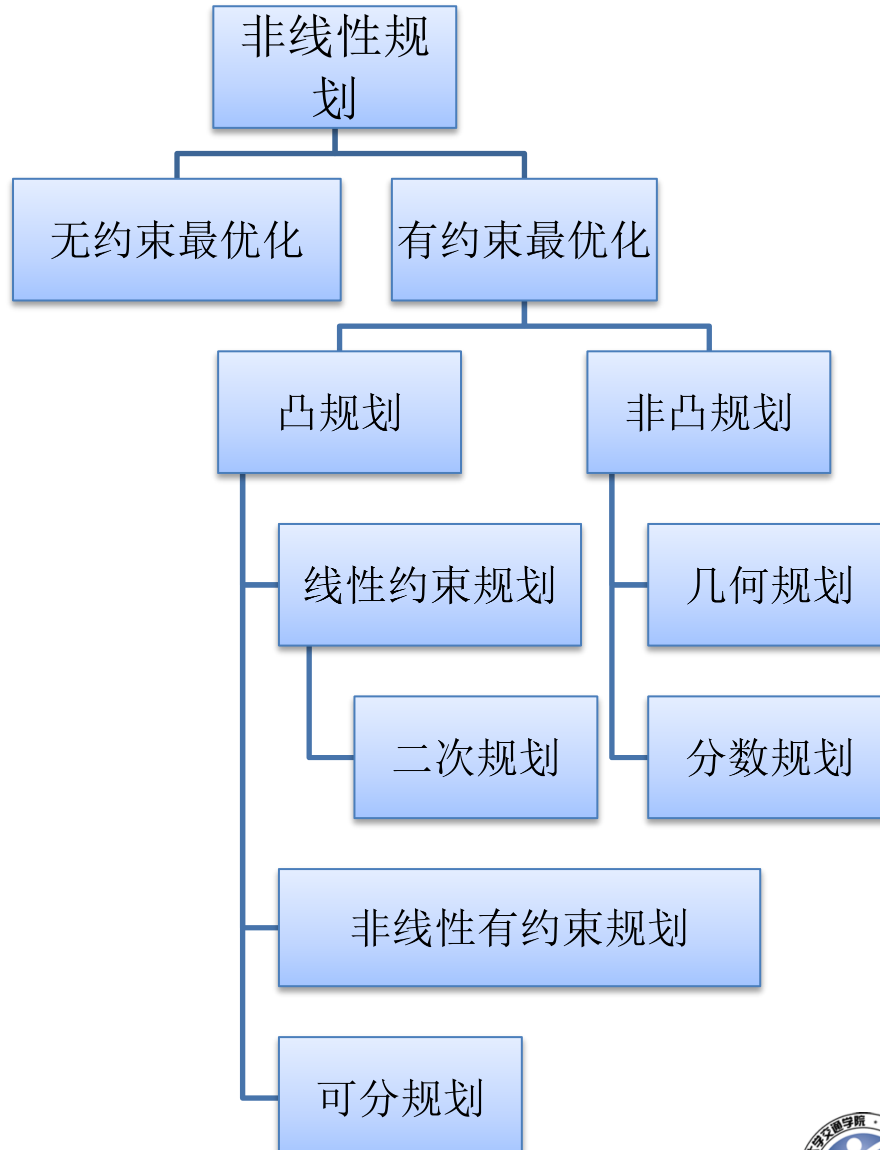
非线性规划

学习目标

- 了解非线性规划（Nonlinear Programming, NLP ）与凸规划（ Convex Programming ）的概念和分类；
- 学习KKT（Karush-Kuhn-Tucker）条件；
- 知道如何用KKT条件找到局部最优解；
- 了解求解凸规划问题的最速下降方向法；
- 掌握求解凸规划问题的Frank-Wolfe算法.

数学规划

- 线性规划
- 非线性规划
- 动态规划
- 图论
- 随机规划
- ……



NLP的定义

问题描述

确定一组决策变量的值, 使目标函数达到最优， 同时满足约束条件。

$$\min\ (\text{or max})\, f(x_1,\ldots,x_n) \longleftarrow \text{目标函数}$$

subject to

$$g_1(x_1,\cdots,x_n)\le 0$$
$$g_2(x_1,\cdots,x_n)\le 0 \longleftarrow \text{不等式约束}$$
$$\vdots$$
$$g_{m_1}(x_1,\cdots,x_n)\le 0$$
$$h_1(x_1,\cdots,x_n)=0$$
$$h_2(x_1,\cdots,x_n)=0 \longleftarrow \text{等式约束}$$
$$\vdots$$
$$h_{m_2}(x_1,\cdots,x_n)=0$$

NLP的定义

- **向量表示:**
  - 决策变量: $x=\left(x_1,\cdots,x_n\right)^T$
  - 不等式约束: $g=\left(g_1,\cdots,g_{m_1}\right)^T$
  - 等式约束: $h=\left(h_1,\cdots,h_{m_2}\right)^T$
- 假设

  $f(x)$, $g(x)$ 和 $h(x)$ 都是连续可微函数
- 非线性规划的向量表示

$$\min_x f(x)$$

subject to

$$g(x)\le 0$$

$$h(x)=0$$

$$\min_{x\in\Omega} f(x)$$

where

$$\Omega=\left\{x \mid g(x)\le 0, h(x)=0\right\}$$

- <u>注意向量和集合的符号表达。</u>

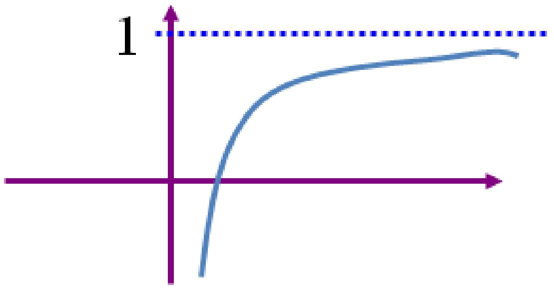
基础知识点

❑NLP的求解难点

1. 局部最优 ≠ 全局最优
2. 与 LP相比, NLP的最优值可能不是在极值点（顶点）处出现
3. 即使 $f(x)$是有界的 , 也可能不存在最优解

例如

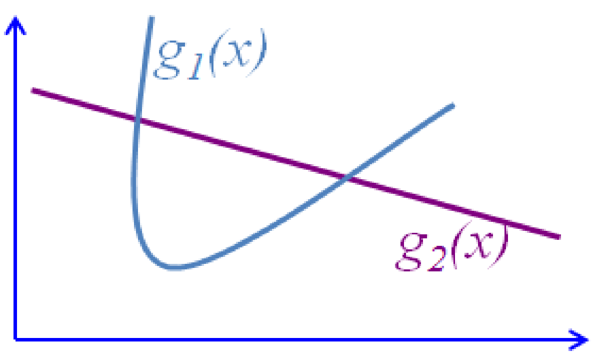
$$f(x)=1-\frac{1}{x},\Omega=\{x:x\geq 0.5\},\Rightarrow \max_{x\in\Omega} f(x)=?$$



4. 可行域可能不是相互连接的

例如

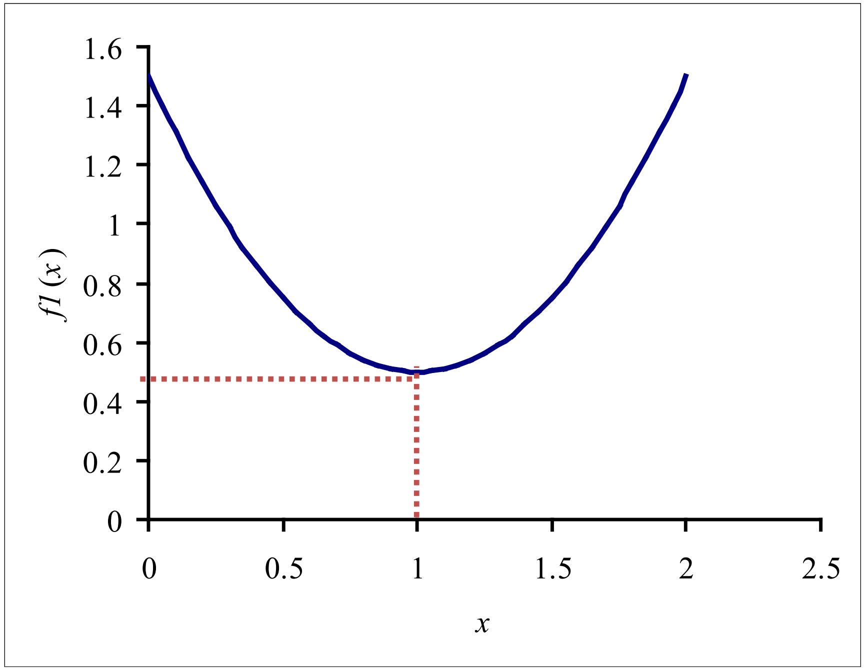
$$\Omega=\{x:g_1(x)\geq g_2(x)\}$$



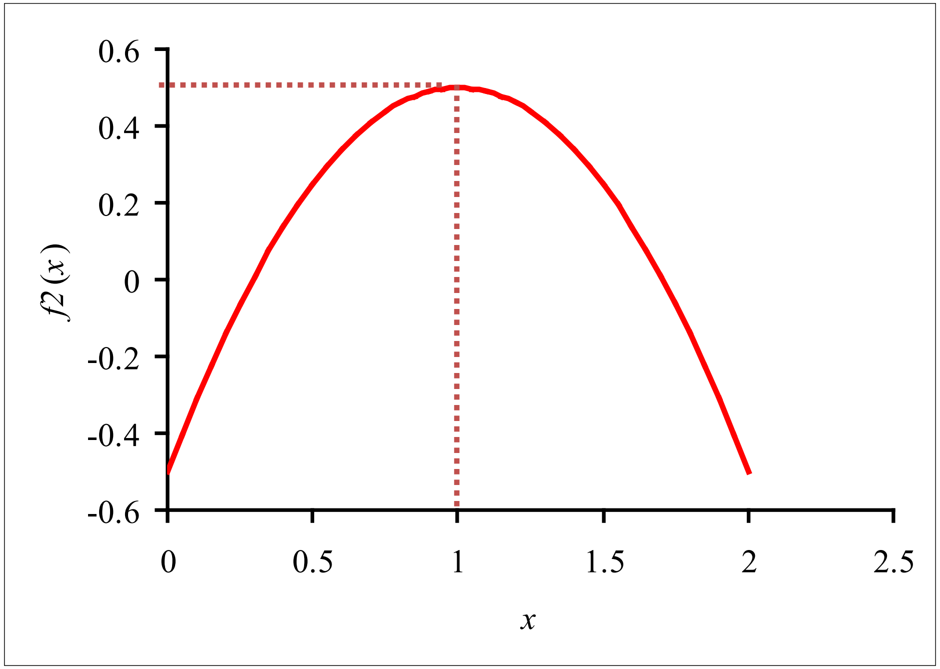
基础知识点

❑ 极大值解和极小值解

- $x^*=1$是函数 $f_1(x)$的全局极小值解
- $x^*=1$是函数 $f_2(x)$的全局极大值解



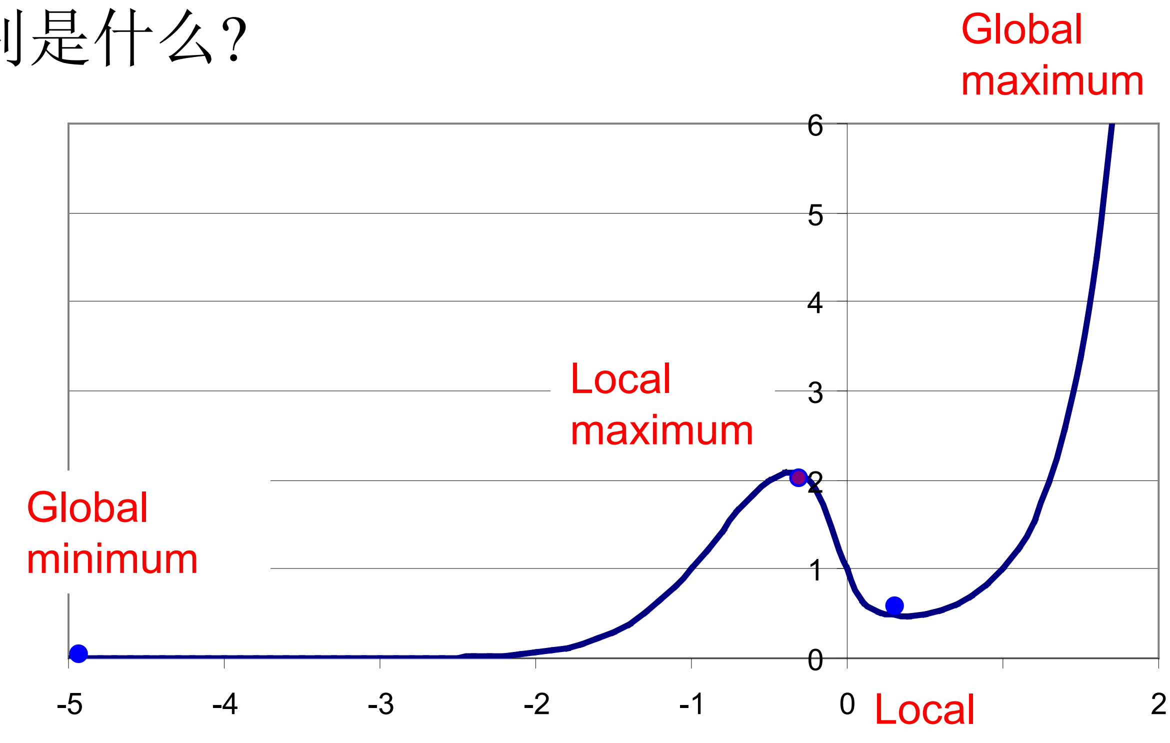
(a) 函数 $f_1(x)=(x-1)^2+0.5$



(b) 函数 $f_2(x)=0.5-(x-1)^2$

基础知识点

- 函数 $f_3(x)$的全局极小值解和极大值解分别是什么?
- 函数 $f_3(x)$在可行域 $-5 \le x \le 2$ 内的全局极小值解和极大值解分别是什么?



(c) 函数 $f_3(x)=\left(x^2\right)$

基础知识点

- **最小化问题的全局极小值解和局部极小值解**

$$\min_{x\in\Omega} f(x)$$

- 全局极小值解 x^*

(i) (可行性) x^* 是极小化问题的一个可行解, 即 $x^* \in \Omega$

(ii) (最优性) 对于任意的 $x \in \Omega$ 都满足 $f(x) \geq f(x^*)$

- 局部极小值解 $\overline{x}$

(i) (可行性) $\overline{x}$是极小化问题的一个可行解, 即 $\overline{x} \in \Omega$

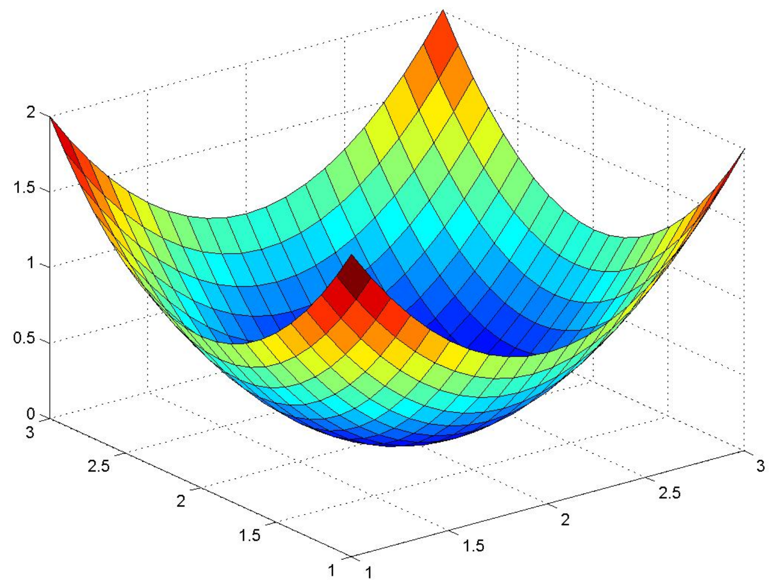
(ii) (最优性) 对于 $\overline{x}$ 邻域内的任意x都满足 $f(x) \geq f(\overline{x})$

多元函数

- 例

$$(1)\ f_1(x) = (x_1 - 2)^2 + (x_2 - 2)^2$$

$$\text{向量}: x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \Re^2$$



$$(2)\ f_2(x) = 2x_1 + 3x_2 + 0.5x_3 - x_4$$

$$\text{向量}: x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \in \Re^4$$

多元函数--梯度

- **一个*n*元函数*f(x)*在最优解*x=x**处的梯度可表示为：**

$$\nabla f\left(x^*\right)=\begin{pmatrix}\partial f\left(x^*\right)/\partial x_1\\ \partial f\left(x^*\right)/\partial x_2\\ \vdots\\ \partial f\left(x^*\right)/\partial x_n\end{pmatrix}$$

$f:R^n\rightarrow R^1$

what if $R^n\rightarrow R^n$?

$Jacobian$矩阵

- **例**

$$(1)\ f_1\left(x\right)=\left(x_1-2\right)^2+\left(x_2-2\right)^2$$

$$\nabla f_1\left(x^*\right)=\begin{pmatrix}2\left(x_1^*-2\right)\\ 2\left(x_2^*-2\right)\end{pmatrix}$$

$$(2)\ f_2\left(x\right)=2x_1+3x_2+0.5x_3-x_4$$

$$\nabla f_2\left(x^*\right)=\begin{pmatrix}2\\ 3\\ 0.5\\ -1\end{pmatrix}$$

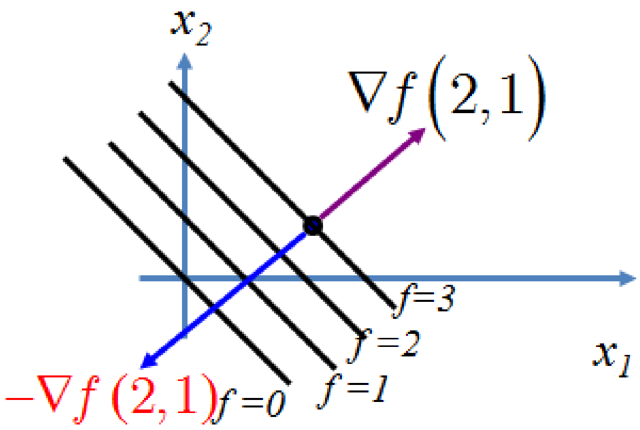
多元函数--梯度

梯度的几何解释

给定一个函数 $f(x)=x_1+x_2$,

在点 $x=(2,1)$处，梯度为：

$$\nabla f\left(2,1\right)=\left(1,1\right)^T$$



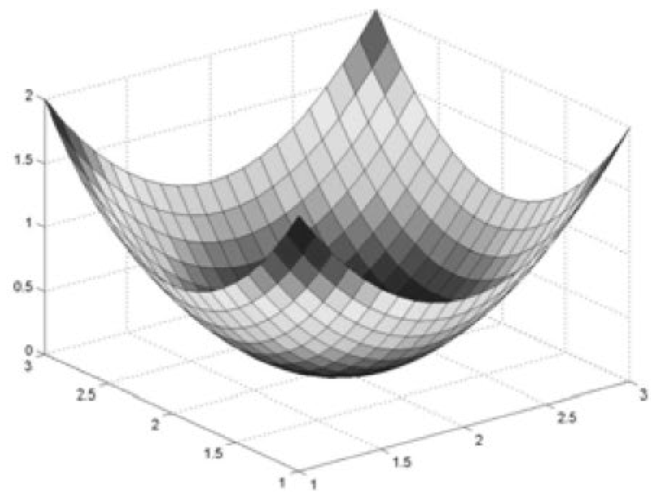
✓ 观察发现：

1. 当沿着点x的梯度方向移动时，函数值将增加；
2. 当沿着点x的负梯度方向移动时，函数值将减小。

对于使目标函数值最小的问题，负梯度方向为其提供了一个重要的线索。如何证明？

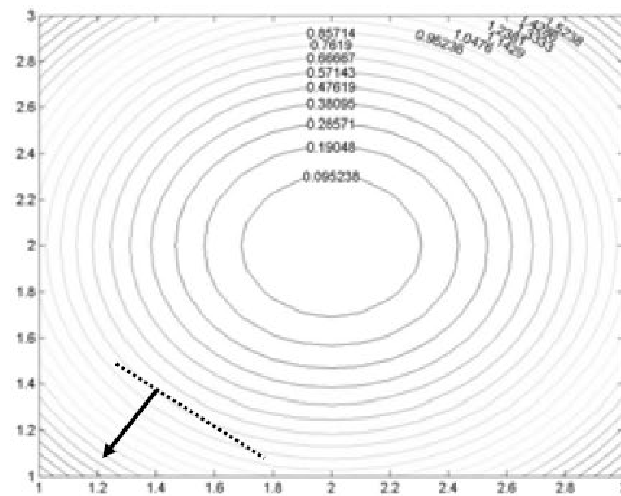
多元函数--梯度

❑梯度的几何解释



$$z=(x_1-2)^2+(x_2-2)^2$$

$$\nabla z(1.4,1.4)=(-1.2,-1.2)$$



Representation in contours

多元函数—海塞矩阵

- **n元函数$f(x)$在最优解 $x=x^*$ 处的海塞矩阵可表示为：**

$$\nabla^2 f(x^*) = \begin{bmatrix} \partial^2 f(x^*)/\partial^2 x_1 & \partial^2 f(x^*)/\partial x_1 \partial x_2 & \cdots & \partial^2 f(x^*)/\partial x_1 \partial x_n \\ \partial^2 f(x^*)/\partial x_2 \partial x_1 & \partial^2 f(x^*)/\partial^2 x_2 & \cdots & \partial^2 f(x^*)/\partial x_2 \partial x_n \\ \vdots & \vdots & \vdots & \vdots \\ \partial^2 f(x^*)/\partial x_n \partial x_1 & \partial^2 f(x^*)/\partial x_n \partial x_2 & \cdots & \partial^2 f(x^*)/\partial^2 x_2 \end{bmatrix}_{n\times n}$$

- **例**

(1) $f_1(x) = (x_1 - 2)^2 + (x_2 - 2)^2$

$$\nabla^2 f(x^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

(2) $f_2(x) = 2x_1 + 3x_2 + 0.5x_3 - x_4$

$$\nabla^2 f_2(x^*) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

矩阵正定的定义

A是一个$n \times n$ 的矩阵

- 正定矩阵

对于任意非零向量 x, 矩阵A 都满足 $x^TAx > 0$

- 半正定矩阵

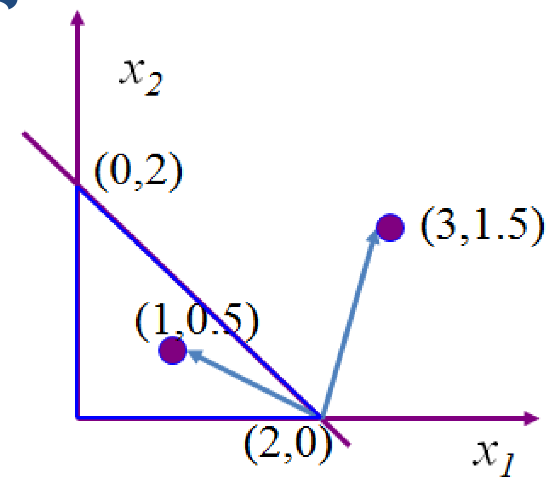
对于任意非零向量 x, 矩阵A 都满足 $x^TAx \geq 0$

- 正定的充要条件

如果矩阵A的$k(k=1,2,\dots,n)$阶顺序主子式都大于0，那么矩阵A是正定的

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \quad |a_{11}| > 0 \quad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0 \quad \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} > 0$$

基础知识：NLP的可行方向



$$\min_{x\in\Omega} f(x) \quad \text{where } \Omega \subseteq R^n$$

- 点x处的可行方向 d

当一个方向d (即一个n维向量) 是可行域Ω内点 x处的可行方向时，那么存在参数α使方向d 满足：

$$x+\alpha d \in \Omega \text{ for all } 0 \le \alpha \le \bar{\alpha}$$

例: $\Omega=\{(x_1,x_2)|x_1+x_2\le 2,\ x_1\ge 0,\ x_2\ge 0\}$

✓ 对于内点 (1,0.5), 任意一个方向都是可行方向(Why?)

✓ 对于顶点 (2, 0), 可行方向: d = (1,0.5) - (2,0) = (-1,0.5)

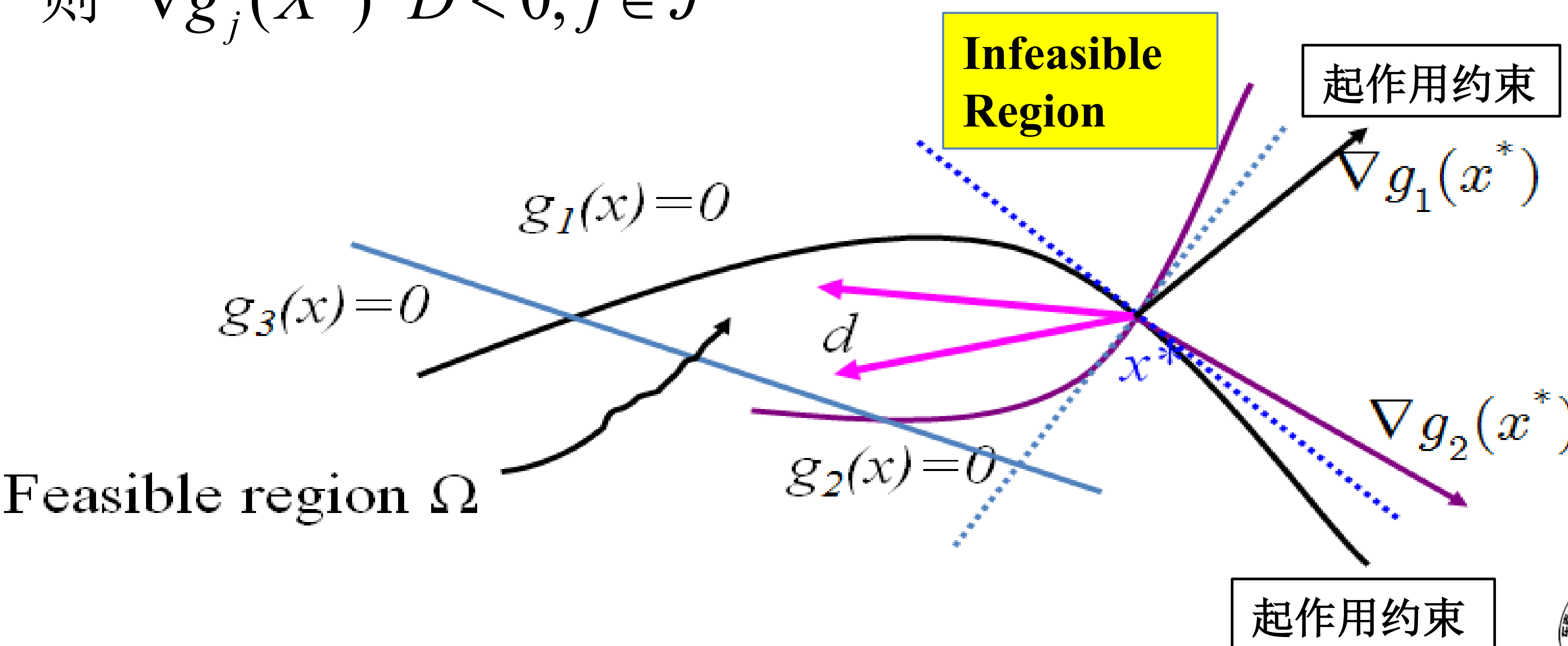
不可行方向: d = (3, 1.5) - (2,0) = (1, 1.5)

基础知识： NLP的可行方向

- 起作用约束（binding constraints）

$$g_i(\mathbf{X}) \le 0 \quad i = 1, 2, \ldots, m$$

假设 D 是点 X^* 处的可行方向；j是点 X^* 处起作用的约束，即 $g_j(X^*) = 0$。假设 J 是点 X^* 处所有起作用约束的集合，则 $\nabla g_j(X^*)^T D < 0, j \in J$



基础知识：　NLP的可行方向

$$X = X^* + \lambda D$$

λ:步长

证明：用泰勒展开

$$f(X) = f(X^*) + \nabla f\left(X^*\right)^{\mathrm{T}}\left(X - X^*\right) + o(X - X^*)$$

$$g_j(X^* + \lambda D) = g_j(X^*) + \lambda \nabla g_j(X^*)^T D + o(\lambda)$$

因为 $g_j(X^*)=0$

而且，步长$\lambda > 0$，$g_j(X^* + \lambda D) \le 0, j \in J$

所以 $\nabla g_j(X^*)^T D < 0, j \in J$

对于不起作用约束：即 $g_i(X^*) < 0, i \notin J$

对于任意方向D，总存在 $\lambda > 0$ ，满足 $g_i(X^* + \lambda D) \le 0, i \notin J$

基础知识： NLP的下降方向

❑NLP的下降方向

- 如果向量D是点$X^{(0)}$处的下降方向, 那么存在实数λ >0, 满足

$$f(X^{(0)}+\lambda D)<f(X^{(0)})$$

- 对于任意下降方向，我们进一步可得到$\nabla f(X^{(0)})^T D<0$
- 证明： 由$f(X)$ 在点 $X^{(0)}$处的泰勒展开，可得：

$$f(X^{(0)}+\lambda D)=f(X^{(0)})+\nabla f\left(X^{(0)}\right)^{\mathrm{T}}\lambda D+o(X-X^{(0)})$$

$$f(X^{(0)}+\lambda D)-f(X^{(0)})<0\Rightarrow\nabla f\left(X^{(0)}\right)^{\mathrm{T}}\lambda D<0$$

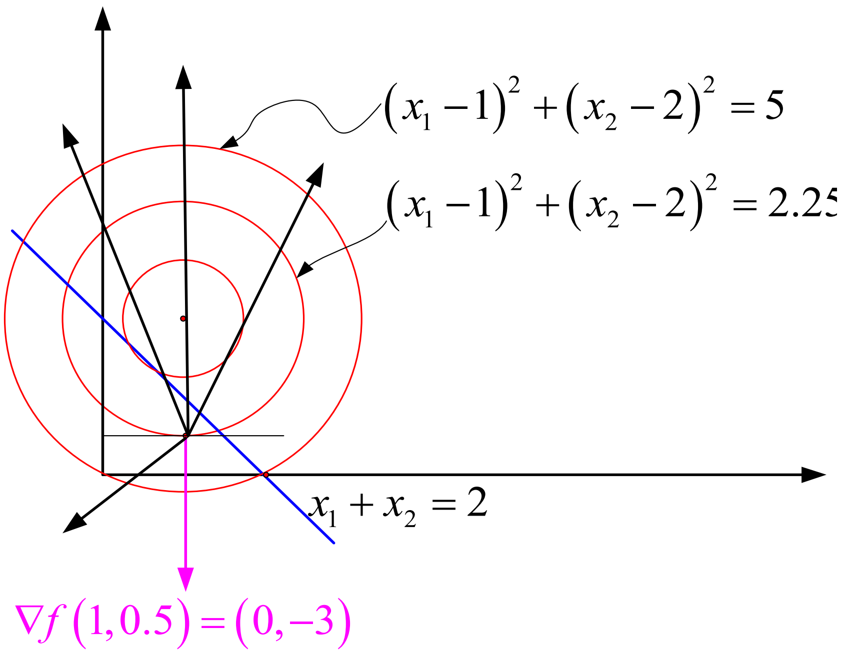
$$\nabla f(X^{(0)})^T D<0$$

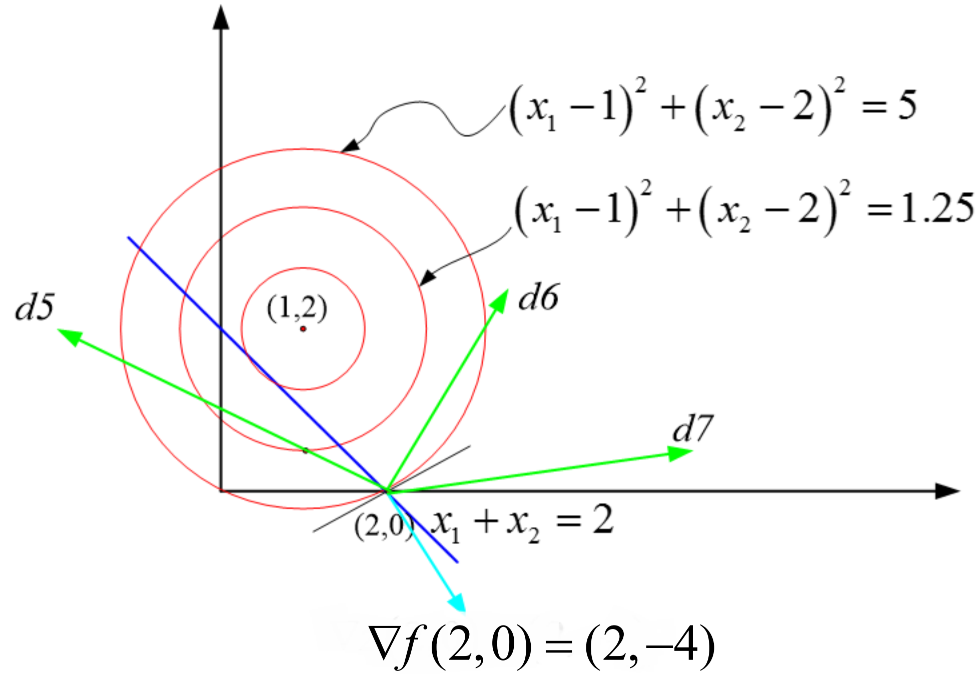
重要的结论:
负梯度方向是最速下降方向

基础知识： NLP的下降方向

NLP的下降方向

例: $\Omega=\{(x_1,x_2)|x_1+x_2\le 2,\ x_1\ge 0,\ x_2\ge 0\}$, $f(x)=(x_1-1)^2+(x_2-2)^2$





在可行域内部点 (1,0.5)处

可行下降方向为 d_1, d_2, d_3

在可行域边界点(2,0)处

可行下降方向为 d_5

基础知识： NLP的下降方向

❑NLP的可行下降方向

从字面上讲，**可行下降方向**即同时满足可行性和使函数值减小的方向

- 如果 $X^{(0)}$ 不是一个局部极小值点，那么求解算法下次迭代的搜索方向应该是该点的可行下降方向；
- 如果 $X^{(0)}$ 是一个极小值点，那么该点没有可行下降方向；
- 反过来讲，如果一个点有可行下降方向，那么该点一定不是局部极小值点

基础知识： NLP的下降方向

▫NLP的可行下降方向

从数学表达来看，如果点 $X^{(0)}$不是局部极小值点，那么一定存在一个方向D满足如下不等式条件：

$$\begin{cases} \nabla f(X^{(0)})^T D < 0 \\ \nabla g_j(X^{(0)})^T D < 0, j \in J \end{cases}$$

从坐标空间来看，

- 可行下降方向D和目标函数的负梯度方向的夹角是锐角；
- 可行下降方向D和起作用约束的负梯度方向的夹角是锐角；

凸优化
（Convex Optimization）

凸优化

- 凸优化（也称：凸规划，Convex Programming）
- 凸优化是非线性规划的一个特殊子集，比一般的NLP模型更容易求解。需满足两个条件：
  - 可行域为凸集
  - 目标函数为凸函数

$$\min_{x\in\Omega} f(x)$$

where

$$\Omega = \{x \mid g(x) \le 0, h(x) = 0\}$$

凸集（Convex Set）

设 Ω 表示n维实数域 R^n 内的一个集合

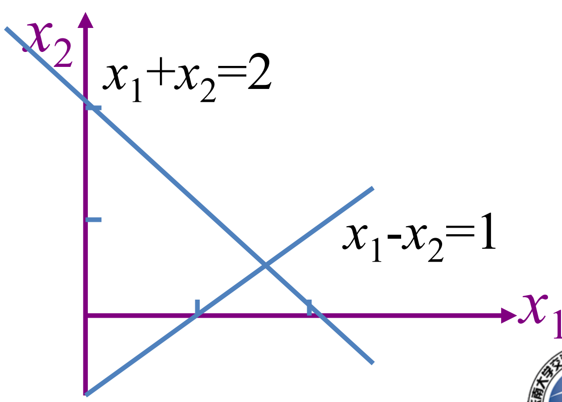
- **定义**

集合内任意两点 $x_1, x_2 \in \Omega$，任意实数 $\alpha(0 \le \alpha \le 1)$ 使 $\alpha x_1 + (1-\alpha)x_2 \in \Omega$，则称集合 Ω 为凸集。

即，集合内任意两点的<u>连线间的点</u>都在集合内。

例1：

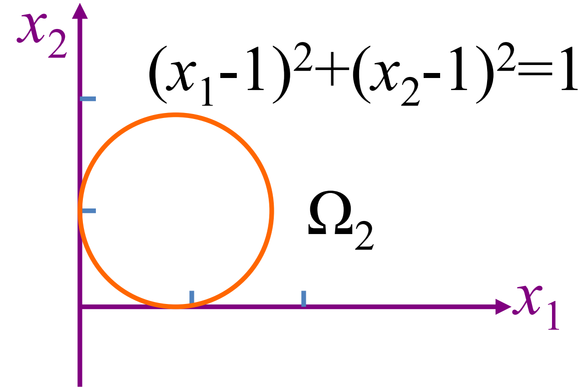
$\Omega_1=\{(x_1,x_2)|\ x_1+x_2\le 2;\ x_1-x_2\le 1;\ x_1\ge 0; x_2\ge 0\}$



凸集

- 例2：

$\Omega_2=\{(x_1,x_2)|\ (x_1-1)^2+(x_2-1)^2\le 1\}$

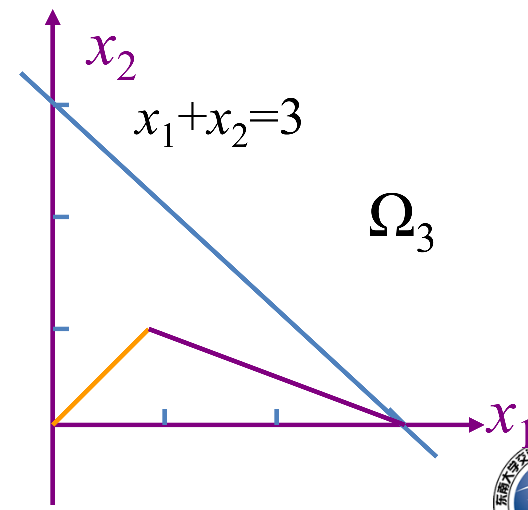


- 例3：

$\Omega_3=\{(x_1,x_2)|\ x_1+x_2\le 3;\ g(x)<=0; x_1\ge 0; x_2\ge 0\}$

其中

$$g(x)=\begin{cases} x_1-x_2, & 0\le x_1\le 1 \\ 3-x_1-2x_2, & x_1\ge 1 \end{cases}$$

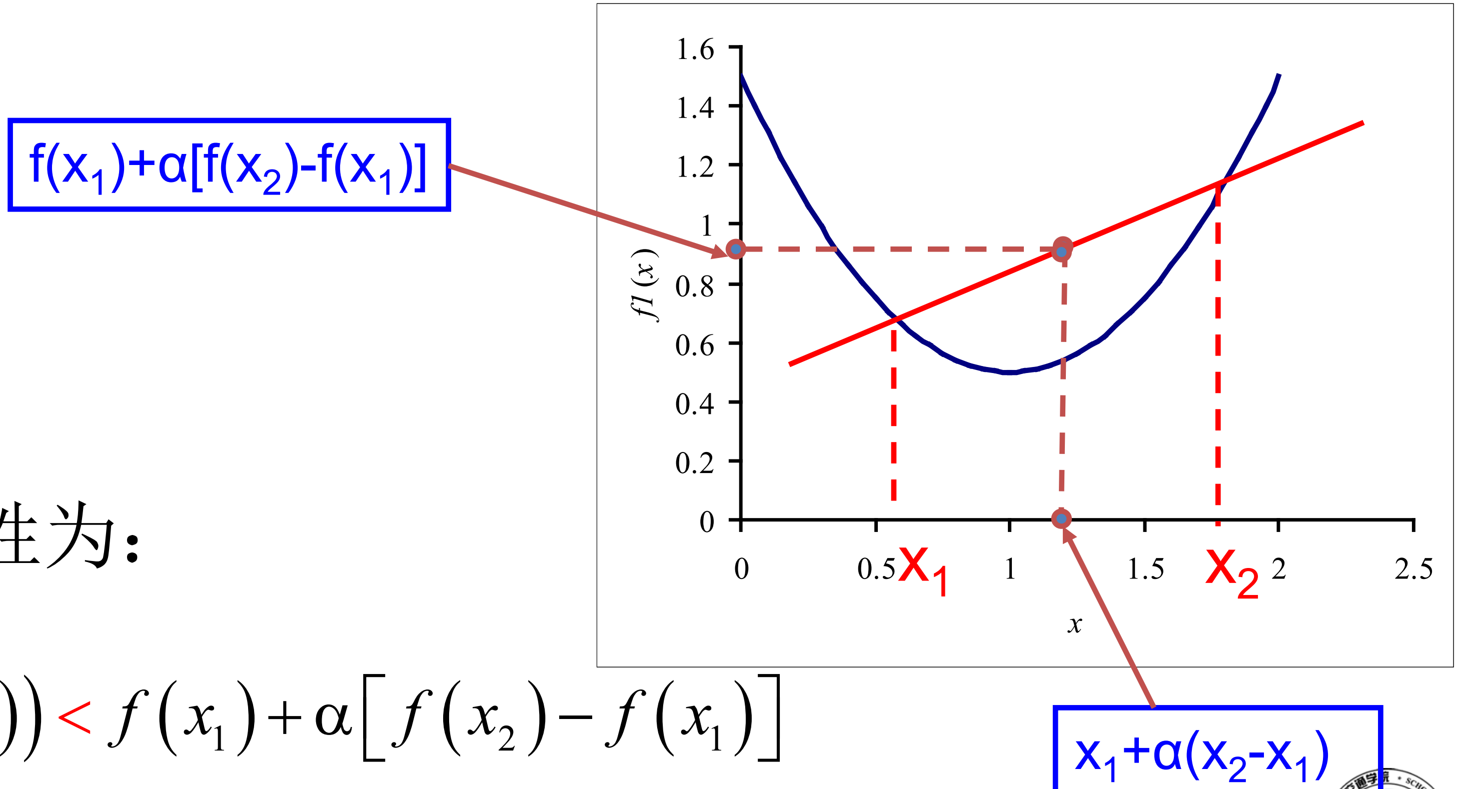


凸函数(Convex Function)：定义

函数 $f(x)$ 在凸集合Ω内的凸性为：

$$f\left(x_1+\alpha\left(x_2-x_1\right)\right)\le f\left(x_1\right)+\alpha\left[f\left(x_2\right)-f\left(x_1\right)\right],\forall x_1,x_2\in\Omega,\forall\alpha\in\left[0,1\right]$$

$$f: R^n \to R$$



- 其严格凸性为：

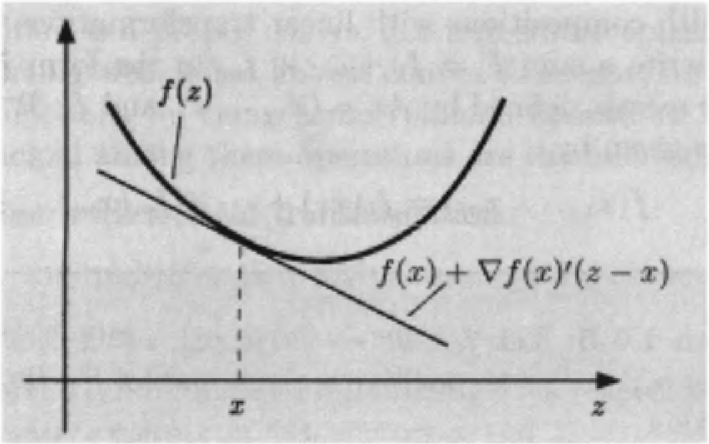
$$f\left(x_1+\alpha\left(x_2-x_1\right)\right)< f\left(x_1\right)+\alpha\left[f\left(x_2\right)-f\left(x_1\right)\right]$$

$$\forall x_1,x_2\in\Omega,\forall\alpha\in\left[0,1\right]$$

凸函数：一阶等价条件

如果 $f: R^n \to R$

在凸集合Ω内的任一点可微



▪**充分/必要条件**

(i) f 在凸集合Ω内是凸函数的一阶等价条件是：

$$f(z) \geq f(x) + \nabla f(x)^{\mathrm{T}}(z-x), \forall x, z \in \Omega$$

(ii) f 在凸集合Ω内是严格凸函数的一阶等价条件是：

$$f(z) > f(x) + \nabla f(x)^{\mathrm{T}}(z-x), \forall x \neq z, x, z \in \Omega$$

凸函数：二阶等价条件

❑ 检验函数凸性的条件

假设对于凸集合Ω内的任意可行点处的海塞矩阵$\nabla^2 f(x)$都存在

- f在凸集合Ω内是凸函数的二阶等价条件是

  (i) 当且仅当海塞矩阵$\nabla^2 f(x)$是半正定矩阵时，函数$f(x)$是凸函数；

  (ii)当且仅当海塞矩阵$\nabla^2 f(x)$是正定矩阵时，函数$f(x)$是严格凸函数；

凸函数

- **例**
  - 函数 $f_1(x)=(x_1-2)^2+(x_2-2)^2$

    它是一个凸函数

  - 函数 $f_2(x)=2x_1+3x_2+0.5x_3-x_4$

    它是一个凸函数但不是严格凸函数

  - 注意：

    所有的线性函数都是凸函数。是不是凹函数？

凸函数

- 例 1:

  证明 $f(x) := \sum_{j=1}^{n} c_j x_j$ 在它的自然域内是凸函数

- 例 2:

  假设 $c_j > 0, j = 1, 2 \cdots n.$

  证明 $f(x) := \sum_{j=1}^{n} c_j x_j{}^2$ 是一个凸函数(提示: $2ab \le a^2 + b^2$)

- 例 3:

  试举例写出 c_j（$j = 1, 2 \cdots n$）的值,

  满足 $f(x) := \sum_{j=1}^{n} c_j x_j{}^2$ 不是凸函数

凸函数

关于凸优化问题的阐述:

- 如果局部极小值点存在, 那么该局部极小值点是全局极小值点.
- 所有（全局）极小值点的集合是凸集
- 对于每个严格凸函数， 如果该函数有一个极小值，那么该极小值是唯一的。

例:

- 最小二乘法
- 线性规划
- 受线性约束的凸二次最优化
- 锥优化
- 有约束的熵最大化问题

凸优化问题的极小值和极大值

局部极小值点

如果 x^* 是凸集Ω内的一点，对于点x^*处的任意可行方向d 满足$\nabla f(x^*)^T d \geq 0$, 那么 x^* 是凸规划问题$\min_{x\in\Omega} f(x)$的一个全局极小值点

证明：

假设 x^* 不是全局极小值点, 那么在集合内Ω存在点y 满足 $f(y)<f(x^*)$.

因为 Ω 是凸集, 对于任意$0\leq\alpha\leq 1$, $x^*+\alpha(y-x^*)$ 仍在凸集Ω内. 换句话说, $y-x^*$ 是点x^*处的可行方向。

因为$f(x)$是凸函数, 那么可得

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

这与假设矛盾

KKT条件

KKT条件

- 怎样**证明**x^*是下列非线性规划的局部极小值点：

$$
\begin{aligned}
&\min f(x) \\
\text{s.t.}\ & \\
&g_i(x) \le 0,\ \ i = 1,\cdots,m_1 \qquad u_i \\
&h_j(x) = 0, j = 1,\cdots,m_2 \qquad \lambda_j
\end{aligned}
$$

或者，怎样有效地找到局部极小值?

KKT条件

□ 必要最优性条件

如果 x^*是一个局部极小值点, 那么存在拉格朗日乘子$\{u_1, u_2, \ldots, u_{m1}\}$ 和 $\{\lambda_1, \lambda_2, \cdots \lambda_{m_2}\}$满足下列KKT条件:

$$(1)\nabla f\left(x^*\right)+\sum_{i=1}^{m_1} u_i \nabla g_i\left(x^*\right)+\sum_{j=1}^{m_2} \lambda_j \nabla h_j\left(x^*\right)=0 \longleftarrow$$ (KKT等式)

$$(2)\, u_i \geq 0, i=1, \cdots, m_1 \longleftarrow$$ (非负约束)

$$(3)\, u_i g_i\left(x^*\right)=0, i=1, \cdots, m_1 \longleftarrow$$ (互补松弛条件)

$$(4)\, g_i\left(x^*\right) \leq 0, i=1, \cdots, m_1, h_j\left(x^*\right)=0, j=1 \cdots, m_2 \longleftarrow$$ (可行性)

约束条件 (CQ), 或正则条件

- 当KKT条件成立时, $g_i(x)$和$h_j(x)$必须满足一定的正则条件 (通常也成为***constraint qualifications***).
- 最常用的CQs如下：
- **线性约束条件(LCQ):**
  如果 $g_i(x)$和$h_j(x)$是仿射函数，则不需要其他的条件成立.
- **线性独立约束条件 (LICQ):**
  $\nabla g_i(x)$和$\nabla h_j(x)$是线性独立的.
- ……

Reference:

- Chapter 5 of *Nonlinear Programming: Theory and Algorithms* by Bazaraa et al. (1993)
- Eustaquio, R. G., Karas, E. W., & Ribeiro, A. A. (2010). Constraint qualifications for nonlinear programming.

KKT 条件

- 注意
  - KKT条件是NLP获得局部极小值的必要条件
  - KKT条件是凸规划问题获得局部极小值的充要条件
    - 对于凸规划问题，任意一个局部极小值点就是全局极小值点。
    - 因此，KKT条件是凸规划问题获得全局极小值的充要条件。

KKT 条件

- 例1

检查 x*=(0.5,1.5) 是不是如下最小化问题的唯一全局最优解：

$$\min f(x) = (x_1 - 2)^2 + (x_2 - 3)^2$$

subject to

$$x_1 + x_2 \le 2 \quad \longleftarrow \quad u_1$$

$$x_1 \ge 0 \quad \longleftarrow \quad u_2$$

$$x_2 \ge 0 \quad \longleftarrow \quad u_3$$

(1) 检验可行性条件

$$x_1^* + x_2^* = 2$$

$$x_1^* = 0.5 > 0$$

$$x_2^* = 1.5 > 0$$

KKT 条件

(2) 检验互补松弛条件

$$u_2 \times \left(-x_1^*\right) = 0, \quad u_3 \times \left(-x_2^*\right) = 0, \quad u_1^* \times \left(x_1^* + x_2^* - 2\right) = 0$$

可以知道 $u_2=0, u_3=0$ ，而u_1 是一个未知的非负乘子

(3) 解KKT等式

$$\begin{pmatrix} 2\left(x_1^* - 2\right) \\ 2\left(x_2^* - 3\right) \end{pmatrix} + u_1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} + u_2 \begin{pmatrix} -1 \\ 0 \end{pmatrix} + u_3 \begin{pmatrix} 0 \\ -1 \end{pmatrix} = 0 \quad \Rightarrow \quad u_1 = 3$$

(4) 检验非负条件

$$u_1 = 3 > 0, u_2 = 0, u_3 = 0$$

海塞矩阵 $\nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ 为正定，所以原问题为凸优化问题。

因此，x*=(0.5,1.5) 是原问题的唯一全局最优解。

KKT条件

例2：用KKT条件解如下非线性规划问题

$$\begin{cases} \min f(x) = (x-3)^2 \\ 0 \le x \le 5 \end{cases}$$

解： 先将该非线性规划问题写成以下形式：

$$\begin{cases} \min f(x) = (x-3)^2 \\ \quad g_1(x) = -x \le 0 \\ \quad g_2(x) = x - 5 \le 0 \end{cases}$$

算出其目标函数和约束函数的梯度：

$$\nabla f(x) = 2(x-3),$$

$$\nabla g_1(x) = -1, \nabla g_2(x) = 1$$

KKT条件

- 给出两个不等式约束的乘子

$$\begin{cases} \min f(x) = (x-3)^2 \\ \quad g_1(x) = -x \le 0 \\ \quad g_2(x) = x-5 \le 0 \end{cases}$$

两个约束的乘子： γ_1 γ_2

- 该问题的可行域为凸集,其目标方程为凸函数(求海塞矩阵可知)，因此该问题是一个凸优化问题。

KKT条件

根据KKT条件，某个局部最优解 $x^*, \gamma_1^*, \gamma_2^*$ 应该满足如下条件

$$
\begin{aligned}
&2(x^* - 3) - \gamma_1^* + \gamma_2^* = 0 && \gamma_1^*, \gamma_2^* \geq 0 \\
&\gamma_1^* x^* = 0 && -x^* \leq 0 \\
&\gamma_2^*(x^* - 5) = 0 && x^* - 5 \leq 0
\end{aligned}
$$

为解上述方程组，分以下几种情形讨论：

（1）令 $\gamma_1^* \neq 0, \gamma_2^* \neq 0,$ 无解；

（2）令 $\gamma_1^* \neq 0, \gamma_2^* = 0,$ 解之，得 $x^* = 0, \gamma_1^* = -6,$ 不满足KKT条件；

（3）令 $\gamma_1^* = 0, \gamma_2^* \neq 0,$ 解之，得 $x^* = 5, \gamma_2^* = -4,$ 不满足KKT条件；

（4）令 $\gamma_1^* = \gamma_2^* = 0,$ 解之，得 $x^* = 3,$ 满足KKT条件,得到局部最小值点。由于该问题为凸规划，所以此为全局极小值点。

小结：KKT条件

- 目标函数和约束条件必须是连续可微的；
- 可行域需满足CQ，KKT条件才成立。
- 如果一个问题有多个局部最优解，那么就会有许多解都满足KKT条件。
- KKT包括许多等式和不等式，因此当问题规模变大之后，KKT条件很难直接求解。所以对于大规模问题，需要利用有效的算法来求解（下一节课）。

凸规划问题的求解算法

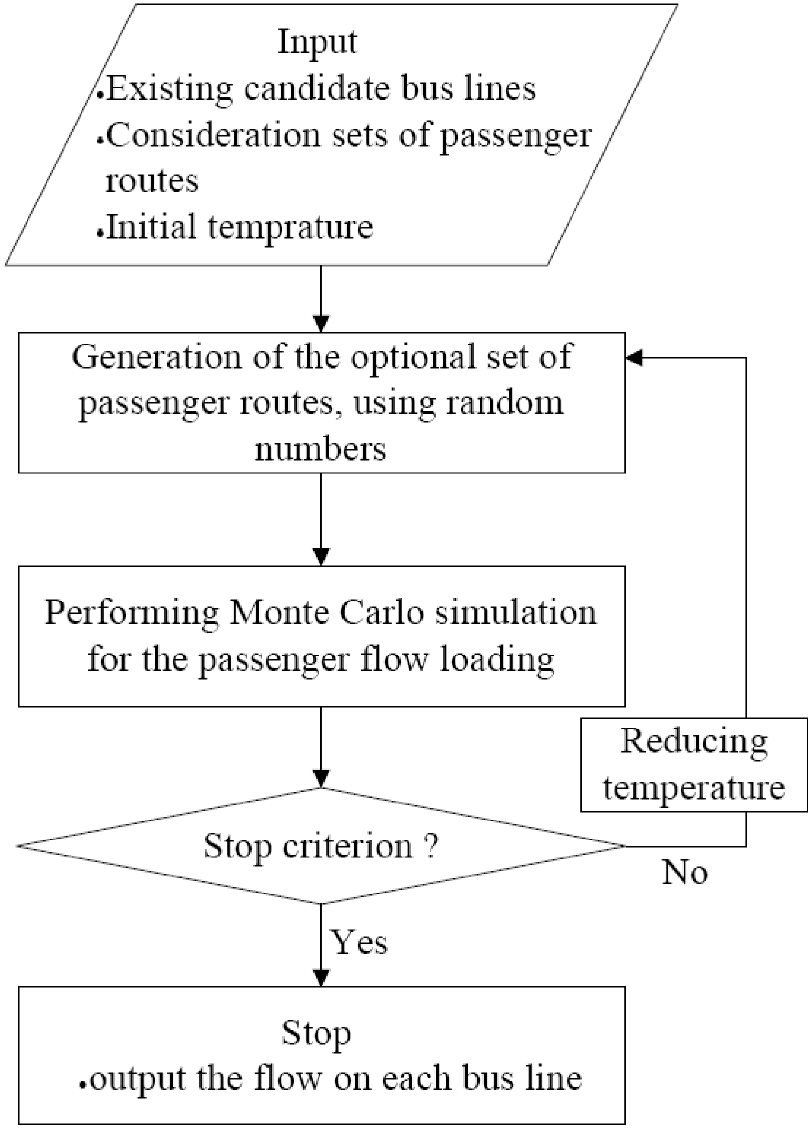
算法(Algorithm) $x^0 \rightarrow x^n \rightarrow x^{n+1} \rightarrow x^*$

- 你对算法的理解是什么？
- 求解数学模型的算法过程包括什么？
  - 输入(Input)
  - 初始化(Initialization)
  - 下降方向(Descent Direction)
  - 一维搜索(Line Search)
  - 停止检验(Stop Check)
    - 最常用的停止规则（Stop Criterion）： $|x_{k+1} - x_k| \leq \varepsilon$

ε 是一个给定的大于零的常数

算法

- 怎样画算法的流程图（Flowchart）？



算法

- 一维搜索法；
- 在确定下降方向后，需要进行线性搜索，以便确定下降方向上的最优点，作为下一个迭代点；
- 一维搜索问题的常用求解算法：
  - 二分法
  - 黄金分割法
  - 牛顿法

二分法

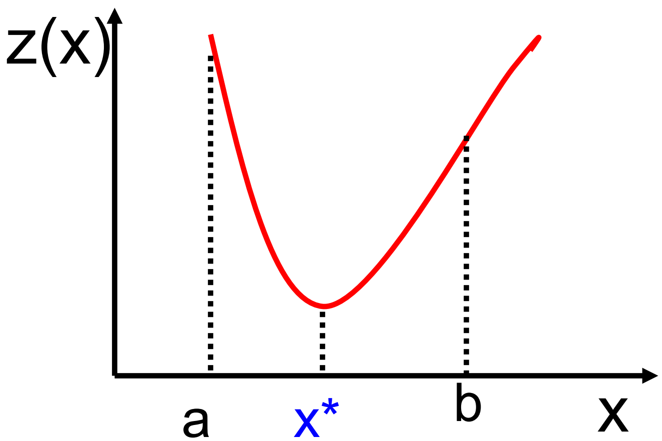
- **假设**

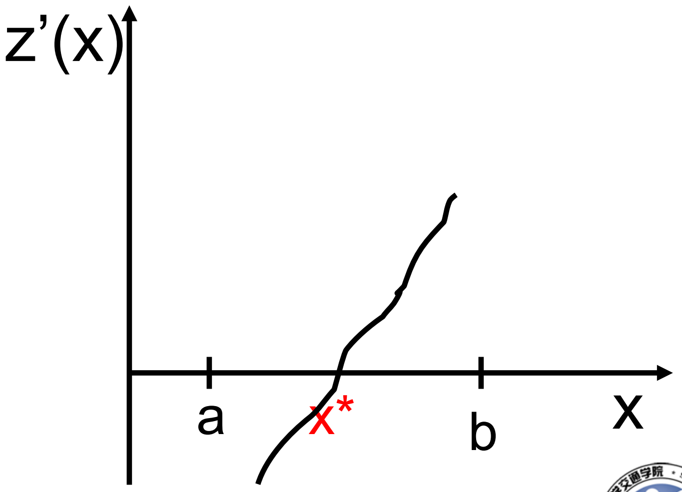
  *z(x)*是凸函数，且在区间[a, b]内是连续可微函数。

- **原理**

  求其导函数z’(x)在区间 [a, b]内的零点（根）, 即

  $z'(x)=0$ with $a \le x \le b$

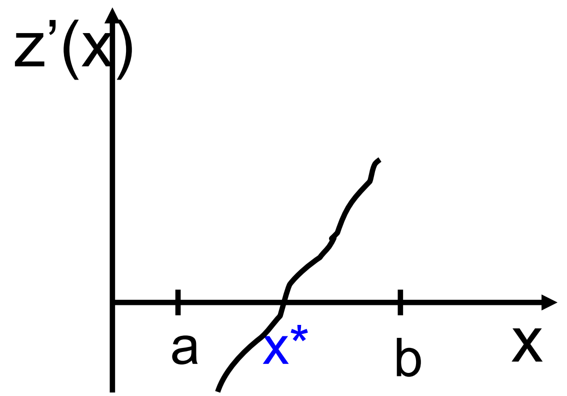




二分法

- 算法步骤

$$\boxed{(b_{k+1}-a_{k+1})=0.5*(b_k-a_k)}$$



Step 0: 给定一个初始解

$[a_0,b_0]\subseteq[a,b]$ 其中 $x^*\in[a,b]$，$z'(a_0)<0, z'(b_0)>0$

令k=0

Step 1: 如果 $b_k-a_k\leq\varepsilon$, 则停止，得到一个极小值$x^*=(b_k+a_k)/2$

否则, 转入 Step 2

Step 2: 令$x_k=(b_k+a_k)/2$

如果 $z'(x_k)\geq0$，则 $b_{k+1}=x_k$, $a_{k+1}=a_k$

如果 $z'(x_k)<0$，则 $a_{k+1}=x_k$, $b_{k+1}=b_k$

令 k=k+1，并返回Step 1.

二分法

- 例1

用二分法解如下非线性规划问题：

$$\min z(x) = \sin x$$

$$s.t.\ 3 \le x \le 6$$

解：

因为： $z'(x) = \cos x, a_0 = 3, b_0 = 6, x_0 = 4.5$

所以： $z'(3) = -0.99 < 0, z'(6) = 0.96 > 0, z'(x_0) = -0.2108$

因此： $a_1 = 4.5, b_1 = 6, x_1 = 5.25$

重复上述过程，迭代过程如下表所示：

二分法

- 例1

二分法的迭代过程

k	a_k	b_k	x_k	$z'(x_k)$	$z'(a_k)$	$z'(b_k)$
0	3	6	4.5	-0.2108	-0.99	0.96017
1	4.5	6	5.25	0.51209	-0.2108	0.96017
2	4.5	5.25	4.875	0.1619	-0.2108	0.51209
3	4.5	4.875	4.6875	-0.0249	-0.2108	0.1619
4	4.6875	4.875	4.78125	0.06881	-0.0249	0.1619
5	4.6875	4.78125	4.73438	0.02198	-0.0249	0.06881
6	4.6875	4.73475	4.71113	-0.0013	-0.0249	0.02236

关于二分法的讨论

- 在 step 2, 为什么我们不考虑$z(x_k)=0$ 的情况?
  - 由于问题的结构和数值误差（problem structure and numerical errors），$z(x_k)=0$的可能性是很低的
- 输出应该选哪一个?
  - $(a_k+b_k)/2$, or a_k or b_k
  - 这不重要，因为可容忍的误差ε一般是很小的。

关于二分法的讨论

- 在停止条件里，如何确定 ε？
  - ε 的取值较大时，将使算法的收敛速度很快，但是精度较低。因此 ε 是在效率和精度的权衡下确定的；
  - 例如，如果x表示零售市场的单位价格，那么ε =1分就是可接受的。
- 其他常用的停止条件有哪些?
  - 要求算法在一个给定的迭代次数后停止；
  - 要求算法在一个给定的CPU时间后停止。

结论

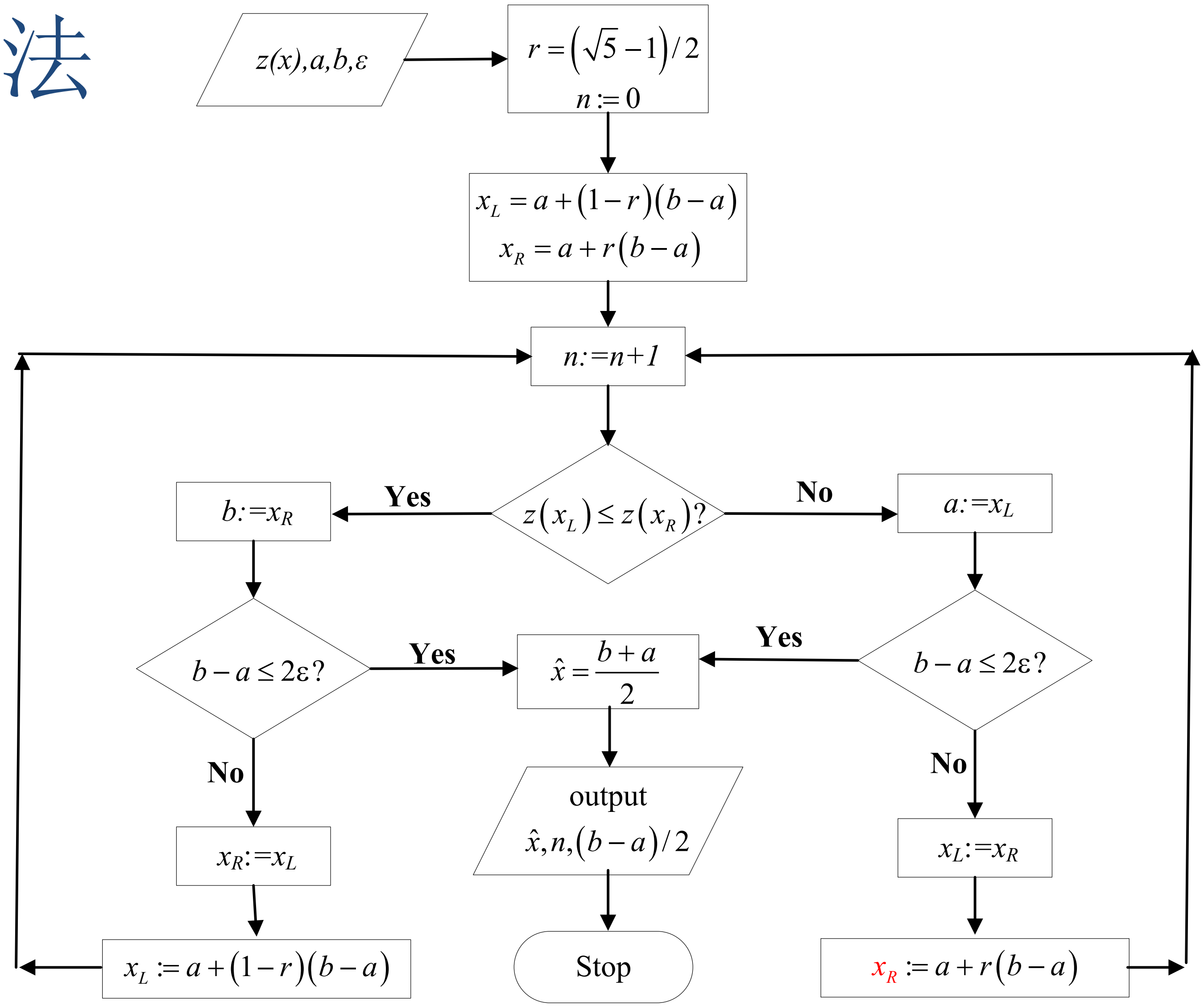
- 二分法是用来求解凸的连续可微函数？对吗？
  - 如果问题是不可微的，二分法就是无效的
- 而黄金分割法不需要原问题是可微的，而且不要求该问题是凸函数。

黄金分割法

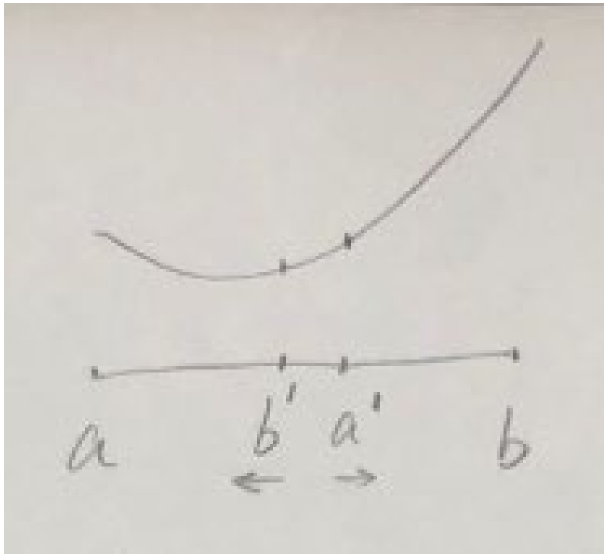
$$\min_{a\le x\le b} z(x)$$

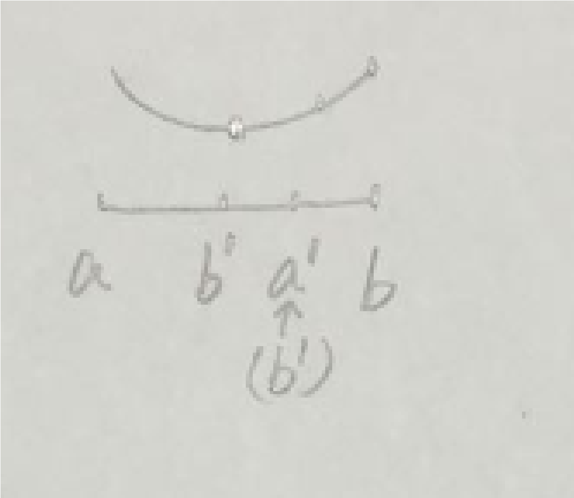
Example:

$$\min_{2\le x\le 4} z(x)=(x-1)^2+1$$



黄金分割法





牛顿法

- 一元非线性规划问题： $\min\limits_{a\le x\le b} f(x)$

  $f: R^1 \to R^1$ 在可行域 $[a,b]$ 内具有连续的一阶和二阶导函数

- 基本原理

因为 $f(x)$ 在$[a,b]$上连续可微，所以原函数的最小值问题等价于求 $f'(x)=0$

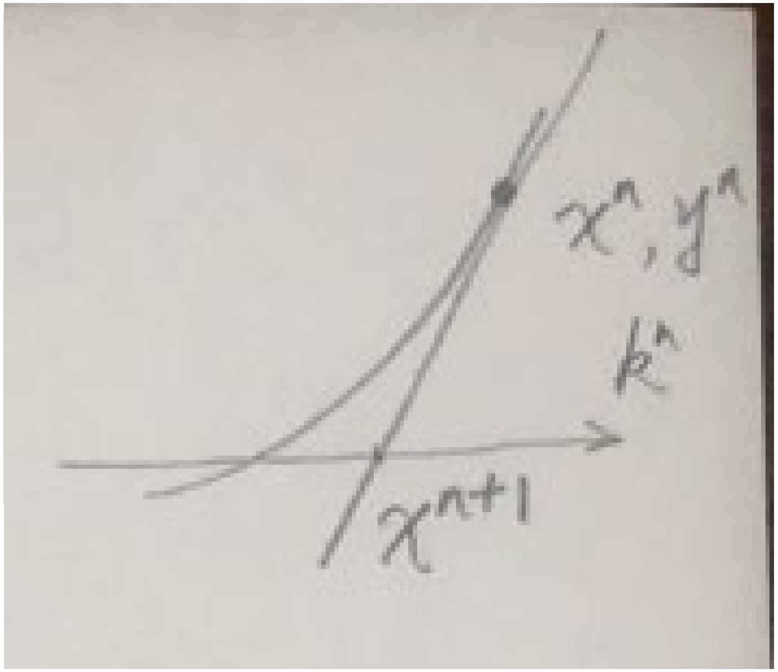
设在区间$[a,b]$中经过k次迭代所得到的变量值为 x_k

过点 $(x_k, f'(x_k))$ 作曲线 $y=f'(x)$ 的切线，其方程是

$$y - f'(x_k) = f''(x_k)(x - x_k)$$

然后用这条切线与横轴交点的横坐标 x_{k+1} 作为根的新的近似点。在此点处 $y=0$，从而得到：

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$ ——此即为牛顿法的迭代公式

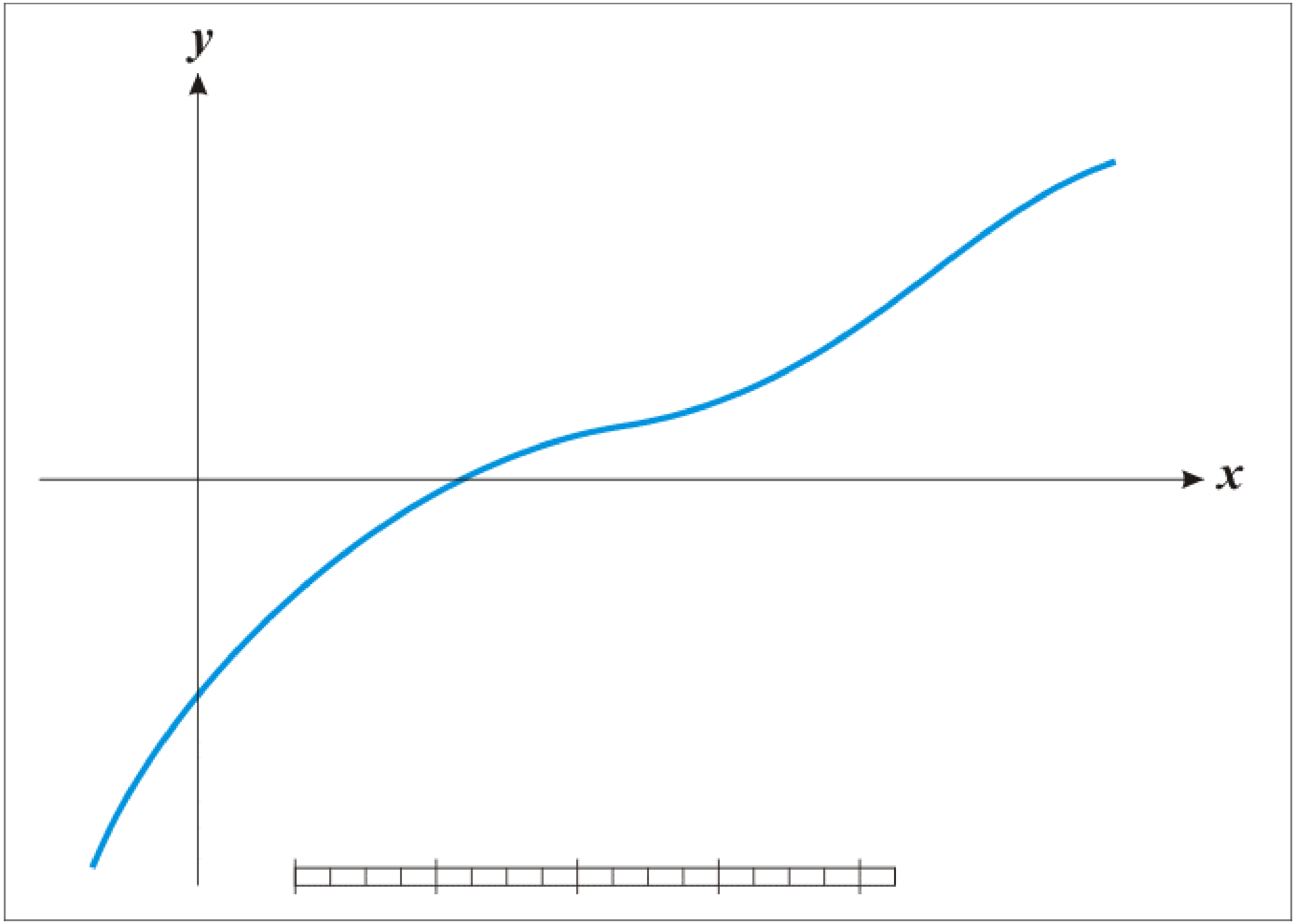


$$y - y^n = k^n(x - x^n)$$

$$x = x^n - \frac{y^n}{k^n}$$

牛顿法

- 基本原理的几何表示



牛顿法

- 算法步骤

已知 $f(x), f'(x)$ 表达式，停止条件 ε

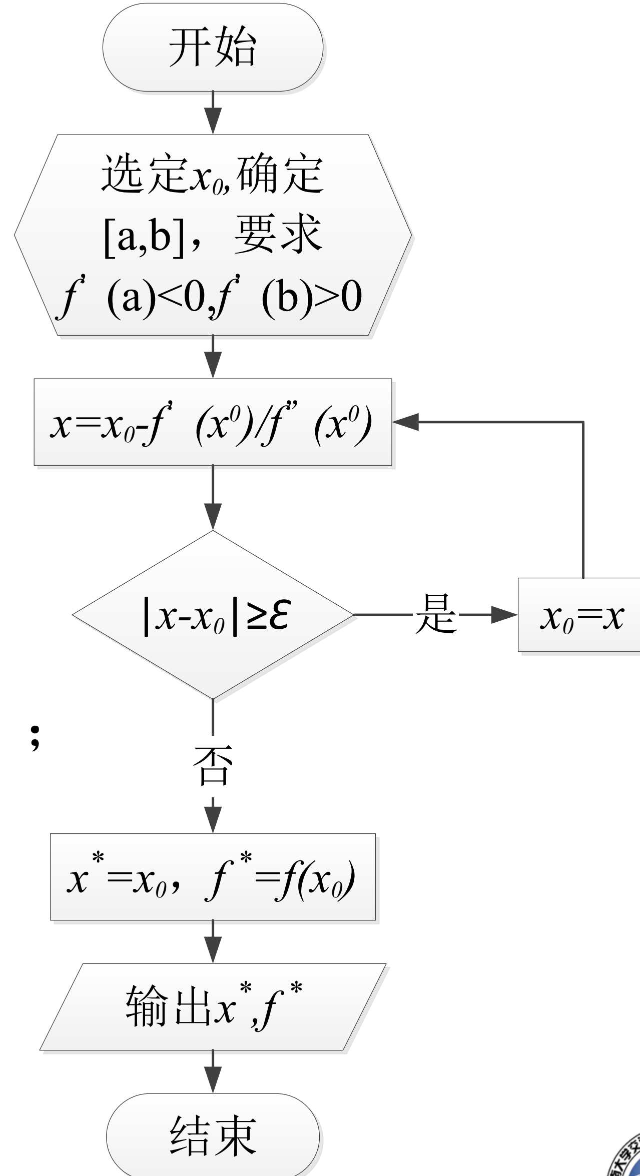
(a) 确定初始搜索区间 [a, b]，要求 $f'(a)<0, f'(b)>0.$

(b) 选定 x_0 ；

(c) 计算 $x=x_0-f'(x_0)/f''(x_0)$ ；

(d) 若 $|x-x_0|\geq\varepsilon$, 则 $x_0=x$, 转入（c）；否则转入（e）；

(e) 输出 $x, f(x)$,结束。



梯度下降法

梯度下降法

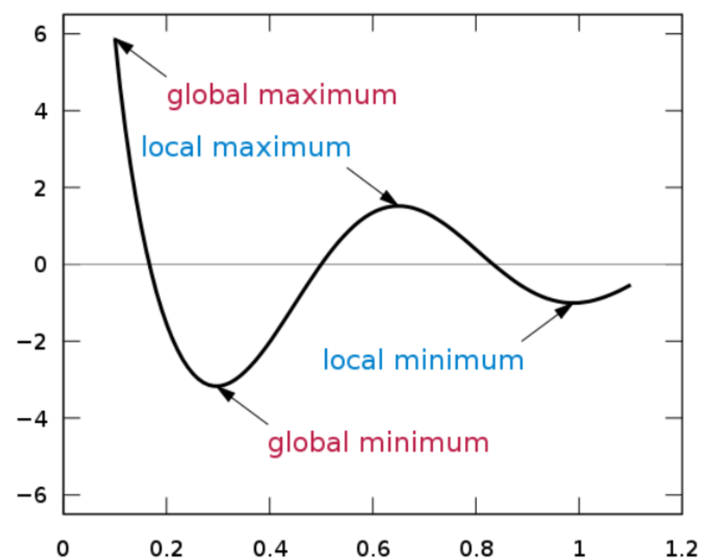
- 问题

确定可微函数 $z(x), x \in R^n$的局部极小值点.

x^* 如果满足$z(x^*) \le z(x), \forall x \in R^n$，则$x^*$ 是$z(x)$ 的一个（全局）极小值点 .

如果存在$\varepsilon > 0$，x^* 满足$z(x^*) \le z(x), \forall x \in R^n$和$\left\| x - x^* \right\| < \varepsilon$,则$x^*$是$z(x)$的局部极小值点

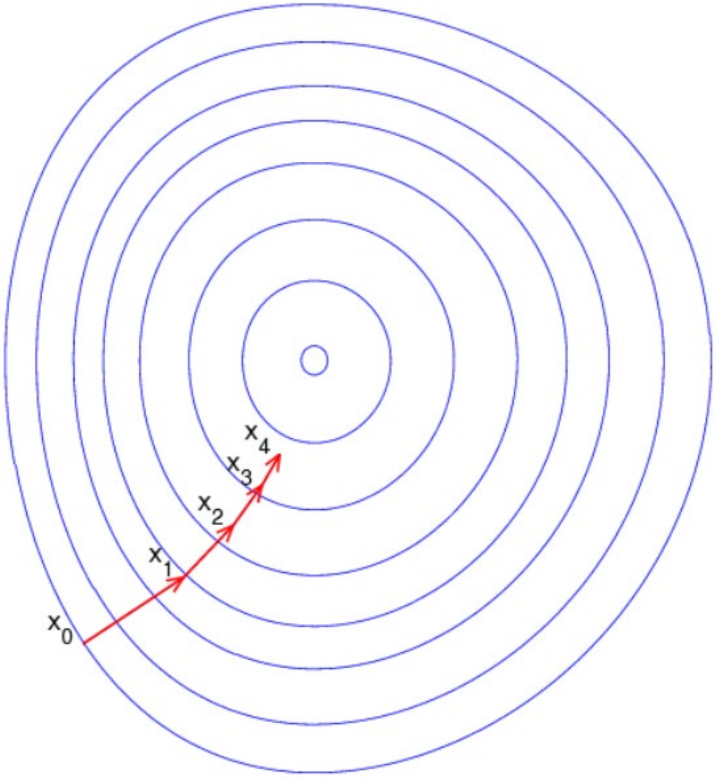
.



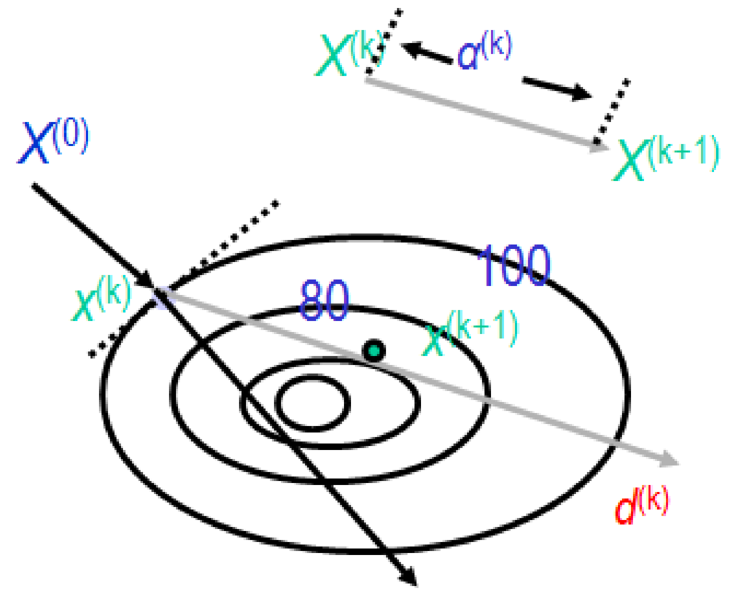
梯度下降法

❑下降方向法的基本原理

- 每次沿着可行下降方向，从一个点移动到另一个点，来使目标函数值减小



梯度下降法



❑下降方向法的基本原理

- Step 0: 选择初始点$x^{(0)}$
- Step 1: 检查停止条件
- Step 2: 确定可行下降方向 $d^{(k)}$
- Step 3:确定与方向$d^{(k)}$ 相关的最优步长 $\alpha^{(k)}$
- Step 4: 令 $x^{(k+1)}=x^{(k)}+\alpha^{(k)}d^{(k)}$

梯度下降法

- 最速下降法/梯度下降法

- 基本原理是用负梯度方向

$$\nabla z\left(x^{(k)}\right):=\begin{pmatrix}\dfrac{\partial z}{\partial x_1^{(k)}}\\ \dfrac{\partial z}{\partial x_2^{(k)}}\\ \vdots\\ \dfrac{\partial z}{\partial x_n^{(k)}}\end{pmatrix}$$

梯度下降法

- 梯度下降法步骤

Step 0: 选择初始点$x^{(0)}$，并令$k=0$，设定$\varepsilon>0$

Step 1: 如果 $|\nabla z(x^{(k)})|<\varepsilon$, 则停止. 否则，转入Step 2

Step 2: 令$d^{(k)}:=-\nabla z(x^{(k)})$，求解一元最小化问题：

$$\min_{\substack{\alpha\geq 0 \\ x^{(k)}+\alpha d^{(k)}\in\Omega}} z\left(x^{(k)}+\alpha d^{(k)}\right)$$来确定$\alpha^{(k)}$

Step 3:令 $x^{(k+1)}=x^{(k)}+\alpha^{(k)}d^{(k)}$, 并令 $k=k+1$, 返回 Step 1.

梯度下降法

❑ 梯度下降法步骤

- 在Step 0, 一般情况下，最好选择接近局部极小值点的初始点(例如，用启发式算法的最优解来确定初始解);
- 在Step 1, 有多种停止条件;
- Step 2的子问题是线性搜索。这比原来的问题更容易求解，在这里我们可以用KKT条件、二分法或黄金分割法来求解 。

梯度下降法

- 例

$$\min_{x\in R^2} z(x)=(x_1-2)^2+10(x_2-2)^2$$

- 梯度 $\nabla z(x)=(2x_1-4,20x_2-40)^T$
- 线性搜索

$$f(\alpha)=z\left(x^{(k)}-\alpha\nabla z\left(x^{(k)}\right)\right)=\left(x_1+\alpha\left(4-2x_1^{(k)}\right)-2\right)^2+10\left(x_2^{(k)}+\alpha\left(40-20x_2^{(k)}\right)-2\right)^2$$

✓令 $f'(\alpha)=0$, 可得:

$$0\le\alpha_k=\frac{(4-2x_1^{(k)})^2+\left(40-20x_2^{(k)}\right)^2}{2(4-2x_1^{(k)})^2+20\left(40-20x_2^{(k)}\right)^2}$$

梯度下降法

- 例

- 迭代方案 (2 iterations)

k	$x^{(k)}$	$d^{(k)}$	$\|d^{(k)}\|$	$\alpha^{(k)}$	$z(x^{(k)})$
0	(-4,-3)	(12,100)	100.7	0.051	286
1	(-3.392,2.065)	(10.784,-1.294)	10.9	0.443	29.118
2	(1.389, 1.491)				

梯度下降法

讨论

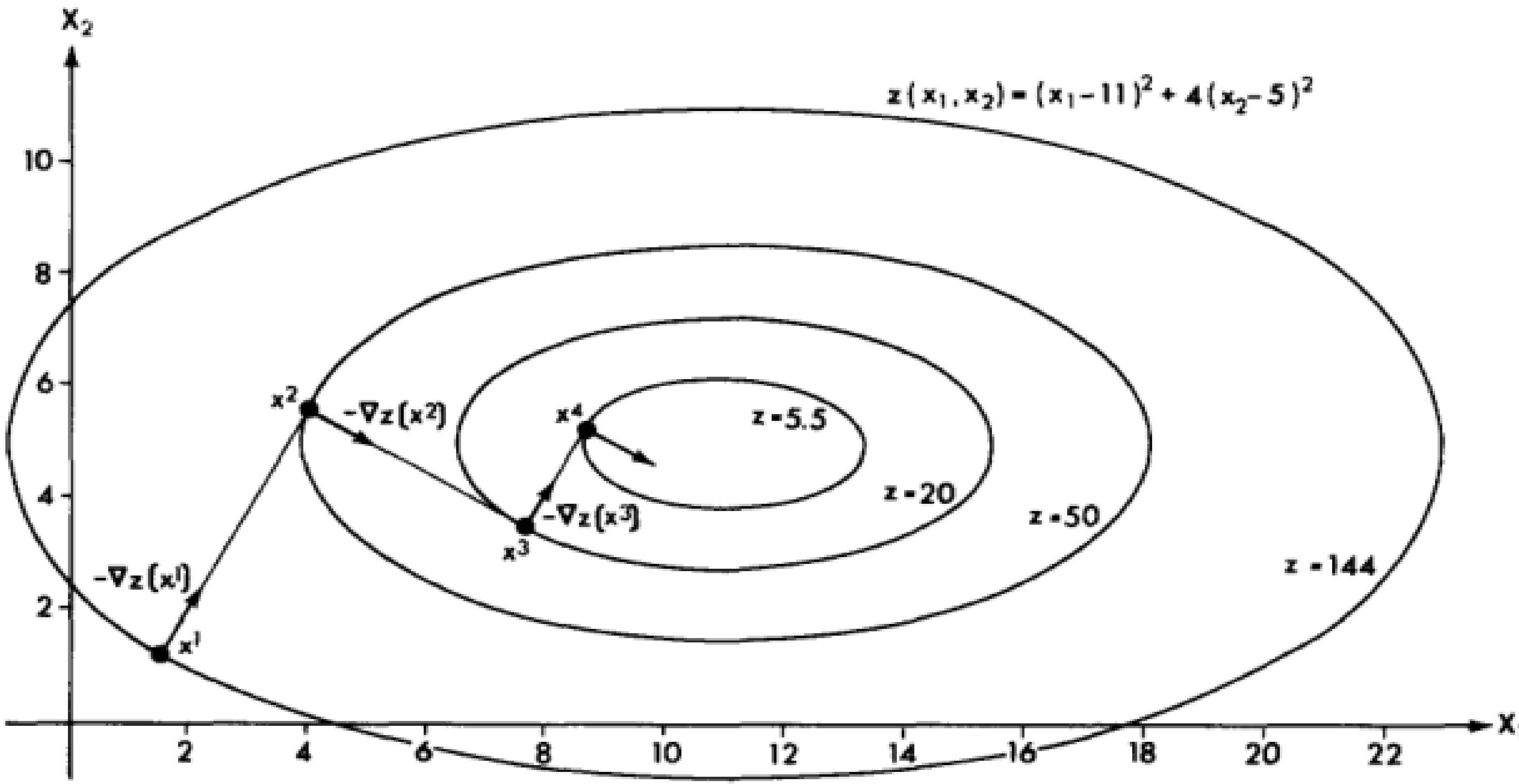
- 当 z(x) 是凸函数时, 梯度下降法将在一个全局最优解处停止
- 由于梯度下降法 “zigzaging” 的性质，该方法收敛速度慢
  - 二次函数
  - 罗森布鲁克函数（ Rosenbrock function ）

$$f(x_1, x_2) = (1 - x_1)^2 + 100[x_2 - (x_1)^2]^2$$

梯度下降法

- 讨论

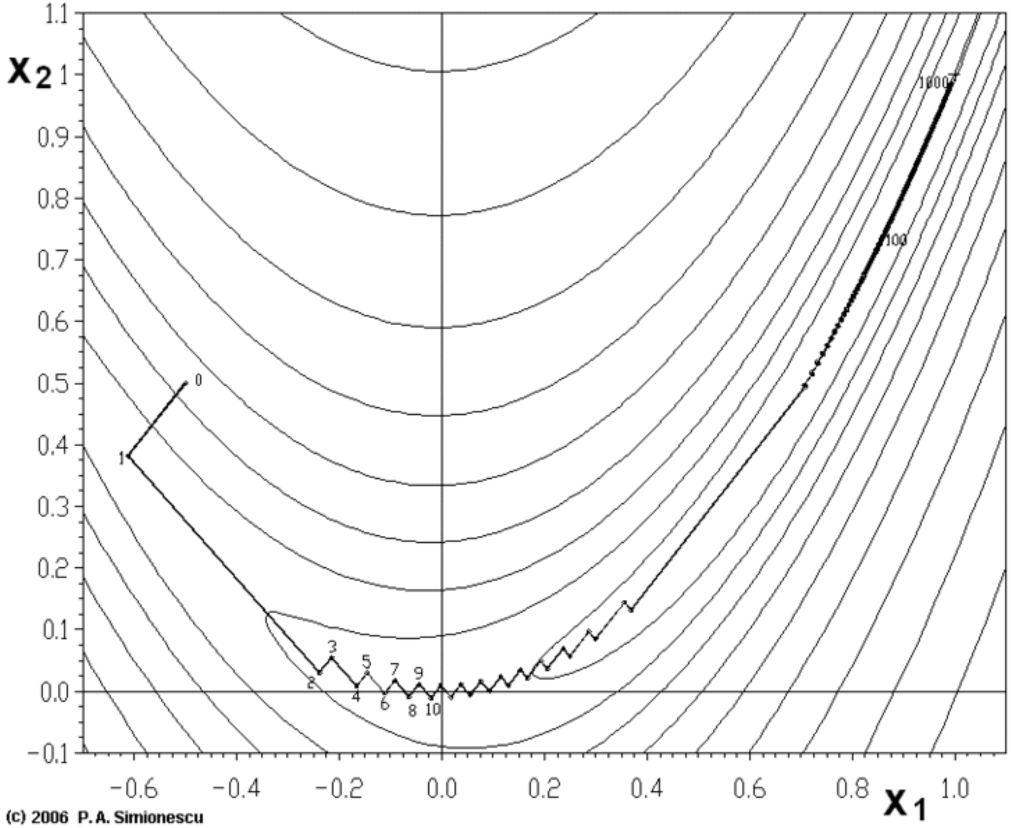
- Zigzaging



梯度下降法

- 讨论

• 罗森布鲁克函数（ Rosenbrock function ）



Frank-Wolfe算法

Frank-Wolfe 算法

- Frank Wolfe Algorithm (convex combination algorithm)

求解下面凸规划问题的最优解：

$$\min_{x} f(x)$$

subject to

$$g(x) \le 0$$

$$h(x) = 0$$

- Frank, M. and Wolfe (1956) An algorithm for quadratic programming. Naval Research Logistics quarterly Research, Vol. 14, pp. 43-53.

Frank-Wolfe 算法

❑ 算法步骤

Step 0: 选择一个初始可行点 $x^{(0)}$， 并令 k=0

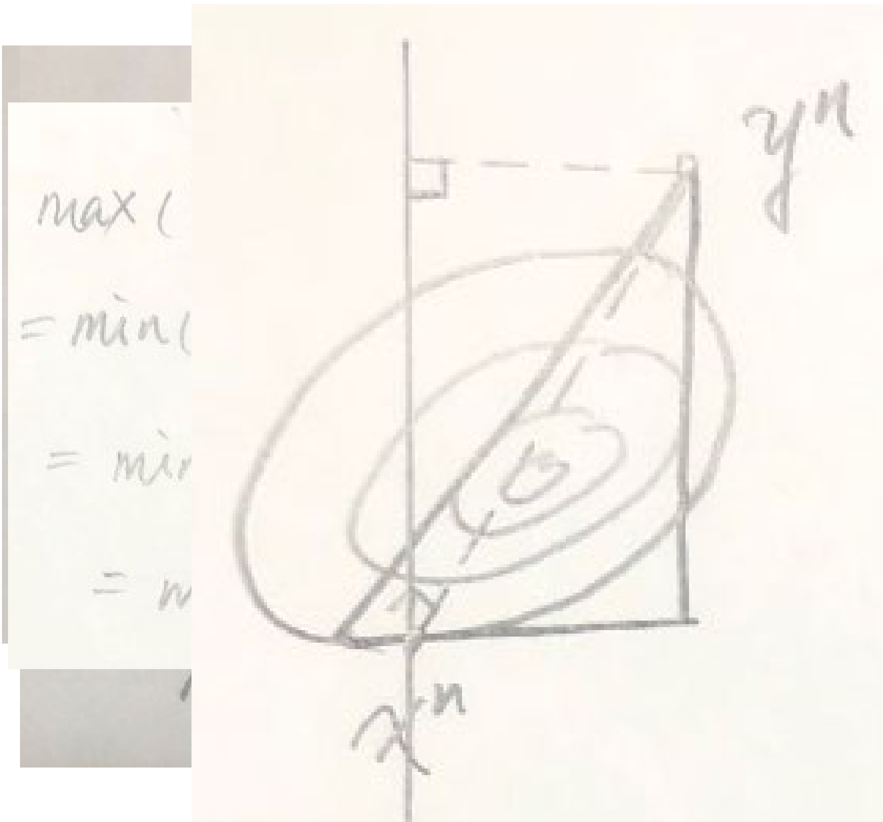
Step 1: 确定可行下降方向$d^{(k)}=y^{(k)}-x^{(k)}$， 其中$y^{(k)}$ 是 如下问题的最优解:

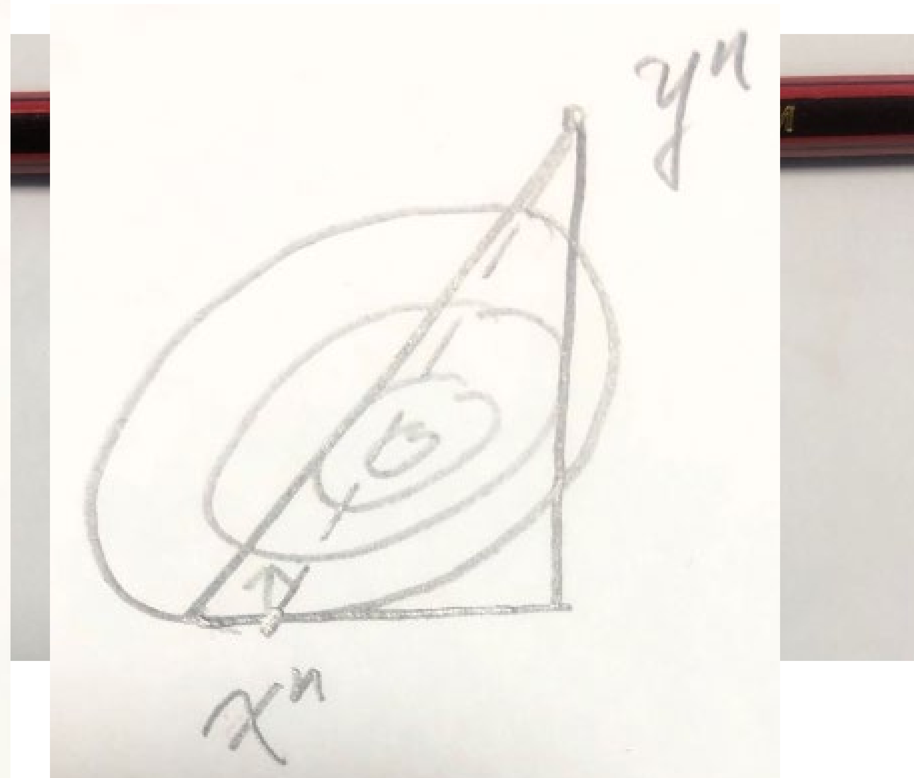
$$\min_{y} y^{\mathrm{T}} \nabla f\left(x^{(k)}\right)$$

subject to

$$g(y) \leq 0$$

$$h(y) = 0$$


max (
= min(
y^n
x^n


y^n
x^n

Frank-Wolfe 算法

Step 2:确定α_k： 通过一维搜索法求解如下一元最小化问题的最优解来确定α_k

$$\min_{0\le\alpha\le 1} f\left(x^{(k)}+\alpha\left(y^{(k)}-x^{(k)}\right)\right)$$

Step 3: 令$x^{(k+1)}=x^{(k)}+\alpha_k\left(y^{(k)}-x^{(k)}\right)$ 和 k=k+1

Step 4: 如果$\left|\sqrt{\sum_{i=1}^{n}\left[\left(x_i^{(k)}-x_i^{(k-1)}\right)^2\right]}\right|\le\varepsilon$,则停止迭代；否则，转到 Step 1

Frank-Wolfe 算法

- 例

用F-W算法求解下面最小化问题的最优解，停止条件为 ε=0.01 ，初始解为 $x^{(0)}=(0,0)$

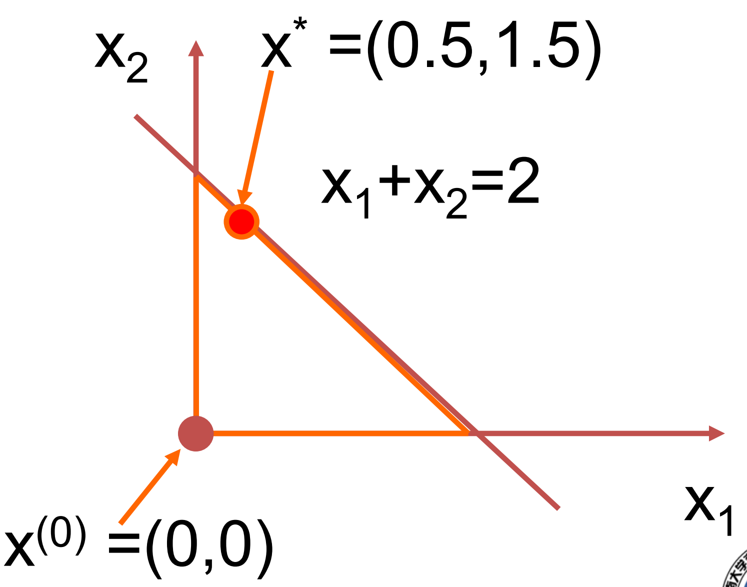
$$\min f(x)=(x_1-2)^2+(x_2-3)^2$$

subject to

$$x_1+x_2\le 2$$

$$x_1\ge 0$$

$$x_2\ge 0$$



Frank-Wolfe 算法

迭代 1 (k=0)

$$\nabla f\left(x^{(0)}\right)=(-4,-6)^T$$

$x^{(0)}=(0,0)^T$

(1)解下面的线性规划问题确定$y^{(0)}$：

$$\min z(y)=-4y_1-6y_2$$

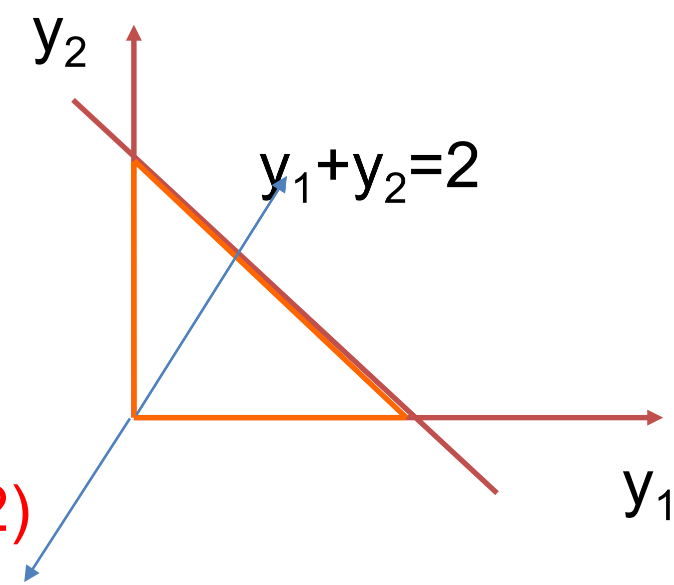
subject to

$$y_1+y_2\le 2$$

$$y_1\ge 0$$

$$y_2\ge 0$$

$\Rightarrow$ $y^{(0)}=(0,2)$



(2) 检查停止条件:

$$\left|\sum_{i=1}^{n}\left[\left(\partial f\left(x^{(0)}\right)/\partial x_i\right)\left(y_i^{(0)}-x_i^{(0)}\right)\right]\right|=\left|\left[-4\times(0-0)+(-6)(2-0)\right]\right|>0.01$$

Frank-Wolfe 算法

(3) 通过二分法求下面一元最优化问题的最优解 α_0 ：

$$x^{(0)}+\alpha\left(y^{(0)}-x^{(0)}\right)=(0,0)^T+\alpha\left[(0,2)^T-(0,0)^T\right]=(0,2\alpha)^T$$

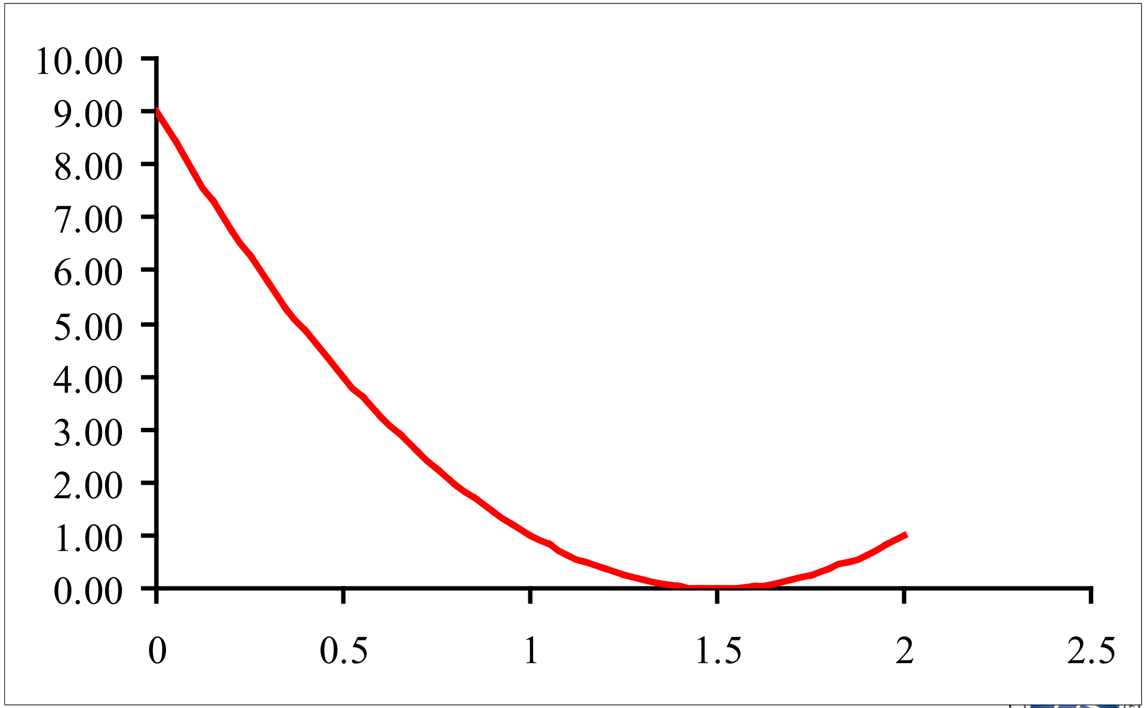
$$\min_{0\le\alpha\le1} f\left(x^{(0)}+\alpha\left(y^{(0)}-x^{(0)}\right)\right)=(0-2)^2+(2\alpha-3)^2 \Rightarrow \alpha_0=1$$

(4) 更新

$$x^{(1)}=x^{(0)}+\alpha_0\left(y^{(0)}-x^{(0)}\right)$$

$$=\begin{pmatrix}0+1\times(0\text{-}0)\\0+1\times(2\text{-}0)\end{pmatrix}$$

$$=\begin{pmatrix}0\\2\end{pmatrix}$$



Frank-Wolfe 算法

<u>迭代 2 (k=1)</u>

$$\nabla f\left(x^{(1)}\right)=(-4,-2)^T \qquad x^{(1)}=(0,2)$$

(1)解下面的线性规划问题确定$y^{(1)}$:

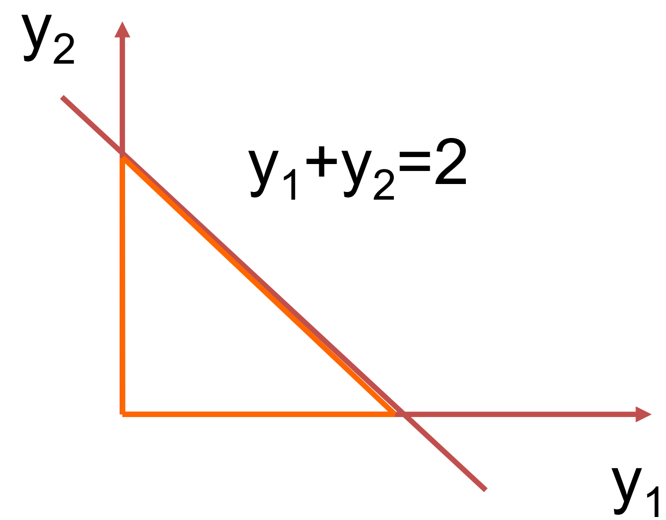
$$\min z(y)=-4y_1-2y_2$$

$$y_1\geq 0$$

$$y_2\geq 0$$

$$y_1+y_2\leq 2$$

$y^{(1)}=(2,0)$



(2)检查停止条件:

$$\left|\sum_{i=1}^{n}\left[\left(\partial f\left(x^{(1)}\right)/\partial x_i\right)\left(y_i^{(1)}-x_i^{(1)}\right)\right]\right|=\left|\left[-4\times(2-0)+(-2)(0-2)\right]\right|>0.01$$

Frank-Wolfe 算法

(3)通过二分法求下面一元最优化问题的最优解 α_0：

$$\min_{0\le\alpha\le1} f\left(x^{(1)}+\alpha\left(y^{(1)}-x^{(1)}\right)\right)=(2\alpha-2)^2+(-2\alpha-1)^2 \Rightarrow \alpha_1=0.25$$

(4)更新

$$x^{(2)}=x^{(1)}+\alpha_1\left(y^{(1)}-x^{(1)}\right)$$

$$=\begin{pmatrix}0+0.25\times(2\text{-}0)\\ 2+0.25\times(0\text{-}2)\end{pmatrix}$$

$$=\begin{pmatrix}0.5\\ 1.5\end{pmatrix}$$